



**МОДЕЛИРОВАНИЕ НА ЭВМ
ОБРАБОТКИ
ГЕОЛОГО-ГЕОФИЗИЧЕСКОЙ ИНФОРМАЦИИ**

**ПРОГРАММЫ
„КОНТУР“, „ГРАФИН“, „СИМВОЛ“**

(МЕТОДИЧЕСКОЕ ПОСОБИЕ)

НОВОСИБИРСК-1971

ИНСТИТУТ ГЕОЛОГИИ И ГЕОФИЗИКИ СО АН СССР
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИБИРСКИЙ НАУЧНО- ИССЛЕДОВАТЕЛЬСКИЙ ИНСТИТУТ
ГЕОЛОГИИ ГЕОФИЗИКИ И МИНЕРАЛЬНОГО СЫРЬЯ
МИНИСТЕРСТВО ГЕОЛОГИИ СССР

**МОДЕЛИРОВАНИЕ НА ЭВМ
ОБРАБОТКИ
ГЕОЛОГО-ГЕОФИЗИЧЕСКОЙ ИНФОРМАЦИИ**

**ПРОГРАММЫ
„КОНТУР“, „ГРАФИК“, „СИМВОЛ“**

НОВОСИБИРСК-1971

УДК 550-831+550.838:550.83

А Н Н О Т А Ц И Я

Рассматриваются задачи, наиболее часто встречающиеся в практике при обработке геолого-геофизических данных и предлагаются методы их решения с помощью цифровых вычислительных машин и графических устройств. Прежде всего, это методы построения контурных линий, позволяющие строить контуры в виде плавных и ломаных линий. Вторую группу составляют методы построения символов. Эти методы в сочетании с методами построения контурных линий дают возможность автоматизировать построение геологических и геофизических разрезов и карт. Для перечисленных методов приводятся оптимальные варианты программ для счета на ЭВМ типа М-20, М-220, БЭСМ-3М, БЭСМ-4.

О т в е т с т в е н н ы й р е д а к т о р
ч л е н - к о р р е с п о н д е н т А Н С С С Р
Э.Э.Фотиади

А в т о р ы - с о с т а в и т е л и:

А. Борискин, В.А.Головкин, Н.А.Клемберг,
В.С.Кривошукский, Н.А.Осипов, К.А.Шемякина,
М.Л.Шемякин.

Настоящий выпуск является **пятым** в серии опубликованных нами методических пособий. Он включает программы "Контур", "График", "Символ", выполняющие автоматическое построение с помощью цифровых вычислительных машин (ЦВМ) и графопостроителей (ГП), применяемых при обработке и интерпретации данных геологии и геофизики. Например, контурные линии на схемах тектонического районирования, районирования по вещественному составу, структурных схемах и другую нагрузку на специальных картах - оси и линии, на графиках, а также различные надписи и подписи.

Изображение информации в графическом виде широко используется в практике обработки геолого-геофизических данных. Задачи, решаемые с помощью публикуемых программ, относятся к разработкам систем отображения геолого-геофизических данных с помощью ЦВМ и графопостроителей, которые относятся к весьма актуальным в общей проблеме создания автоматических систем управления и обработки геологической информации.

Настоящее пособие состоит из двух частей. В первой из них описаны методы построения контурных геолого-геофизических линий с помощью программ "Контур". Построения контурных линий производятся без изломов. В одном методе построение производится с помощью полиномов третьей степени, в другом с помощью дуг окружностей, в третьем с помощью окружностей и прямых. Описывается также метод построения ломаных контурных линий. Для каждого метода приведено краткое его содержание, структура и работа операторной схемы, программа для работы на ЭВМ типа М-20, М-220 и, наконец, инструкция к программе.

Во второй части пособия описываются элементы автоматического оформления графиков, осуществляемое с помощью программ "График" и "Символ". Элементы автоматического оформления графиков включают контроль вычерчивания кривых, выбор формата и смену листа, кадрирование, выбор масштаба,

координатные оси и сетку, выбор шага осей, рамку, символы и текст.

Перечисленные операции могут с помощью описываемых программ осуществляться раздельно в следующих трех вариантах. В одном случае программы обеспечивают нанесение разнообразных символов и текста. В другом случае по желанию с помощью этих программ выполняется выбор формата и смена листа, кадрирование, выбор масштаба, вычерчивание рамок, осей, сетки, после чего осуществляется вычерчивание кривых. В третьем случае с помощью программ может выполняться вычерчивание кривой с контролем по заданному массиву точек.

Настоящая работа является результатом исследований сотрудников Новосибирского государственного университета Министерства высшего и среднего специального образования РСФСР, Института Геологии и Геофизики Сибирского отделения АН СССР, Института геологии, геофизики и минерального сырья Министерства Геологии СССР и Научно-производственного объединения "Факел" при Советском РК ВЛКСМ г. Новосибирска.

Часть первая методического пособия написана А.Борискиным, Н.Клемберг, В.Кривоцуцким, Н.Осиповым, М.Шемякиным. Часть вторая - В.Головкиным, К.Шемякиной, М.Шемякиным.

Выполнению работы способствовало постоянное внимание, советы и консультации Ю.А.Воронина, Н.Г.Загоруйко, Г.И.Каратаева, В.А.Львова, В.С.Суркова. Большую помощь в оформлении работы оказала Д.Я.Юренас.

Член-корреспондент АН СССР
Э.Э.Фотиади

**МЕТОДЫ ПОСТРОЕНИЯ КОНТУРНЫХ
ГЕОЛОГО- ГЕОФИЗИЧЕСКИХ ЛИНИЙ**

Часть I

§1 МЕТОД ПОСТРОЕНИЯ КОНТУРНОЙ ЛИНИИ С ПОМОЩЬЮ ПОЛИНОМОВ ТРЕТЬЕЙ СТЕПЕНИ

Соединить любые две точки кривой можно бесконечным числом способов (см. рис. 1а). При построении линии вместе с набором точек дается условие - соединить эти точки без изломов с минимальным количеством точек перегиба (см. рис. 1б). В данном случае предлагается один из способов соединения точек - с помощью полиномов третьей степени.

$$a_0 + a_1 \cdot x + a_2 x^2 + a_3 x^3 = y.$$

Между каждой парой соседних точек строится свой полином, который, естественно, является "гладкой кривой", а для того, чтобы избежать излома в местах "стыкования" полиномов (т.е. в заданных точках), нужно задать одинаковую производную для стыкующихся полиномов в точках стыкования.

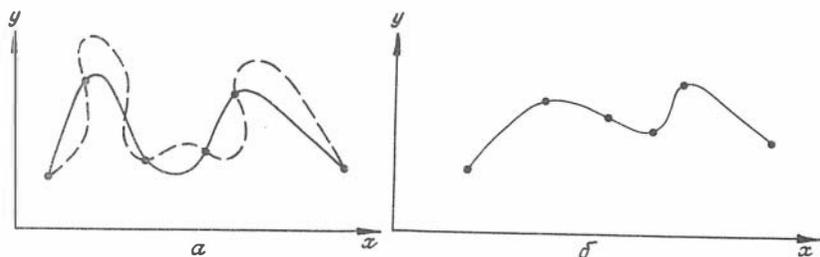


Рис.1

Пусть x_i, y_i — координаты точки под номером i , $i = 1, \dots, n$, где n — число точек, по которым нужно построить линию. Производную f_i в каждой точке i можно задать следующим образом

$$f_i = \frac{y_{i+1} - y_{i-1}}{x_{i+1} - x_{i-1}}, \quad \text{для } i = 2, \dots, n-1,$$

т.е. наклон прямой, имеющей производную f_i , совпадающей с наклоном прямой, проведенной через точки $i-1$, $i+1$. Для незамкнутой линии производные в первой и в последней точках заданы, например, наклоном прямой, соединяющей первую и вторую, предпоследнюю и последнюю точки соответственно. Для замкнутой линии, для которой координаты первой и последней точек совпадают, производная

$$f_1 = f_n = \frac{y_2 - y_{n-1}}{x_2 - x_{n-1}} \quad (\text{см. рис. 2})$$

Так как прямая и парабола являются частным случаем полинома третьей степени, то построение с помощью полиномов третьей степени охватывает метод построения параболами и прямыми.

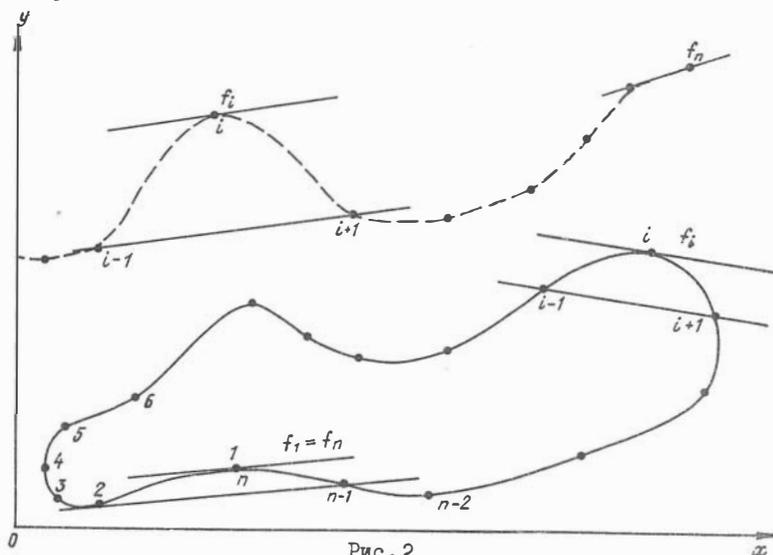


Рис. 2

Итак, мы имеем набор точек с координатами (x_i, y_i) и производными в них f_i ($i = 1, \dots, n$). Построим полином между каждой парой точек i -й и $i+1$ -й. Для этого нужно найти четыре величины, коэффициенты при степенях x . Составим уравнения из тех условий, что полином должен проходить через точки i и $i+1$ и иметь в этих точках производные f_i и f_{i+1}

$$\left\{ \begin{array}{l} a_0 + a_1 x_i + a_2 x_i^2 + a_3 x_i^3 = y_i \\ a_0 + a_1 x_{i+1} + a_2 x_{i+1}^2 + a_3 x_{i+1}^3 = y_{i+1} \\ a_1 + 2a_2 x_i + 3a_3 x_i^2 = f_i \\ a_2 + 2a_2 x_{i+1} + 3a_3 x_{i+1}^2 = f_{i+1} \end{array} \right.$$

Получились четыре уравнения с четырьмя неизвестными. Из линейной алгебры известно, что такого типа уравнения имеют единственное решение только в том случае, если детерминант, составленный из коэффициентов уравнений системы, не равен нулю

$$\begin{vmatrix} 1 & x_i & x_i^2 & x_i^3 \\ 1 & x_{i+1} & x_{i+1}^2 & x_{i+1}^3 \\ 0 & 1 & 2x_i & 3x_i^2 \\ 0 & 1 & 2x_{i+1} & 3x_{i+1}^2 \end{vmatrix} = -(x_{i+1} - x_i)^4.$$

Таким образом, единственное решение существует в том случае, если $x_{i+1} - x_i \neq 0$, т.е. точки не лежат на прямой, параллельной оси y . Для того, чтобы производные имели конечное значение, требуется, чтобы $x_{i+2} - x_i \neq 0$ и $x_{i+1} - x_{i-1} \neq 0$. Условия существования полинома принимают вид:

$$x_{i+1} - x_i \neq 0,$$

$$x_{i+2} - x_i \neq 0,$$

$$x_{i+1} - x_{i-1} \neq 0.$$

Решая систему уравнений, получаем, что коэффициенты полинома имеют следующие значения:

$$a_3 = \frac{2(y_i - y_{i+1})}{(x_i - x_{i+1})^3} + \frac{f_i + f_{i+1}}{(x_i - x_{i+1})^2},$$

$$a_2 = \frac{(f_i - f_{i+1})}{2(x_i - x_{i+1})} - \frac{3a_3(x_i + x_{i+1})}{2},$$

$$a_1 = f_i - 3a_3 x_i^2 - 2a_2 x_i,$$

$$a_0 = y_i - a_1 x_i - a_2 x_i^2 - a_3 x_i^3.$$

Сделаем предположения: пусть в плоскости x, y имеется прямоугольная сетка с шагом h по оси x и с шагом h_1 по оси y . Точки заданы так, что они лежат или на линии, параллельной оси y , или на линии, параллельной оси x , но не на пересечении их. В одном квадрате может лежать или две соседних точки, или ни одной. Рассмотрим варианты положения точек в квадрате. Вариант первый — точка i находится на левой стороне квадрата, $i+1$ — на правой (рис. 3а). Условие $x_{i+1} - x_i \neq 0$ выполняется, рассмотрим, будут ли выполняться условия $x_{i+2} - x_i \neq 0$, $x_{i+1} - x_{i-1} \neq 0$. Точка $i-1$ лежит на одной из линий квадрата правее точки i , но не на пересечениях, а точка $i+2$ правее точки $i+1$, таким образом, условия выполняются. Построение полинома, зависящего от x , возможно при таком варианте.

Вариант второй (рис. 3б), точка i лежит на нижней стороне, $i+1$ — на верхней стороне квадрата. Если взять полином не по переменной x , а по переменной y , то этот случай полностью эквивалентен первому.

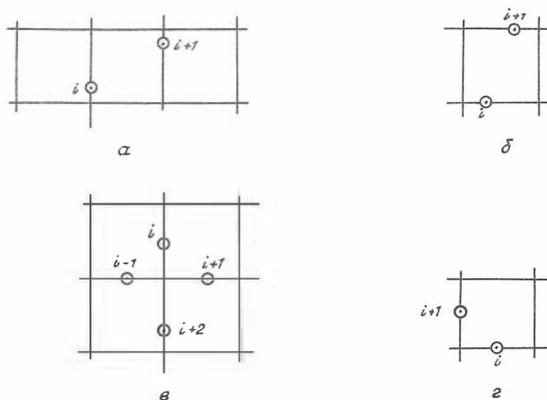


Рис. 3

Вариант третий — точка i лежит на прямой, параллельной оси y , точка $i+1$ лежит на прямой, параллельной оси x (рис. 3в). С точки зрения построения полиномами вариант самый неблагоприятный, потому что возможен случай, когда точки $i-1$ и $i+2$ лежат так, как указано на рис. 3в. В данном варианте невозможно использование полинома от переменной x или y . В этом случае приходится строить промежуточную точку с координатами $(x_0; y_0)$, лежащую внутри квадрата, на сторонах которого лежат i и $i+1$ точки; производную в ней брать

$$f_0 = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}$$

и находить полином по переменной x от i -й точки до 0-й. Затем от 0-й до $i+1$ -й находить полином по переменной y . Если же $y_{i+1} - y_{i-1} \neq 0$ или $x_i - x_{i+2} \neq 0$, то возможно использование полинома от точки i до точки $i+1$ по какой-то одной переменной. Если условие $x_{i+1} - x_{i-1} \neq 0$ выполняется, то по переменной y . Если условие $x_{i+2} - x_i \neq 0$ выполняется, то по переменной x .

Четвертый вариант (см.рис. 3г) заменой x на y приводится к третьему варианту. Данные варианты охватывают вероятные положения точек, а значит, дают возможность построить гладкую линию по заданным точкам.

Возможен еще один вариант построения линии полинома третьей степени, используя повороты системы координат при построении между каждой парой точек.

1. О п и с а н и е о п е р а т о р н о й с х е м ы п р о г р а м м ы "К о н т у р - 1"

В состав операторной схемы алгоритма входят следующие операторы:

- A_1 - ввод информации.
- A_2 - нахождение номеров первой и последней точек в линии.
- A_3 - проверка последней точки.
- A_4 - проверка первой точки.
- A_5 - проверка линии на замкнутость.
- A_6 - определение варианта расположения точек.
- $A_7, A_{10}, A_{17}, A_{21}, A_{29}, A_{35}, A_{37}, A_{42}$ - вычисление производной в i точке.
- $A_8, A_{11}, A_{18}, A_{22}, A_{30}, A_{34}, A_{40}, A_{45}$ - вычисление производной в $i+1$ точке.
- $A_9, A_{12}, A_{39}, A_{41}, A_{46}$ - вычисление коэффициентов полинома и вывод точек на печать через заданный шаг.
- P_{13}, P_{25} - проверка первой точки.
- $P_{14}, P_{24}, P_{28}, P_{32}$ - проверка возможности построения полинома по x .
- P_{15}, P_{27}, P_{49} - проверка последней точки.
- $P_{16}, P_{20}, P_{26}, P_{36}$ - проверка возможности построения полинома по y .
- $P_{19}, P_{23}, P_{31}, P_{35}$ - проверка возможности построения полинома при данных производных.
- A_{38}, A_{43} - вычисление координат промежуточной точки и производной в ней.

- K_{47} - счетчик очередной точки.
 P_{48} - проверка последней точки последней линии.
 A_{50} - вывод недозаполненных массивов.
 $Я_{51}$ - конец работн.

Структура операторной схемн имеет вид:

A_1 ^{1,48} A_2 ^{2,49} A_3 A_4 A_5 P_6 $M_1 \div M_6$ ^{6,14,24,31,35} A_7 A_8 K_9 ⁴⁷ ^{6,19,23,26,36} A_{10} A_{11} K_{12} ⁴⁷
⁶ $P_{13 \div 15}$ ⁷ $P_{14 \div 17}$ ¹³ $P_{15 \div 20}$ $P_{16 \div 17}$ A_{17} A_{18} ¹⁰ $P_{19 \div 17}$ ¹⁵ $P_{20 \div 24}$ A_{21} A_{22} ¹⁰ $P_{23 \div 17}$
²⁰ $P_{24 \div 17}$ ⁶ $P_{25 \div 27}$ ¹⁰ $P_{26 \div 42}$ ²⁵ $P_{27 \div 32}$ $P_{28 \div 42}$ A_{29} A_{30} ⁷ $P_{31 \div 42}$ ²⁷ $P_{32 \div 36}$
 A_{33} A_{34} ⁷ $P_{35 \div 42}$ ³² $P_{36 \div 42}$ ^{6,14,16,19,23,24} A_{37} A_{38} A_{39} A_{40} A_{41} ⁴⁷ ^{6,26,28,31,35,36} A_{42}
 A_{43} A_{44} A_{45} A_{46} ^{9,12,46,47} K_{47} ⁵⁰ P_{48} ⁴² $P_{49 \div 3}$ ⁴⁸ A_{50} $Я_{51}$.

Операторная схема работает следующим образом. Ввод информации в ЦВМ осуществляет оператор A_1 . Оператор A_2 определяет, сколько точек в линии - находит номера первой и последней точек. Последующие операторы A_3 и A_4 анализируют точки i и $i+1$. Оператор A_3 проверяет, не является ли точка i первой точкой линии, а оператор A_4 - не является ли точка $i+1$ последней точкой линии. Результаты проверки запоминаются данными операторами и используются при дальнейшей работе программ. Оператор A_5 проверяет линию на замкнутость и также запоминает результат проверки.

Оператор P_6 определяет вариант расположения точек $i-1$, i , $i+1$ и $i+2$, от которого переключает программу на одну из меток M_1 , M_2 , M_3 , M_4 , M_5 , M_6 .

В случае переключения на метки M_1 или M_2 работа программы организуется следующим образом. Вычисляется производная в i точке (оператор A_7 за меткой M_1 и A_{10} за меткой M_2), затем производная в $i+1$ точке (операторы A_8 и A_{11}) и значения полинома в точках и вывод их на печать через заданный шаг (операторы A_9 и A_{12}). После операторов A_9 и A_{12} происходит переход на оператор K_{47} .

Если программа переключена на метки M_3 или M_4 , то работает оператор P_{13} (за меткой M_3) или P_{25} (за меткой

M_4), которые проверяют, не является ли точка i первой. Если да, то работают операторы P_{14} (после P_{13}) или P_{26} (после P_{25}). Эти операторы проверяют возможность построения полинома по x (P_{14}) или по y (P_{26}). Если построение полинома возможно, то управление передается на метку M_1 (от P_{14}) или на M_2 (от P_{26}). Если же построение полинома невозможно, то управление передается на метку M_5 (от P_{14}) или на M_6 (от P_{26}).

От операторов P_{13} и P_{25} , если точка i не первая, происходит переход соответственно на операторы P_{15} и P_{27} , которые проверяют, не является ли точка i последней. В том случае, если точка последняя, после операторов P_{15} и P_{27} работают операторы P_{16} и P_{28} , проверяющие соответственно возможность построения полинома по y (оператор P_{16}) и по x (P_{28}). В случае, если построение полинома возможно, то далее работают операторы A_{17} (после P_{16}) и A_{29} (после P_{28}), вычисляющие производную в i точке, если нет, то управление от оператора P_{16} передается на метку M_5 , а от оператора A_{29} - на метку M_6 .

После операторов A_{17} и A_{29} работают соответственно операторы A_{18} и A_{30} , вычисляющие производную в $i+1$ точке, а затем операторы P_{19} и P_{31} , проверяющие возможность построения полинома при данных производных. Если построение полинома возможно, то от этих операторов управление передается на метку M_2 (от P_{19}) или на M_1 (от P_{31}), если нет, то соответственно на метку M_5 или M_6 .

Если операторы P_{15} и P_{27} обнаруживают последнюю точку, то от них происходит переход соответственно на операторы P_{20} и P_{32} . Оператор P_{20} проверяет возможность построения полинома по y , оператор P_{32} - по x . Если построение полинома невозможно, то от них управление передается соответственно операторам P_{24} и P_{36} . Если же построение возможно, то работают операторы A_{21} и A_{33} , вычисляющие производную в i точке, далее A_{22} и A_{34} , вычисляющие производную в $i+1$ точке и операторы P_{23} и P_{35} , которые проверяют возможность построения полинома при данных производных и передают управление соответственно на метки M_2 и M_1 , если построение полинома возмож-

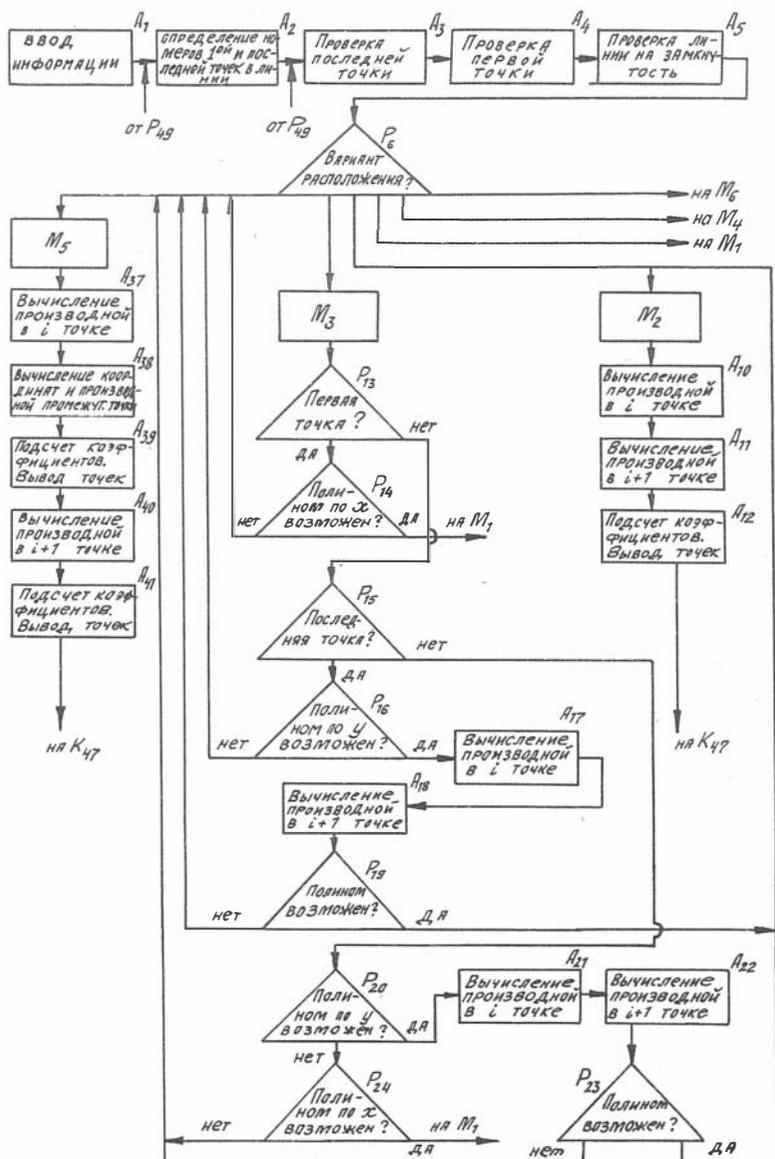
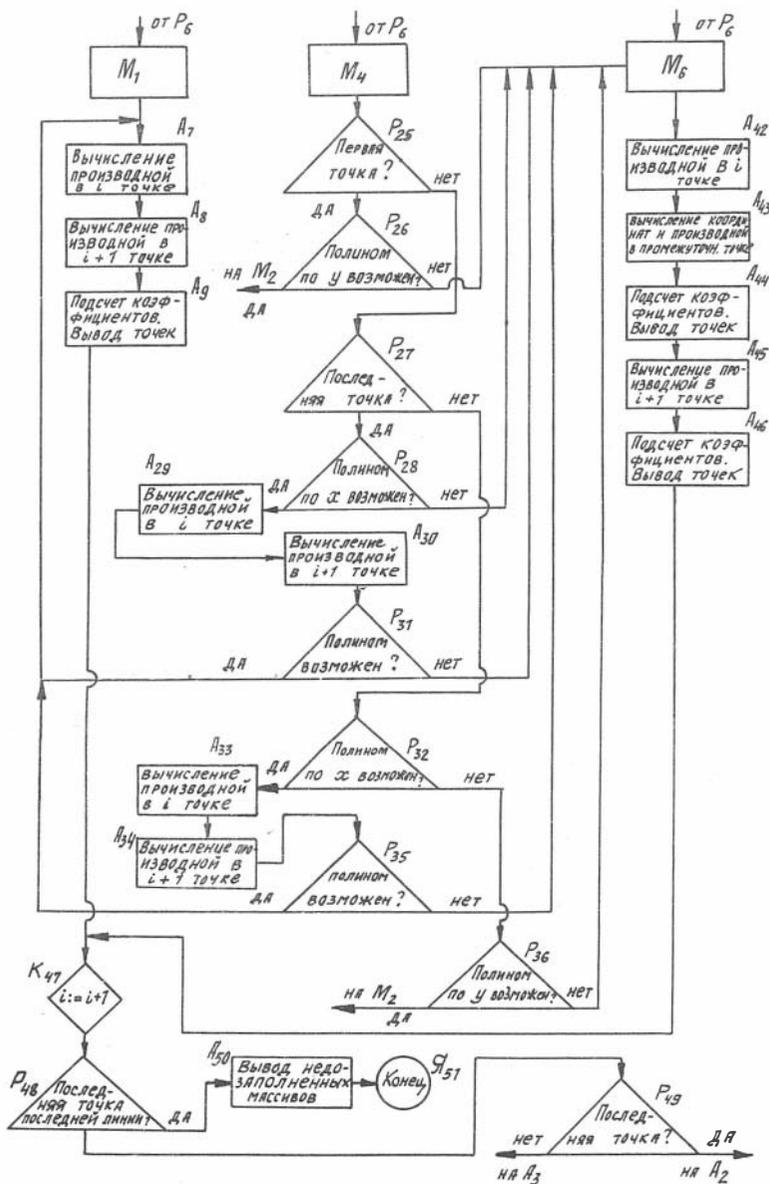


Рис. 4. Блок-схема программы



но и на метки M_5 и M_6 в противном случае.

Операторы P_{24} и P_{36} проверяют возможность построения полинома по x и y соответственно и передают управление на метки M_1 и M_2 , если такая возможность есть, а в противном случае — на метки M_5 и M_6 .

После меток M_5 и M_6 работают соответственно операторы A_{37} и A_{42} , вычисляющие производную в i точке; A_{38} и A_{43} , вычисляющие координаты промежуточной точки и производную в ней; A_{39} и A_{44} , вычисляющие коэффициенты полинома и производящие вывод точек на печать через заданный шаг; A_{40} и A_{45} , вычисляющие производную в $i+1$ точке и операторы A_{41} и A_{46} , которые вычисляют коэффициенты полинома и выводят точки на печать через заданный шаг.

После операторов A_9 , A_{12} , A_{41} и A_{46} работает оператор K_{47} , который является счетчиком очередной точки, т.е. значению i он присваивает значение $i+1$.

Оператор P_{48} проверяет, не является ли точка i последней точкой в последней линии. Если нет, то оператором P_{49} проверяется, не является ли эта точка последней в рассматриваемой линии. Если нет, значит можно рассматривать снова две точки i и $i+1$, происходит переход на оператор A_3 . Если точка последняя, то после P_{48} происходит переход на оператор A_2 за новой линией.

Когда оператор P_{48} обнаруживает последнюю точку, то после него работает оператор A_{50} , производящий вывод недозаполненных массивов и на этом работа заканчивается.

2. И н с т р у к ц и я к п р о г р а м м е "К о н т у р - 1"

В ЭВМ вводятся:

1. Координаты всех вводимых точек по оси x , $K\Sigma$
2. Координаты всех вводимых точек по оси y , $K\Sigma$
3. Шаг сетки по оси x , $K\Sigma$
4. Шаг сетки по оси y , $K\Sigma$
5. Шаг, через который выводятся точки; $K\Sigma$

Вводимых точек может быть не больше 500. В конце линии, т.е. перед $\kappa\Sigma$ записывается число 10^5 как по оси x , так по оси y . После введения всех линий ставится 10^6 также перед $\kappa\Sigma$. Замкнутая линия должна иметь одинаковыми первую и последнюю точки. Если линия состоит из 2-х точек, то последняя точка пробивается 2 раза.

Выдаются на печать массивы по 100 элементов в каждом. Координаты точек содержатся в соответствующих номерах элементов массивов. Если пронумеровать выводимые массивы, то их можно сгруппировать по парам (1,2), (3,4), ..., (2n-1, 2n), ... n=1,2,... Тогда координаты точек содержатся в соответствующих парах, если k номер одного из массивов будет координатой x , то номер второго из массивов в этой паре будет координатой этой точки по оси y . Каждый из выводимых массивов содержит координаты только по одной из осей.

Конец линии отмечается в массиве координат по x и в массиве координат по y одним и тем же числом 10^5 , а конец последней выводимой линии отмечается числом 10^6 .

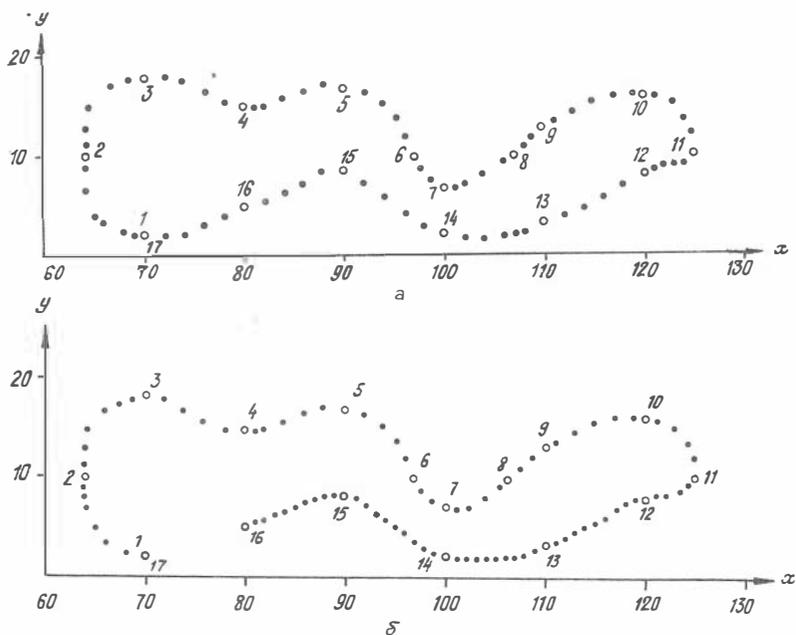


Рис. 5

ГРАФИКИ КОНТУРНОЙ ЛИНИИ а) ЗАМКНУТОЙ, б) НЕЗАМКНУТОЙ

ТАБЛИЦА РАСПРЕДЕЛЕНИЯ ПАМЯТИ

	НАЧАЛО	КОНЕЦ	ДЛИНА	ПРИМЕЧАНИЯ
ПРОГРАММА	0020	1455	1436	
ЧИСЛОВНЕ КОНСТАНТЫ	1456	1516	0041	
ФОРМИРУЕМНЕ КОНСТАНТЫ	1517	1545	0027	
СКАЛЯРН	1546	1644	0077	
РАБОЧИЕ ЯЧЕЙКИ	1645	1652	0006	
МАССИВН				
1	1653	2016	0144	
2	2017	2162	0144	
3	2163	3146	0764	
4	3147	4132	0764	
РАБОЧЕЕ ПОЛЕ ИС	НАЧАЛО	КОНЕЦ	ДЛИНА	
	4133	7500	3345	

АДРЕСА ПРОГРАММНЫХ ОСТАНОВОВ

1455

3. КОНТРОЛЬНЫЙ ПРИМЕР
К ПРОГРАММЕ "КОНТУР - 1"

ТАБЛИЦА ЗАДАНЫХ ТОЧЕК ЛИНИИ

а - замкнутой; б - незамкнутой.

а

<i>N</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
<i>X</i>	70	64	70	80	90	97	100	107	110	120	125	120	110	100	90	80	70
<i>Y</i>	2	10	18	15	17	10	7	10	13	16	10	8	3	2	8	5	2

б

<i>N</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
<i>X</i>	70	64	70	80	90	97	100	107	110	120	125	120	110	100	90	80
<i>Y</i>	2	10	18	15	17	10	7	10	13	16	10	8	3	2	8	5

Т а б л и ц а В ы б

---0-----	+++0270000000	+++0269000000	+++0268000000
+++0264444444	+++02641875000	+++0264055555	+++02640069444
+++0263800000	+++02643515625	+++02654625000	+++02666437371
+++0271000000	+++02720000000	+++02730000000	+++02740000000
+++0279000000	+++02860000000	+++02810000000	+++02820000000
+++0287000000	+++02880000000	+++02890000000	+++02900000000
+++02946262872	+++02958710390	+++02957720091	+++02959868651
+++02996227271	+++03100000000	+++03101000000	+++03102000000
+++03107000000	+++03107999999	+++03108847826	+++03109694769
+++03114000000	+++03115000000	+++03116000000	+++03117000000
+++03122000000	+++03123000000	+++03123750000	+++03124411865
+++03123035714	+++03121931216	+++03120866402	+++03120000000
+++03115000000	+++03114000000	+++03112999999	+++03112000000
+++01236848497	+++01269675946	+++01342009544	+++01400000000
+++01900000000	+++01200000000	+++02110000000	+++02120000000
+++02166926407	+++02171977382	+++02176111420	+++02179671901
+++02174419999	+++02169531250	+++02164279999	+++02159183749
+++02152381178	+++02155437941	+++02159143531	+++02163051474
+++02170000000	+++02166666666	+++02163287356	+++02159359038
+++02119666384	+++02109666384	+++02100000000	+++01877777863
+++01734985399	+++01773994208	+++01823148692	+++01879900297
+++02128162307	+++02130000000	+++02134758461	+++02139667692
+++02159367693	+++02160916920	+++02161195384	+++02160000000
+++02134999999	+++02125000000	+++02115000000	+++02105000000
+++01799999994	+++01750900018	+++01697866667	+++01642299992
+++01375466688	+++01334100026	---0-----	

.....

+++03107999999	+++03108847826	+++03109694769	+++03110000000
+++03115000000	+++03116000000	+++03117000000	+++03118000000
+++03123000000	+++03123750000	+++03124411865	+++03124785665
+++03121931216	+++03120866402	+++03120000000	+++03119000000
+++03114000000	+++03112999999	+++03112000000	+++03111000000
+++03106000000	+++03105000000	+++03104000000	+++03102999999
+++02980000000	+++02970000000	+++02960000000	+++02950000000
+++02900000000	+++02890000000	+++02880000000	+++02870000000
+++02820000000	+++02810000000	+++02800000000	+++02790000000
+++02840000000	+++02850000000	+++02860000000	+++02870000000
+++02919881337	+++02929005065	+++02937866584	+++02946262872
+++02970000000	+++02980000000	+++02987499999	+++02996227271
+++01773994208	+++01823148692	+++01879300297	+++01939300299
+++02130000000	+++02134758461	+++02139667692	+++02144524615
+++02160916920	+++02161195384	+++02160000000	+++02158640003
+++02125000000	+++02115000000	+++02105000000	+++02100000000
+++01750900018	+++01697866667	+++01642299992	+++01585600000
+++01334100026	+++01300000000	+++01270500002	+++01243200001
+++01165950000	+++01168799999	+++011179850000	+++01200000000
+++01512499964	+++01591199982	+++01664099992	+++01726399982
+++01776150006	+++01744800013	+++01706250006	+++01663200015
+++02151526250	+++02150000001	+++02150419706	+++02152381178
+++02169688531	+++02171524708	+++02171777354	+++02170000000
+++02148506862	+++02139666384	+++02129666384	+++02119666384
+++01708892250	+++01700000000	+++01709271109	+++01734985399

++02670000000	++02660000000	++02655000000	++02646680555
++02640000000	++0263978125	++02637375000	++02636484375
++02677529266	++02688325332	++02698955659	++02700000000
++02750000000	++02760000000	++02770000000	++02780000000
++02830000000	++02840000000	++02850000000	++02860000000
++02910589570	++02919881337	++02929005065	++02937865084
++02963094846	++02970000000	++02980000000	++02987499999
++03102999999	++03104000000	++03105000000	++03106000000
++03110000000	++03111000000	++03112000000	++03112999999
++03118000000	++03119000000	++03120000000	++03121000000
++03124785665	++03124953703	++03124998285	++03125000000
++03119000000	++03118000000	++03117000000	++03116000000
++03111000000	-----0-----	++01199999976	++01221099162
++01500000000	++01600000000	++01700000000	++01800000000
++02130000000	++02140000000	++02150000000	++02160000000
++02180000000	++02181736250	++02181040001	++02178422749
++02154760001	++02151526250	++02150000001	++02150419700
++02166715295	++02169688531	++02171524708	++02171777354
++02154613059	++02148506862	++02139666384	++02129666364
++01782466316	++01708892250	++01700000000	++01709271109
++01999300299	++02100000000	++02109418869	++02119741783
++02144524615	++02149126153	++02153269228	++02156750767
++02153640003	++02156853351	++02152080001	++02145000000
++02100000000	++01900000000	++01866666667	++01833333333
++01585600000	++01529166687	++01474400002	++01422699999

.....

++03111000000	++03112000000	++03112999999	++03114000000
++03119000000	++03120000000	++03121000000	++03122000000
++03124953703	++03124998285	++03125000000	++03123035714
++03118000000	++03117000000	++03116000000	++03115000000
++03110000000	++03109000000	++03107999999	++03107000000
++03102000000	++03101000000	++03100000000	++02990000000
++02940000000	++02930000000	++02920000000	++02910000000
++02860000000	++02850000000	++02840000000	++02830000000
++02800000000	++02810000000	++02820000000	++02830000000
++02880000000	++02890000000	++02900000000	++02910589570
++02953710390	++02957720091	++02959868651	++02963094846
++03100000000	++03101000000	++03102000000	-----0-----
++02100000000	++02109418869	++02119741783	++02128162307
++02149126153	++02153269228	++02156750767	++02159367693
++02156853351	++02152080001	++02145000000	++02134999999
++01900000000	++01866666667	++01833333333	++01799999994
++01529166687	++01474400002	++01422699999	++01375466668
++01285499999	++01197800001	++01181250000	++01170399999
++01235699975	++01289599967	++01356900000	++01432799971
++011773300004	++01799999994	++01806449991	++01797599983
++01618350011	++01574400002	++01534050011	++07100000000
++02155437941	++02159143531	++02163051474	++02166715295
++02166666666	++02163287356	++02159359038	++02154613059
++02109666384	++02100000000	++01877777863	++01782466316

-----0-----

4. ПРОГРАММА "КОНТУР - 1" (на Альфа - языке)

Лист 1

1 Начало массив $x, y [1:500], M, N [1:100]$; вещ $x_1, x_2, y_1, y_2, f_1, f_2, S, h, h_1, a_0, a_1$;
 2 целый $n_1, n_2, n_3, i, k, t, n, \varphi, \psi$; логич l, l_1, l_2 ; проц $ПЯ_1(x, y, i, k, l_2, f_1, n_1)$;
 3 {если $i < n_1 + 0,5$ то {если l_2 то $f_1 := (y[i+1] - y[k-2]) / (x[i+1] - x[k-2])$ иначе $f_1 :=$
 4 $(y[i+1] - y[i]) / (x[i+1] - x[i])$ } иначе $f_1 := (y[i+1] - y[i-1]) / (x[i+1] - x[i$
 5 $-1])$ }; проц $ПЯ_2(x, y, i, k, l_2, f_2, n_1)$; {если $i > k - 2,5$ то {если l_2 то $f_2 := (y[n_1+1] - y[$
 6 $i]) / (x[n_1+1] - x[i])$ иначе $f_2 := (y[i+1] - y[i]) / (x[i+1] - x[i])$ } иначе $f_2 := (y[i+2$
 7 $] - y[i]) / (x[i+2] - x[i])$ }; проц $ТЧ(x_1, x_2, y_1, y_2, f_1, f_2, S, n_2, n_3, \varphi, \psi, M,$
 8 $N)$; {вещ $a_0, a_1, a_2, a_3, a, \sigma, k_1, f$; $a_3 := -2 \times (y_1 - y_2) / (x_1 - x_2) + 3 \times (f_1 + f_2) /$
 9 $(x_1 - x_2) + 2$; $a_2 := (f_1 - f_2) / 2 / (x_1 - x_2) - 3/2 \times a_3 \times (x_1 + x_2)$; $a_1 := f_1 - 3 \times a_3 \times x_1$
 0 $+ 2 - 2 \times a_2 \times x_1$; $a_0 := y_1 - a_1 \times x_1 + a_2 \times x_1 + 2 - a_3 \times x_1 + 3$; $f := 0$; для $k_1 := 0, k_1 + f$ пока

23

Лист 2

11 $k_1 < a_0 \sigma (x_2 - x_1)$ цикл {если $x_2 - x_1 > 0$ то $a := x_1 + k_1$ иначе $a := x_1 - k_1$; $v := a_0 + a_1 \times a + a_2 \times a + 2$
 12 $+ a_3 \times a + 3$; $f := a_0 \sigma (a_1 + 2 \times a_2 \times a + 3 \times a_3 \times a + 2)$; если $f > 0$ то $f := S / f$ иначе $f := S$; если $f < S / 3$
 13 то $f := S / 3$; $M[n_2] := a$; $N[n_2] := v$; если $a_0 \sigma (n_2 - 100) < 10^{-6}$ то {вывод ($M,$
 14 N); $n_2 := 1$;
 15 $n_3 := \overline{n_3 + 1}$ иначе $n_2 := n_2 + 1$ }; свод (x, y, h, h_1, S); $n_1 := n_2 := n_3 := 1$; $q_2 : l :=$
 16 $l_1 := l_2 :=$ ложь; для $i = n_1, \dots, 500$ цикл { $k := i$; если $a_0 \sigma (x[i] - 10^5) / 10^5 < 10^{-6}$ то на q ; если $a_0 \sigma$
 17 $(x[i] - 10^6) / 10^6 < 10^{-6}$ то на q_1 }; $l :=$ истина; на q ; $q_1 : l_1 :=$ истина; $q :$ если
 18 l то $k := k + 1$; для $i = n_1, \dots, k - 2$ цикл { $x_1 := x[i]$; $x_2 := x[i+1]$; $y_1 := y[i]$; $y_2 := y[i+1]$ };
 19 если $a_0 \sigma (x[n_1] - x[k-1]) < 10^{-6}$ и $a_0 \sigma (y[n_1] - y[k-1]) < 10^{-6}$ то $l_2 :=$ истина; $\varphi := 0$; $\psi := 1$;

Лист 3

20 если $abs(x1-x2) <_{10}^{-6} \wedge abs(y1-y2) <_{10}^{-6}$ то на q6; $a0 := x1/h$; $a1 := entier(a0)$; если
 21 $abs(a0-a1) <_{10}^{-6}$ то $\{m := a1$; если $(x1-x2) > 0$ то $m := m-1$; $n := entier(y1/h1)$; если $y1 <$
 22 $n \times h1$ то $n := n-1$ иначе $\{n := entier(y1/h1)$; если $y1-y2 > 0$ то $n := n-1$; $m := entier(x1/h)$;
 23 если $x1 < m \times h$ то $m := m-1$ }; если $(h-abs(x2-x1)) <_{10}^{-6}$ то на M1; если $(h1-abs(y2-y1)) <_{10}^{-6}$
 24 то на M2; если $abs(x1-m \times h) <_{10}^{-6} \vee abs(x1-(m+1) \times h) <_{10}^{-6}$ то на M3; если $abs(y1-n \times h1)$
 25 $<_{10}^{-6} \vee abs(y1-(n+1) \times h1) <_{10}^{-6}$ то на M4;
 26 на q6;
 27 $M1 : \{ \text{ПЯ}_1(x, y, i, k, l_2, f_1, n1)$;
 28 $\text{ПЯ}_2(x, y, i, k, l_2, f_2, n1)$ };
 29 если $(abs(f_1) < 1/3 \vee abs(f_2) < 1/3) \wedge abs(f_1+f_2) > 4$ то на

Лист 4

30 $M5$; ТЧ $(x1, x2, y1, y2, f1, f2, S, n2, n3, \varphi, \psi, M, N)$; на q6
 31 $\}$; $M2 : \{ \text{ПЯ}_1(y, x, i, k, l_2, f_1, n1)$;
 32 $\text{ПЯ}_2(y, x, i, k, l_2, f_2, n1)$ };
 33 если $(abs(f_1) < 1/3 \vee abs(f_2) < 1/3) \wedge abs(f_1+f_2) > 4$ то на
 34 $M6$; ТЧ $(y1, y2,$
 35 $x1, x2, f1, f2, S, n2, n3, \psi, \varphi, N, M)$; на q6}; $M3 : \{$
 36 если $i < n1 + 0.5$ то если $abs(x[i+2]$
 37 $-x1) >_{10}^{-6}$ то на M1 иначе на M5}; если $i > k - 2.5$ то если $abs(y[i-1]-y2) >_{10}^{-6}$ то
 38 $\{ \text{ПЯ}_1(y, x, i, k, l_2, f_1, n1)$;
 39 $\text{ПЯ}_2(y, x, i, k, l_2, f_2, n1)$ };
если $(abs(f_1) < 1/3 \vee abs(f_2) < 1/3) \wedge abs(f_1+f_2) > 4$ то на
 $M5$; на M2}

40 иначе на M5; если $\text{abs}(y[i-1]-y_2) > 10^{-6}$ то
 41 $\{ \text{ПЯ}_1(y, x, i, k, l_2, f_1, n_1); \text{ПЯ}_2(y, x, i, k, l_2, f_2, n_1);$
 42 если $(\text{abs}(f_1) < 1/3 \vee \text{abs}(f_2) < 1/3) \wedge \text{abs}(f_1+f_2) > 4$ то на
 43 M5; на M2 }
 44 ; если $\text{abs}(x[i+2]-x_1) > 10^{-6}$ то на M1; M5:
 45 $\{ \text{ПЯ}_1(x, y, i, k, l_2, f_1, n_1); f_2 := (y_2 - y_1) / (x_2 - x_1); x_2 := (x_1 + 3 \times x_2) / 4; y_2 :=$
 46 $(3 \times y_1 + y_2) / 4; \text{T4}(x_1, x_2, y_1, y_2, f_1, f_2, S, n_2, n_3, \varphi, \psi, M, N); x_1 := x_2;$
 47 $y_1 := y_2; x_2 := x[i+1]; y_2 := y[i+1]; f_1 := 1/f_2; \text{ПЯ}_2(y, x, i, k, l_2, f_2, n_1); \text{T4}$
 48 $(y_1, y_2, x_1, x_2, f_1, f_2, S, n_2, n_3, \psi, \varphi, N, M);$ на q6 } } ; M4 : {
 49 если $i < n_1 + 0.5$ то { если abs

50 $(y[i+2] - y_1) > 10^{-6}$ то на M2 иначе на M6; если $i > k - 2.5$ то { если $\text{abs}(x[i-1]-x_2) > 10^{-6}$ то
 51 $\{ \text{ПЯ}_1(x, y, i, k, l_2, f_1, n_1); \text{ПЯ}_2(x, y, i, k, l_2, f_2, n_1);$
 52 если $(\text{abs}(f_1) < 1/3 \vee \text{abs}(f_2) < 1/3) \wedge \text{abs}(f_1+f_2) > 4$ то на
 53 M6; на M1 }
 54 иначе на M6; если $\text{abs}(x[i-1]-x_2) > 10^{-6}$ то
 55 $\{ \text{ПЯ}_1(x, y, i, k, l_2, f_1, n_1); \text{ПЯ}_2(x, y, i, k, l_2, f_2, n_1);$
 56 если $(\text{abs}(f_1) < 1/3 \vee \text{abs}(f_2) < 1/3) \wedge \text{abs}(f_1+f_2) > 4$ то на
 57 M6; на M1 }
 58 ; если $\text{abs}(y[i+2]-y_1) > 10^{-6}$ то на M2 ;
 59 M6 : { $\text{ПЯ}_1(y, x, i, k, l_2, f_1, n_1); f_2 := (x_2 - x_1) / (y_2 - y_1); x_2 := (3 \times x_1 + x_2) / 4;$

Лист 7

60 $y_2 := (z * y_2 + y_1) / 4$; ТЧ ($y_1, y_2, x_1, x_2, f_1, f_2, S, n_2, n_3, \varphi, \psi, N, M$); $f_1 := 1 / f_2$
61 ; $x_1 := x_2$; $y_1 := y_2$; $x_2 := x[i+1]$; $y_2 := y[i+1]$; ПЯ₂($x, y, i, k, l_2, f_2, n_1$); ТЧ(
62 $x_1, x_2, y_1, y_2, f_1, f_2, S, n_2, n_3, \varphi, \psi, M, N$)); q_6 }; если и то $n_1 := n_1 + k - 1$
63 иначе $n_1 := 1 + k$; если и то { $M[n_2] := x[k]$; $N[n_2] := y[k]$; вывод (M, N);
64 на q_5 }; если и то { $M[n_2] := N[n_2] := x[k]$; если $abs(n_2 - 100) < 10^{-6}$ то {
65 вывод (M, N); $n_2 := 1$; $n_3 := n_3 + 1$ } иначе $n_2 := n_2 + 1$ }; на q_2 ; q_5 : конец*

§2. МЕТОД ПОСТРОЕНИЯ КОНТУРНОЙ ЛИНИИ ДУГАМИ ОКРУЖНОСТЕЙ

Если задан набор точек $\vec{z}_i = (x_i, y_i)$, $i = 1, 2, \dots, n$, принадлежащих некоторой линии, то для построения линии по этим точкам с помощью дуг окружностей необходимо задать в каждой из этих точек направление построения линии. Зададим его с помощью единичного вектора $\vec{k}_i = (p_i, q_i)$, компоненты которого p_i и q_i будем вычислять по формулам:

$$p_i = \frac{x_i - x_{i-1}}{\alpha} + \frac{x_{i+1} - x_i}{\beta}; \quad q_i = \frac{y_i - y_{i-1}}{\alpha} + \frac{y_{i+1} - y_i}{\beta};$$

$$p_i = \frac{P_i}{\sqrt{P_i^2 + Q_i^2}}; \quad q_i = \frac{Q_i}{\sqrt{P_i^2 + Q_i^2}},$$

где $\alpha = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}$, $\beta = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$;

вектор $\left(\frac{x_i - x_{i-1}}{\alpha}, \frac{y_i - y_{i-1}}{\alpha} \right)$ — единичный вектор, направ-

ленный из точки \vec{z}_{i-1} в точку \vec{z}_i ;

вектор $\left(\frac{x_{i+1} - x_i}{\beta}, \frac{y_{i+1} - y_i}{\beta} \right)$ — единичный вектор, направ-

ленный из точки \vec{z}_i в точку \vec{z}_{i+1} (рис.6).

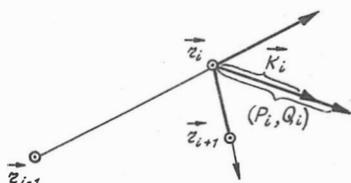


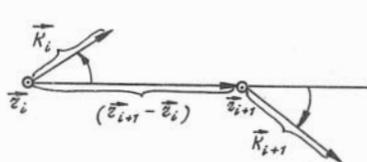
Рис. 6

Таким образом, направление построения линии в точке \vec{z}_i задает вектор \vec{K}_i , компоненты которого определяются из расположения двух соседних точек \vec{z}_{i-1} и \vec{z}_{i+1} .

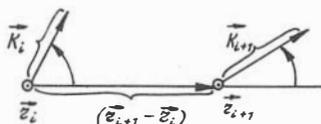
При построении линии между парой точек \vec{z}_i и \vec{z}_{i+1} , с заданными

в них \vec{K}_i и \vec{K}_{i+1} , будем рассматривать два случая взаимного расположения векторов \vec{K}_i, \vec{K}_{i+1} и $\vec{z}_{i+1} - \vec{z}_i$.
С л у ч а й 1. Угол между векторами $\vec{z}_{i+1} - \vec{z}_i$ и \vec{K}_i имеет знак, противоположный знаку угла между векторами

$$\vec{z}_{i+1} - \vec{z}_i \quad \text{и} \quad \vec{K}_{i+1}, \quad [\vec{K}_i, \vec{z}_{i+1} - \vec{z}_i] \cdot [\vec{K}_{i+1}, \vec{z}_{i+1} - \vec{z}_i] < 0.$$



Случай 1



Случай 2

Рис. 7

С л у ч а й 2. Углы имеют одинаковые знаки (рис. 7)

$$[\vec{K}_i, \vec{z}_{i+1} - \vec{z}_i] \cdot [\vec{K}_{i+1}, \vec{z}_{i+1} - \vec{z}_i] > 0.$$

В первом случае точки \vec{z}_i и \vec{z}_{i+1} соединятся двумя дугами так, что касательная прямая к первой дуге в точке \vec{z}_i параллельна $\vec{\kappa}_i$, касательная прямая ко второй дуге в точке \vec{z}_{i+1} параллельна $\vec{\kappa}_{i+1}$, и в точке стыковки дуг касательные обеих дуг совпадают и параллельны вектору $\vec{z}_{i+1} - \vec{z}_i$.

Для построения необходимо знать координаты точки стыковки и центров обеих дуг. Центр первой дуги лежит на прямой, проходящей через точку \vec{z}_i , перпендикулярно вектору $\vec{\kappa}_i$. (рис. 8). Если $\vec{\kappa}_i = (p_i, q_i)$, то

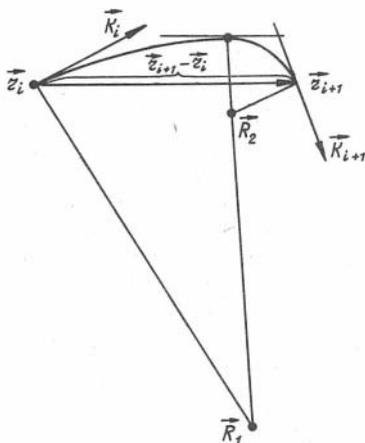


Рис. 8

вектор \vec{m}_i , перпендикулярный $\vec{\kappa}_i$, будет $\vec{m}_i = (q_i, -p_i)$ при повороте по часовой стрелке. Вектор $\vec{z}_{i+1} - \vec{z}_i$ повернем против часовой стрелки на 90° и нормируем его на единицу; результат обозначим через \vec{l}_i

$$S = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}, \quad \vec{l}_i = \left(-\frac{y_{i+1} - y_i}{S}, \frac{x_{i+1} - x_i}{S} \right).$$

Складывая два вектора \vec{m}_i и \vec{l}_i , получим вектор, направленный из точки \vec{z}_i в точку стыковки $\vec{z}_c = (x_c, y_c)$; для $\vec{m}_{i+1} = (q_{i+1} - p_{i+1})$ вектор $\vec{m}_{i+1} + \vec{l}_i$ будет направлен в точку стыковки из точки \vec{z}_{i+1} . Отсюда найдем координаты точки \vec{z}_c .

$$\vec{z}_c = \vec{z}_i + \alpha (\vec{m}_i + \vec{l}_i)$$

$$\vec{z}_c = \vec{z}_{i+1} + \beta (\vec{m}_{i+1} + \vec{l}_i),$$

где α и β некоторые постоянные. Чтобы найти их, решим систему

$$\alpha \left(q_i - \frac{y_{i+1} - y_i}{S} \right) - \beta \left(q_{i+1} - \frac{y_{i+1} - y_i}{S} \right) = x_{i+1} - x_i,$$

$$\alpha \left(-p_i + \frac{x_{i+1} - x_i}{S} \right) - \beta \left(-p_{i+1} + \frac{x_{i+1} - x_i}{S} \right) = y_{i+1} - y_i.$$

Определитель систем

$$\Delta = - \left(q_i - \frac{y_{i+1} - y_i}{S} \right) \cdot \left(-p_{i+1} + \frac{x_{i+1} - x_i}{S} \right) +$$

$$+ \left(q_{i+1} - \frac{y_{i+1} - y_i}{S} \right) \cdot \left(-p_i + \frac{x_{i+1} - x_i}{S} \right);$$

$$\alpha = \left[(x_{i+1} - x_i) \left(p_{i+1} - \frac{x_{i+1} - x_i}{S} \right) + \left(q_{i+1} - \frac{y_{i+1} - y_i}{S} \right) (y_{i+1} - y_i) \right] / \Delta;$$

$$\beta = \left[\left(q_i - \frac{y_{i+1} - y_i}{S} \right) (y_{i+1} - y_i) + \left(p_{i+1} - \frac{x_{i+1} - x_i}{S} \right) (x_{i+1} - x_i) \right] / \Delta.$$

Итак,

$$\vec{z}_c = \vec{z}_i + \alpha (\vec{m}_i + \vec{z}_i),$$

$$\vec{R}_1 = \vec{z}_i + \alpha \vec{m}_i,$$

$$\vec{R}_2 = \vec{z}_{i+1} + \beta \vec{m}_{i+1}.$$

Второй случай сводится к первому вводом промежуточной точки, в которой направление построения линии выбирается так, чтобы реализовался первый случай для обоих промежутков. Заметим, что эта точка является точкой перегиба. Координаты этой точки будем вычислять по формуле

$$\vec{z}_{\text{перегиба}} = (\vec{z}_{i+1} + \vec{z}_i) / 2,$$

а направление построения найдем по формулам:

$$P_{\text{перегиба}} = x_{i+1} - x_i - \frac{1}{4} \cdot S \cdot (p_{i+1} + p_i);$$

$$Q_{\text{перегиба}} = y_{i+1} - y_i - \frac{1}{4} \cdot S \cdot (q_{i+1} + q_i);$$

$$P_{\text{перегиба}} = P_{\text{перегиба}} / \sqrt{P_{\text{перегиба}}^2 + Q_{\text{перегиба}}^2}$$

$$Q_{\text{перегиба}} = Q_{\text{перегиба}} / \sqrt{P_{\text{перегиба}}^2 + Q_{\text{перегиба}}^2}.$$

Для линейно-интерполирующих графопостроителей каждую дугу необходимо представить точками, расположенными на ней так, что ломаная линия, проходящая через эти точки, отстояла от дуги не более, чем на ε , где ε — допустимая погрешность. Если заданы \vec{z}_0 — начало и \vec{z}_K — конец дуги, а также \vec{K} — ее центр и \vec{K}_0 — направление построения линии в точке \vec{z}_0 , то такие точки могут быть найдены следующим образом. Отступим от точки \vec{z}_0 к центру дуги на расстояние $\delta = 4\varepsilon - 2\varepsilon^2 / R$

(R - радиус дуги). Затем переместимся по направлению вектора $\vec{\kappa}_0$ на расстояние $s = \sqrt{2R\delta - \delta^2}$ и получим первую искомую точку \vec{z}_1 . Зададим в ней направление построения $\vec{\kappa}_1$ по формуле:

$$\vec{\kappa}_1 = \left(\vec{z}_1 + \frac{\vec{R} - \vec{z}_1}{R} \cdot \delta - \vec{z}_0 \right) / s.$$

Принимая найденную точку за начальную точку укоротившейся дуги и повторяя описанную выше процедуру до тех пор, пока не будет пройдена вся дуга, получим все необходимые для построения этой дуги точки.

Описанный метод интерполяции имеет следующий недостаток. При построении дуг окружностей малого радиуса, сравнимого с погрешностью ε , интерполирующие отрезки прямых образуют между собой заметные углы и линия в этом месте не выглядит гладкой. Поэтому, повидимому, целесообразнее в качестве погрешности задавать не абсолютное отклонение интерполирующего отрезка от дуги, а относительное - в единицах радиуса. Если задать теперь $\varepsilon = 0,07$, то угол между интерполирующими отрезками будет 164° , т.е. окружность будет интерполирована n -угольником, где $n \approx \frac{360^\circ}{16^\circ} \approx 22$. Впрочем, этот случай легко сводится к предыдущему. Вычислив радиус интерполируемой дуги, находим для нее абсолютное значение

$$\varepsilon \text{ и } \delta = 4\varepsilon R - 2\varepsilon^2 R^2 / R = 2\varepsilon R (2 - \varepsilon).$$

Для того, чтобы описанный метод применить, необходимо исключить случаи получения дуг бесконечно большого радиуса. Для этого перед вычислением координат центров дуг рассматривается векторное произведение

$$[\vec{\kappa}_i, \vec{z}_{i+1} - \vec{z}_i]$$

и, если оно мало, то точки \vec{z}_i и \vec{z}_{i+1} соединяются отрезком прямой, т.е. вводятся на графопостроитель координаты точек \vec{z}_i и \vec{z}_{i+1} , и нужно переходить к следующей

паре точек.

1. Описание операторной схемы программы "Контур - 2"

В состав операторной схемы входят следующие операторы:

- A₁ - оператор ввода.
- A₂ - нахождение первой и последней точек в линии.
- P₃ - проверка линии на замкнутость.
- A₄, A₉, A₁₅ - определение единичного вектора, указывающего наклон линии в точке.
- P₅ - проверка: не лежат ли предпоследняя, последняя и первая точка в замкнутой линии на одной прямой.
- K₆ - счетчик очередной точки.
- A₇ - определение вспомогательной точки, лежащей перед первой в случае незамкнутой кривой.
- A₈ - определение вспомогательной точки, лежащей после последней в случае незамкнутой кривой.
- P₁₀ - проверка: не лежат ли три точки на одной прямой.
- A₁₁, A₂₃ - загрузка точки в выводимый массив.
- P₁₂, P₂₄ - проверка: массив укомплектован или нет.
- A₁₃, A₂₅ - ввод массива.
- K₁₄ - счетчик очередной точки.
- P₁₆ - проверка: не лежат ли три точки близко к прямой линии.
- A₁₇ - сдвиг точки.
- P₁₈ - проверка положения точек на вариант 2.
- A₁₉ - вычисление промежуточных величин.
- A₂₀ - вычисление промежуточных величин.
- A₂₁ - вычисление радиусов окружностей и координат их центров.
- A₂₂ - вычисление точек через определенный шаг.
- P₂₆ - сравнение расстояния между точками с заданной погрешностью.
- K₂₇ - выбор новой пары точек.
- P₂₈ - проверка: не последняя ли точка в последней линии.

P_{29} - проверка: не последняя ли точка в линии.

A_{31} - конец. A_{30} - оператор вывода.

Структура операторной схемы имеет вид:

A_1 $^{1,29}A_2$ P_{3+7} 6A_4 P_{5+9} K_6^4 3A_7 A_8 $^{5,6,74}A_9$ P_{10+15} A_{11} P_{12+14} A_{13} $^{12}K_{14}^9$
 $^{10}A_{15}$ $^{15,29}P_{16+18}$ A_{17} $^{16,17}P_{18+20}$ A_{19}^{21} $^{18}A_{20}$ $^{19,20}A_{21}$ $^{26}A_{22}$ A_{23} P_{24+26} A_{25}
 $^{24,25}P_{26+22}$ K_{27} P_{28}^{130} P_{29+16}^{12} $^{29}A_{30}$ A_{31} .

Операторная схема работает следующим образом. Оператор A_1 вводит информацию и передает управление оператору A_2 , который находит номера первой и последней точек в линии. Затем работает оператор P_3 , проверяющий линию на замкнутость. Если линия замкнутая, то управление передается оператору A_4 , который определяет единичный вектор, указывающий наклон линии в точке. Если линия незамкнутая, то после P_3 работает оператор A_7 , определяющий вспомогательную точку, лежащую перед первой. После оператора A_4 работает оператор P_5 , проверяющий, не лежат ли предпоследняя, последняя и первая точки в замкнутой линии на одной прямой. В случае, когда это условие выполняется, от A_5 управление передается оператору K_6 , увеличивающему номера точек на единицу, после чего снова работает оператор A_4 .

Если же три указанных точки не лежат на одной прямой, то оператор P_5 передает управление оператору A_9 . После оператора A_7 работает оператор A_8 , определяющий вспомогательную точку, лежащую после последней в случае незамкнутой кривой. Вспомогательные точки нужны для определения производной в первой и последней точках незамкнутой кривой. Далее работает оператор A_9 , в который существует также переход от операторов P_5 и K_{14} . Оператор A_9 определяет единичный вектор, указывающий наклон линии в точке и передает управление оператору P_{10} , который проверяет, не лежат ли три точки на одной прямой. Если так, то работает оператор A_{11} , задающий точку в выводимый массив, а затем оператор P_{12} , проверяющий, укомплектован

массив или нет. Если массив укомплектован, то работает оператор ввода A_{13} , который производит ввод массива на печать. После оператора P_{12} , если массив не укомплектован, а также после A_{13} , работает оператор K_{14} , увеличивающий номер очередной точки на 1, т.е. присваивающий $i := i + 1$. От оператора K_{14} управление передается на оператор A_9 и повторяется описанный цикл, в котором работают операторы $A_9 \rightarrow A_{19}$.

Если три точки не лежат на одной прямой (проверка оператором P_{10}), то оператор P_{10} передает управление на оператор A_{15} , определяющий единичный вектор, указывающий наклон в точке, после чего оператор P_{16} проверяет, не лежат ли три точки близко к одной прямой. Если лежат, то работает оператор A_{17} , который осуществляет сдвиг точек, а также после оператора P_{16} , если точки не лежат близко к прямой линии, управление передается оператору P_{18} . Этот оператор проверяет положение точек на вариант 2 (см. описание). Если точки расположены по этому варианту, то управление передается оператору A_{19} , в противном случае — оператору A_{20} . Оба этих оператора вычисляют промежуточные величины для последующих процедур. После любого из операторов A_{19} и A_{20} работает оператор A_{21} , который вычисляет радиусы окружностей и координаты их центров, а затем оператор A_{22} , вычисляющий координаты точек, расположенных через определенный шаг.

Оператор A_{22} передает управление оператору A_{23} , который записывает точки в выводимый массив, после чего оператор P_{24} проверяет, заполнен ли массив. Если массив заполнен, работает оператор вывода массива A_{25} , если нет, то работает оператор P_{26} , который сравнивает расстояние между точками с заданной величиной погрешности. Если расстояние между точками не превышает погрешности, то управление передается на оператор K_{27} , который выбирает следующие две точки, т.е. присваивает i значение $i + 1$. Если же расстояние между точками входит за пределы погрешности, то после оператора P_{26} управление перейдет к оператору A_{22} , который будет вычислять точки с меньшим шагом. Цикл между A_{22} и P_{26} работает, пока шаг между

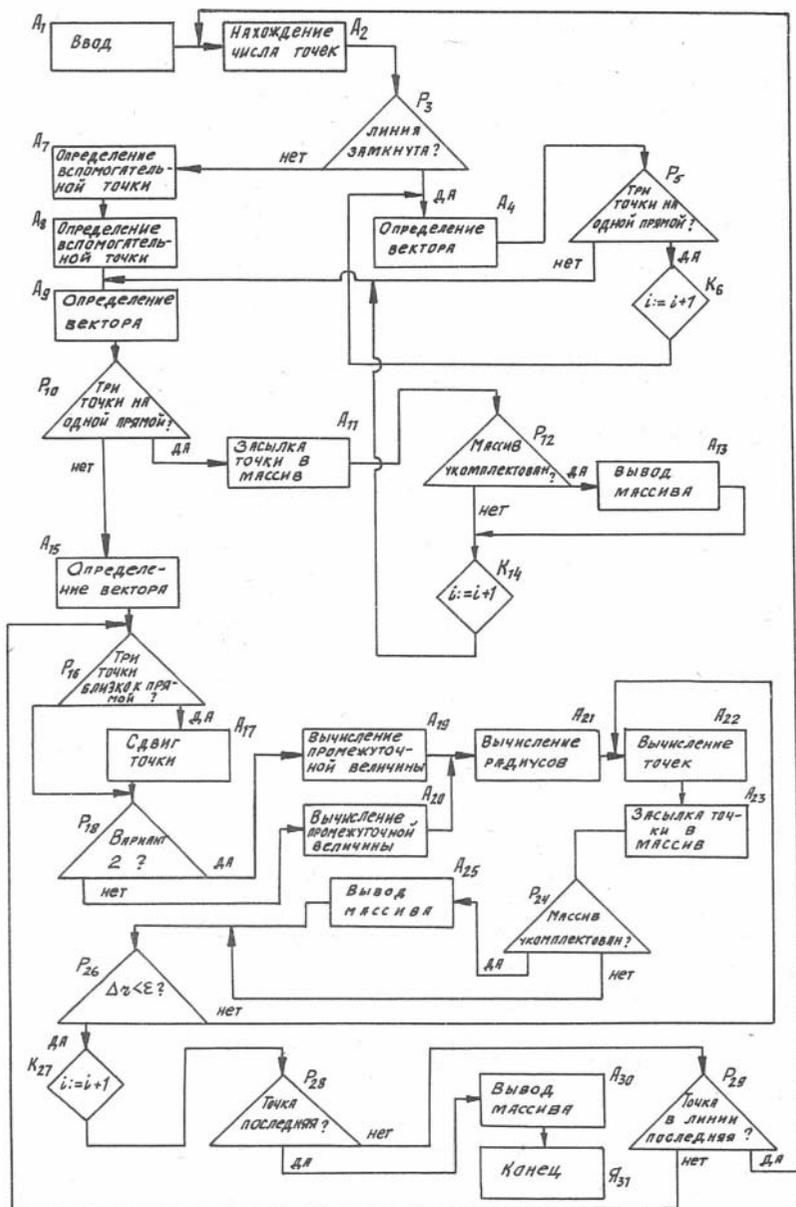


Рис. 9. Блок-схема программы

точками не удовлетворит заданной величине погрешности. После оператора K_{27} работает оператор P_{28} , проверяющий, не является ли точка i последней точкой в последней линии. Если нет, то оператор P_{29} проверяет, не последняя ли эта точка в линии. В случае, если точка последняя, управление передается на оператор A_2 , который отыскивает номера первой и последней точек новой линии. Если точка в линии не последняя, то от оператора P_{29} управление передается оператору P_{16} . Когда точка окажется последней в последней линии, оператор P_{28} передает управление A_{30} , который выводит не полностью укомплектованный массив, на чем работа заканчивается. Я31 - конец.

2. Инструкция для работ с программой "Контур - 2"

Программа строит линию по заданным точкам с помощью окружностей. На печать выводятся координаты таких точек линии, что при последовательном соединении их отрезками прямых полученная ломаная линия не будет иметь углов, меньших, чем $\pi - \sqrt{4,5\varepsilon}$, где ε - параметр, который выбирается, исходя из требований необходимой точности построения линии. Перфокарта со значением ε и ее контрольной суммой $k\Sigma$ ставится сразу после программы. Затем вводятся координаты точек, по которым программа строит линию. Причем сначала вводится массив координат x , который оформлен следующим образом: перед началом массива пробивается число 0, в конце массива число 10^7 и контрольная сумма всего массива, затем вводится массив значений координат y , оформленный также, как и предыдущий. Если необходимо построить несколько линий, то массивы значений оформляются как и для одной линии: пробиваются подряд значения координат x для всех линий, только в конце каждой линии пробивается число 10^6 , а в конце последней линии 10^7 , а затем контрольная сумма всего массива, нуль пробивается только в начале первой линии.

Выдаются на печать массивы по 100 элементов в каждом. Если пронумеровать выводимые массивы и расположить их попарно $(2n-1, 2n)$, $n = 1, 2, \dots$, то координаты точек по оси x находятся в первом массиве каждой пары, а координаты точек по оси y — во втором массиве каждой пары. Начало выводимых координат отмечается числом $--+19922337204$, этим же числом разделяются в массивах координаты одной линии от другой. Конец выводимых координат точек всех линий отмечается числом $+-00499999999$.

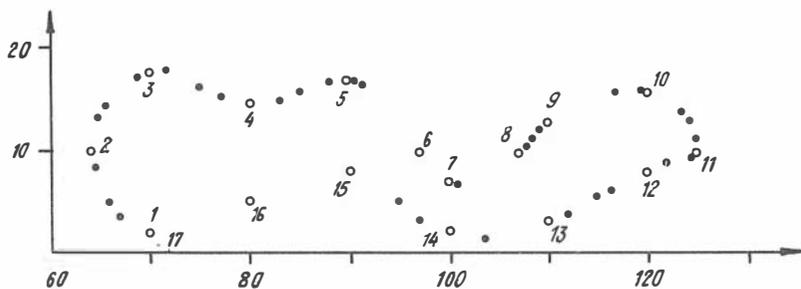


Рис. 10

Контрольный пример.

График контурной незамкнутой линии.

ТАБЛИЦА ВВОДА В ЭВМ (КОНТРОЛЬНЫЙ ПРИМЕР, СМ СТ. 19)

№ перфо- карты	порядок	МАНТИССА	№ перфо- карты	порядок	МАНТИССА	№ перфо- карты	порядок	МАНТИССА
1.	00	1						
2.	КС							
3.	00	0	5.	02	97	8.	01	8
	02	7		03	1		01	3
	02	64		03	107		01	2
	02	7		03	11		01	8
	02	8		03	12		01	5
	02	9		03	125		01	2
	02	97		03	12		07	1
	03	1		03	11		01	2
	03	107		03	1		02	1
	03	11		02	9		02	18
	03	12		02	8		02	15
	03	125		08	1		02	17
			6.	КС				
4.	03	12	7.	00	0	9.	02	1
	03	11		01	2		01	7
	03	1		02	1		02	1
	02	9		02	18		02	13
	02	8		02	15		02	16
	02	7		02	17		02	1
	07	1		02	1		01	8
	02	7		01	7		01	3
	02	64		02	10		01	2
	02	7		02	13		01	8
	02	8		02	16		01	5
	02	9		02	1		08	1
			10.	КС				

Выдача на печать. Тестовый

--0-----	---19922337204	+++02700000000	+++02665672727
+++02656181626	+++02668092243	+++02700000000	+++02715401894
+++02829122603	+++02850000000	+++02879406477	+++02899282462
+++02907265043	+++02906978891	+++02905898012	+++02904260557
+++02899294217	+++02900254823	+++02970000000	+++03100000000
+++03107859579	+++03108500000	+++03109228559	+++03110000000
+++03124279382	+++03125054817	+++03125129186	+++03125000000
+++03122210606	+++03120000000	+++03116833994	+++03115000000
+++02972384060	+++02960000000	+++02924609650	+++02898015475
+++02700000000	+++02665672727	+++02656666667	+++02640420455
+++02700000000	+++02715401894	+++02719805259	+++02750000000
+++02879406477	+++02899282462	+++02901914184	+++02903743416
+++02905898012	+++02904260537	+++02902406596	+++02900707928
+++01386928766	+++01500000000	+++01855505046	+++02100000000
+++02181593109	+++02180262062	+++02165000000	+++02155246595
+++02170648544	+++02169420431	+++02169275342	+++02169910561
+++02177130286	+++02177198905	+++02176464674	+++02175081181
+++01700000000	+++01674333356	+++01667970071	+++01678516604
+++02130000000	+++02160013445	+++02162001198	+++02160000000
+++02100000000	+++01964886098	+++01940604267	+++01942843317
+++01550000000	+++01376852803	+++01300000000	+++01177824007
+++01802483202	+++01804191894	+++01500000000	+++01200000000
+++01855505046	+++02100000000	+++02137078761	+++02148544875
+++02165000000	+++02155246595	+++02150000000	+++02152193644
+++02169275342	+++02169910561	+++02171219535	+++02172956211
+++02176464674	+++02175081181	--0-----	+++02898986891
+++03100999952	+++03100867495	+++03101115543	+++03107000000
+++03116813191	+++03119402855	+++03120000000	+++03123654846
+++03124824878	+++03124520202	+++03124499674	+++03122500000
+++03111965955	+++03110000000	+++03103669895	+++03100000000
+++02893478155	+++02800000000	+++00499999999	+++03108500000
+++03120000000	+++03123654846	+++03124279382	+++03125054817
+++03124499674	+++03122500000	+++03122210606	+++03120000000
+++03103669895	+++03100000000	+++02972384060	+++02950000000
+++02700000000	---19922337204	+++02700000000	+++02665672727
+++02656181626	+++02668092243	+++02700000000	+++02715401894
+++02829122603	+++02850000000	+++02879406477	+++02899282462
+++02907265043	+++02906975891	+++02905898012	+++02904260537
+++02173316430	+++02171510912	+++02170425398	+++02100000000
+++02100000000	+++02107098342	+++02115000000	+++02123989133
+++02139692811	+++02131706515	+++02116807736	+++02100147899
+++01900000000	+++01893799692	+++01800000000	+++01626125144
+++01200000000	+++01310428872	+++01500000000	+++01715031469
+++02115000000	+++02123989133	+++02130000000	+++02160013445
+++02116807738	+++02100147599	+++02100000000	+++01964886098
+++01800000000	+++01626125144	+++01500000000	+++01376852803
+++01500000000	+++01715031469	+++01802483202	+++01804191894
+++01386928766	+++01500000000	+++01855505046	+++02100000000
+++02181593109	+++02180262062	+++02165000000	+++02155246595
+++02170648544	+++02169420431	+++02169275342	+++02169910561
+++02177130286	+++02177198905	+++02176464674	+++02175081181

ПРИМЕР. ПРОГРАММА „КОНТУР -2”

++02656666667	++02640420455	++02640000000	++02648090813
++02719805259	++02750000000	++02769296313	++02800000000
++02901914184	++02903743116	++02905451050	++02906713052
++02902406596	++02900707928	++02899491187	++02898988891
++03100399952	++03100867495	++03101115543	++03107000000
++03116813191	++03119402855	++03120000000	++03123654846
++03124824878	++03124520202	++03124499674	++03122500000
++03111965955	++03110000000	++03103669895	++03100000000
++02899964921	++02800000000	++02700000000	--19922337204
++02640000000	++02648090813	++02656181626	++02688092243
++02769296313	++02800000000	++02829122603	++02850000000
++02905451050	++02906713052	++02907265043	++02906975891
++02899491187	--0-----	--19922337204	++01200000000
++02137076761	++02148544875	++02176108936	++02180000000
++02150000000	++02152193644	++02160000000	++02170995500
++02171219535	++02172956211	++02174765489	++02176262173
++02173316430	++02171510912	++02170425398	++02100000000
++02100000000	++02107098342	++02115000000	++02123989133
++02139692311	++02131706515	++02116807738	++02100147599
++01900000000	++01893799692	++01800000000	++01626125144
++01200000000	++01310428872	++01500000000	++01715031469
--19922337204	++01200000000	++01386926766	++01500000000
++02176108936	++02180000000	++02181593109	++02180262062
++02160000000	++02170995500	++02170648544	++02169420431
++02174765489	++02176262173	++02177130286	++02177198305
++02899294217	++02900254823	++02970000000	++03100000000
++03107859579	++03108500000	++03109228559	++03110000000
++03124279382	++03125054817	++03125129186	++03125000000
++03122210606	++03120000000	++03116333934	++03115000000
++02972384060	++02950000000	++02924609650	++02900000000
++03109228559	++03110000000	++03116813191	++03119402855
++03125129186	++03125000000	++03124824878	++03124520202
++03116333934	++03115000000	++03111965955	++03110000000
++02924609650	++02898015475	++02899964921	++02800000000
++02656666667	++02640420455	++02640000000	++02648090813
++02719805259	++02750000000	++02769296313	++02800000000
++02901914184	++02903743116	++02905451050	++02906713052
++02902406596	++02900707928	++02899491187	--0-----
++01700000000	++01674333356	++01667970071	++01678516804
++02130000000	++02160013445	++02162001198	++02160000000
++02100000000	++01964886098	++01940604267	++01942843317
++01550000000	++01376852803	++01300000000	++01177824007
++01800000000	++01794542374	++01500000000	++00499999999
++02162001198	++02160000000	++02139692811	++02131706515
++01940604267	++01942843317	++01900000000	++01893799692
++01300000000	++01177824007	++01200000000	++010310428872
++01500000000	++01020000000	--19922337204	++01200000000
++02137076761	++02149541875	++02176108936	++02180000000
++02150000000	++02152193644	++02160000000	++02170995500
++02171219535	++02172956211	++02174765489	++02176262173
--0-----			

ТАБЛИЦА РАСПРЕДЕЛЕНИЯ ПАМЯТИ

	НАЧАЛО	КОНЕЦ	ДЛИНА	ПРИМЕЧАНИЕ
ПРОГРАММА	0020	1064	1045	
ЧИСЛОВНЕ КОНСТАНТЫ	1065	1112	0026	
ФОРМИРУЕМНЕ КОНСТАНТЫ	1113	1131	0016	
СКАЛЯРН	1132	1200	0047	
РАБОЧИЕ ЯЧЕЙКИ	1201	1212	0012	
МАССИВН				
1	1213	1216	0004	
2	1217	1222	0004	
3	1223	1227	0005	
4	1230	1234	0005	
5	1235	1241	0005	
6	1242	1246	0005	
7	1247	1412	0144	
10	1413	1556	0144	
11	1557	2542	0764	
12	2543	3526	0764	
РАБОЧЕЕ ПОЛЕ ИС	НАЧАЛО	КОНЕЦ	ДЛИНА	
	3527	7500	3751	

АДРЕСА ПРОГРАММНЫХ ОСТАНОВОВ

1064

3. ПРОГРАММА "КОНТУР - 2" (на Альфа - яэнью)

Лист 1

1 начало массив $x, y [1:500], w, v [1:100], a, b, c, d [1:5], f, e [1:4];$
 2 вещественные $\varepsilon, \delta, S, p, q, r_1, q_1, \gamma_1, \gamma_2, r_2, q_2, x_1, y_1, x_2, y_2, z_1, z_2, \alpha, \beta,$
 3 $R_1, R_2, \lambda_1, \lambda_2, e, \lambda;$ целое $i, j, k, l, m, n, w;$ логические $L_1, L_2;$
 4 процедура ПТ ($x_1, y_1, x_2, y_2, r_1, q_1, r_2, q_2, p, q, z_1, z_2, R_1, R_2$);
 5 начало вещество $\alpha, \beta, \Delta, t, \tau, S, l, d, l_1, d_1;$
 6 $S := \text{sqrt}((x_2 - x_1)^2 + (y_2 - y_1)^2); t := (y_1 - y_2) / S; \tau := (x_2 - x_1) / S;$
 7 $l := t + q_1; d_1 := \tau - p_2; d := \tau - p_1; l_1 := t + q_2; \Delta := d \times l_1 -$
 8 $l \times d_1; \alpha := ((y_2 - y_1) \times l_1 - (x_2 - x_1) \times d_1) / \Delta; \beta := ((y_2 - y_1) \times l$
 9 $- (x_2 - x_1) \times d) / \Delta; p := x_1 + \alpha \times l; q := y_1 + \alpha \times d; z_1 := x_1 + \alpha \times q_1;$
 10 $z_2 := y_1 - \alpha \times r_1; R_1 := x_2 + \beta \times q_2; R_2 := y_2 - \beta \times r_2; \text{конец};$

Лист 2

11 процедура ТП ($x_1, y_1, x_2, y_2, z_1, z_2, r_1, q_1, r_2, q_2, p, q$);
 12 начало вещество $S;$ $S := \text{sqrt}((x_2 - x_1)^2 + (y_2 - y_1)^2);$
 13 $z_1 := (x_1 + x_2) / 2; z_2 := (y_1 + y_2) / 2; p := x_2 - x_1 - .25 \times S \times (r_2 + r_1);$
 14 $q := y_2 - y_1 - .25 \times S \times (q_2 + q_1); S := \text{sqrt}(p^2 + q^2); p := p / S; q := q / S; \text{конец};$
 15 процедура ТЧК ($x_1, y_1, z_1, z_2, \varepsilon, p, q, r_1, q_1, S$);
 16 начало вещество $\delta, R, x_2, y_2;$ $R := \text{sqrt}((x_1 - z_1)^2 + (y_1 - z_2)^2);$
 17 $\delta := 2 \times \varepsilon \times R \times (2 - \varepsilon); S := \text{sqrt}(2 \times R \times \delta - \delta^2); p := x_1; q := y_1; x_2 := p \times$
 18 $(1 - \delta / R) + z_1 \times \delta / R; y_2 := q \times (1 - \delta / R) + z_2 \times \delta / R; x_1 := x_2 + r_1 \times S;$
 19 $y_1 := y_2 + q_1 \times S; x_2 := x_1 \times (1 - \delta / R) + z_1 \times \delta / R; y_2 := y_1 \times (1 - \delta / R) +$
 20 $z_2 \times \delta / R; p_1 := (x_2 - p) / S; q_1 := (y_2 - q) / S; \text{конец};$

21 ввод (ε, x, y); $m := n := 1$; $\alpha_1: КОД(177, 7777, 7777, 7777)$;
 22 $\alpha_2: КОД(077, 7777, 7777, 7777)$; $L: КОД(0, \alpha_1, 0, \omega[m])$; $КОД(0, \alpha_1,$
 23 $0, v[m])$; $m := m + 1$; если $m > 100$ то {ввод (ω, v); $m := 1$ };
 24 $L_1 := L_2 := ложь$; для $i := n, \dots, 500$ цикл { $l := i$; если $авс(10^6 - x[l])$
 25 < 1 то на M_1 ; если $авс(10^7 - x[l]) < 1$ то на M_2 };
 26 $M_2: 2 := истина$; на $M_3, M_1: L_1 := истина$; $M_3: если$
 27 $sqrt((x[n+1] - x[l-1])^2 + (y[n+1] + y[l-1])^2) < 10^{-6}$ то { $x[n] :=$
 28 $x[l-2]$; $y[n] := y[l-2]$; $x[l] := x[n+2]$; $y[l] := y[n+2]$; $M_6:$
 29 $x_1 := x[l-1] - x[l-2]$; $y_1 := y[l-1] - y[l-2]$; $x_2 := x[l-2] - x[l-3]$;
 30 $y_2 := y[l-2] - y[l-3]$; $p := x_1/sqrt(x_1^2 + y_1^2) + x_2/sqrt(x_2^2 + y_2^2)$;

31 $q := y_1/sqrt(x_1^2 + y_1^2) + y_2/sqrt(x_2^2 + y_2^2)$; $S := sqrt(p^2 + q^2)$;
 32 $p := p/S$; $q := q/S$; если $авс(y_1 \times p - x_1 \times q) < \varepsilon \times sqrt(x_1^2 + y_1^2)$
 33 то { $e := x[n]$; $л := y[n]$; для $i := n, \dots, l-1$ цикл { $x[i] := x[i+1]$;
 34 $y[i] := y[i+1]$ }; $x[l] := 2$; $y[l] := л$; на M_6 }} иначе
 35 { $x_2 := x[n+3] - x[n+2]$; $y_2 := y[n+3] - y[n+2]$; $x_1 := x[n+2] -$
 36 $x[n+1]$; $y_1 := y[n+2] - y[n+1]$; $\delta := (x_1 \times x_2 + y_1 \times y_2) / (x_1^2 + y_1^2)$;
 37 $x[n] := x[n+3] - x_1 \times (1 + 2 \times \delta)$; $y[n] := y[n+3] - y_1 \times (1 + 2 \times \delta)$;
 38 $x_1 := x[l-2] - x[l-1]$; $y_1 := y[l-2] - y[l-1]$; $x_2 := x[l-3] - x[l-2]$;
 39 $y_2 := y[l-3] - y[l-2]$; $\delta := (x_1 \times x_2 + y_1 \times y_2) / sqrt(x_1^2 + y_1^2)$;
 40 $x[l] := x[l-3] - x_1 \times (1 + 2 \times \delta)$; $y[l] := y[l-3] - y_1 \times (1 + 2 \times \delta)$ };

41 $j := 1$; для $i := n, \dots, l$ цикл $\{x[j] := x[i]; y[j] := y[i]; j := j + 1\}$;
 42 $k := l - n$; $n := l$; $x_1 := x[2] - x[1]$; $y_1 := y[2] - y[1]$; $x_2 := x[3]$
 43 $- x[2]$; $y_2 := y[3] - y[2]$; $p_1 := x_1 / \text{sqrt}(x_1^2 + y_1^2) + x_2 / \text{sqrt}$
 44 $(x_2^2 + y_2^2)$; $q_1 := y_1 / \text{sqrt}(x_1^2 + y_1^2) + y_2 / \text{sqrt}(x_2^2 + y_2^2)$;
 45 $S := \text{sqrt}(p_1^2 + q_1^2)$; $p_1 := p_1 / S$; $q_1 := q_1 / S$;
 46 для $i := 3, \dots, k$ цикл $\{\text{если } abs(p_1 \times y_2 - q_1 \times x_2) < \epsilon / \text{sqrt}(x_2^2 + y_2^2)$
 47 то $\{w[m] := x[i-1]$; $v[m] := y[i-1]$; $m := m + 1$; если $m > 100$
 48 то $\{\text{сход}(w, v)$; $m := 1\}$; $x_2 := x[i+1] - x[i]$; $y_2 := y[i+1] - y[i]$;
 49 на МЧ}; $x_1 := x[i+1] - x[i]$; $y_1 := y[i+1] - y[i]$; $p_2 := x_1 / \text{sqrt}$
 50 $(x_1^2 + y_1^2) + x_2 / \text{sqrt}(x_2^2 + y_2^2)$; $q_2 := y_1 / \text{sqrt}(x_1^2 + y_1^2)$

51 $+ y_2 / \text{sqrt}(x_2^2 + y_2^2)$; $S := \text{sqrt}(p_2^2 + q_2^2)$; $p_2 := p_2 / S$;
 52 $q_2 := q_2 / S$; если $abs(p_2 \times y_2 - q_2 \times x_2) < \epsilon / \text{sqrt}(x_2^2 + y_2^2)$ то
 53 $\{x[i-1] := x[i-1] + 2 \times \epsilon \times p_1$; $y[i-1] := y[i-1] + 2 \times \epsilon \times q_1\}$; если $(p_2$
 54 $\times y_2 - q_2 \times x_2) \times (p_1 \times y_2 - q_1 \times x_2) > 0$ то $\{\theta[5] := y[i]$;
 55 $a[5] := x[i]$; $w := 4$; $c[5] := p_2$; $d[5] := q_2$; $x_1 := x[i-1]$;
 56 $R_1 := y[i-1]$; $x_2 := a[5]$; $R_2 := \theta[5]$; $\Pi(x_1, R_1, x_2, R_2,$
 57 $e_1, e_2, p_1, q_1, p_2, q_2, p, q)$; $a[3] := e_1$; $\theta[3] := e_2$; $c[3] := p$;
 58 $d[3] := q$ иначе $\{a[3] := a[4] := a[5] := x[i]$; $\theta[3] := \theta[4]$
 59 $:= \theta[5] := y[i]$; $c[3] := c[5] := p_2$; $d[3] := d[5] := q_2$; $w := 2\}$;
 60 $a[1] := x[i-1]$; $\theta[1] := y[i-1]$; $c[1] := p_1$; $d[1] := q_1$; $j := 1$

61 w /2 раз цикл { $x_1 := a[j]$; $y_1 := \sigma[j]$; $x_2 := a[j+2]$; y_2
62 $:= \sigma[j+2]$; $p_1 := c[j]$; $q_1 := d[j]$; $p_2 := c[j+2]$; $q_2 := d[j+2]$;
63 $\Pi T(x_1, y_1, x_2, y_2, p_1, q_1, p_2, q_2, \beta q, z_1, z_2, R_1, R_2)$; $a[j+1] := p$;
64 $\sigma[j+1] := q$; $f[j] := z_1$; $z[j] := z_2$; $f[j+1] := R_1$; $z[j+1] := R_2$;
65 $j := j+2$ }; $c[2j] := c[4j] := x_2 / \text{sqrt}(x_2^2 + y_2^2)$; $d[2j] :=$
66 $d[4j] := y_2 / \text{sqrt}(x_2^2 + y_2^2)$; для $j := 1, \dots, w$ цикл {
67 $x_1 := a[j]$; $y_1 := \sigma[j]$; $z_1 := f[j]$; $z_2 := z[j]$; $p_1 := c[j]$;
68 $q_1 := d[j]$; $M_5 : \text{ТЧК}(x_1, y_1, z_1, z_2, \epsilon, p, q, p_1, q_1, S)$; $w[m] :=$
69 p ; $v[m] := q$; $m := m+1$; если $m > 100$ то { вывод (w, v);
70 $m := 1$ }; если $\text{sqrt}((a[j+1]-p)^2 + (\sigma[j+1]-q)^2) > S$ то на M_5 };

71 $p_1 := p_2$; $q_1 := q_2$; $x_2 := x_1$; $y_2 := y_1$; M_4 };
72 $w[m] := a[k]$; $v[m] := u[k]$; $m := m+1$; если $m > 100$ то { вывод (w, v);
73 $m := 1$ }; если L_1 то на L ; если L_2 то $KOD(0, \alpha_2, 0,$
74 $w[m])$; $KOD(0, \alpha_2, 0, v[m])$; вывод (w, v); конец *

§3. МЕТОД ПОСТРОЕНИЯ КОНТУРНОЙ ЛИНИИ С ПРИМЕНЕНИЕМ ПРЯМЫХ И ОКРУЖНОСТЕЙ

1. На плоскости (x, y) дана упорядоченная система точек с координатами (x_i, y_i) , $i=1, \dots, N$, причем, не исключена возможность, что точка (x, y) совпадает с точкой (x_N, y_N) . Если соединим каждую две соседние точки, то получим ломаную линию (замкнутую, если $(x_1, y_1) = (x_N, y_N)$ и незамкнутую, если $(x_1, y_1) \neq (x_N, y_N)$). Требуется аппроксимировать ломаную линию некоторой гладкой кривой, состоящей из отрезков прямых и сопряженных с ними и между собой окружностей. Сопряжение окружностей должно быть в точках, где не существует производная. Ниже следует алгоритм построения такой кривой, причем на участках, где функция линейна, будут найдены только начало и конец линейности, на участках же нелинейности функции будут найдены ее значения с шагом h .

2. Назовем внутренней точкой точку, лежащую на одной прямой с двумя соседними точками. Одним из первых шагов алгоритма является выбрасывание внутренних точек. Оставшиеся точки перенумеровываются в том же порядке. Число N уменьшается на столько, сколько было внутренних точек. Заметим, что понятие внутренней точки относится только к первоначальным точкам x_1, x_2, \dots, x_N . Если четверть расстояния между двумя точками (x_i, y_i) и (x_{i+1}, y_{i+1}) меньше чем δ_i , то заменяем старшие δ_i и δ_{i+1} на новые, равные четверти вышеупомянутого расстояния. $\delta_i, i=1, \dots, N$ — некоторые числа, большие нуля, от которых зависит точность аппроксимации ломаной линии гладкой. Подробнее о δ_i будет говориться ниже. Проведем прямую AB_{i+1} (см. рис. 11)

через точки (x_i, y_i) и (x_{i+2}, y_{i+2}) :

$$y = y_i + \frac{y_{i+2} - y_i}{x_{i+2} - x_i} (x - x_i), \quad i = 1, \dots, N-2. \quad (1)$$

Если $x_{i+2} = x_i$, то уравнение (1) эквивалентно уравнению вида $x = x_i$. Через точку (x_{i+1}, y_{i+1}) , $i = 1, 2, \dots, N-2$ проводим прямую $C\mathcal{D}_{i+1}$, угловой коэффициент которой равен угловому коэффициенту прямой AB_{i+1} . Уравнение прямой $C\mathcal{D}_{i+1}$ имеет вид:

$$y = y_{i+1} + \frac{y_{i+2} - y_i}{x_{i+2} - x_i} (x - x_{i+1}) \quad \text{ПРИ } x_i \neq x_{i+2}, \quad (2)$$

$$x = x_{i+1} \quad \text{ПРИ } x_i = x_{i+2}.$$

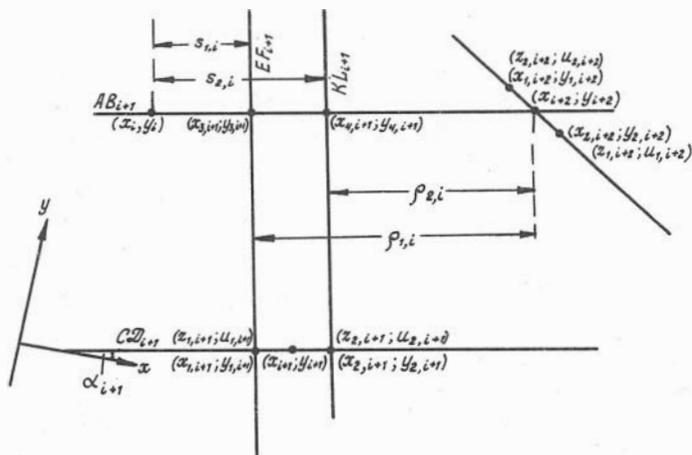


Рис. 11

Если ломаная линия замкнута, то дополнительная прямая AB_N имеет вид:

$$y = y_{N-1} + \frac{y_2 - y_{N-1}}{x_2 - x_{N-1}} (x - x_{N-1}) \quad \begin{array}{l} \text{ПРИ } x_2 \neq x_{N-1}, \\ x = x_{N-1} \quad \text{ПРИ } x_2 = x_{N-1} \end{array} \quad (1')$$

и, соответственно, дополнительная прямая $C\mathcal{D}_N$ имеет вид:

$$y = y_N + \frac{y_2 - y_{N-1}}{x_2 - x_{N-1}} (x - x_N) \quad \begin{array}{l} \text{ПРИ } x_2 \neq x_{N-1}, \\ x = x_N \quad \text{ПРИ } x_2 = x_{N-1}. \end{array} \quad (2')$$

Нетрудно видеть, что формулы (1') и (2') получаются из формул (1) и (2), если в (1) и (2) положить $i = N-1$ и точки с индексом $N+1$ заменить на точки с индексом 2. В дальнейшем будем считать, что если ломаная линия незамкнута, то $i = 1, \dots, N-2$; если ломаная линия замкнута, то $i = 1, \dots, N-1$, но точкам с индексами $i, i+1, i+2$ при $i = N-1$, будем ставить в соответствие точки с индексами $N-1, N, 2$. Заметим, что точку с индексом N , в случае замкнутой ломаной линии, можем заменить точкой с индексом 1. С учетом этого, все ниже изложенные результаты будут оставаться в силе, как для замкнутых, так и незамкнутых ломаных. В некоторых особых случаях будем указывать предел изменения индекса i . Найдем точки $(x_{1,i+1}; y_{1,i+1})$ и $(x_{2,i+1}; y_{2,i+1})$, лежащие на прямых $C\mathcal{D}_{i+1}$ на расстоянии δ_{i+1} от точки (x_{i+1}, y_{i+1}) в одну и другую сторонн:

$$\begin{aligned} x_{1,i+1} &= x_{i+1} - \delta_{i+1} \cos \alpha_{i+1}, & y_{1,i+1} &= y_{i+1} - \delta_{i+1} \sin \alpha_{i+1}, \\ x_{2,i+1} &= x_{i+1} + \delta_{i+1} \cos \alpha_{i+1}, & y_{2,i+1} &= y_{i+1} + \delta_{i+1} \sin \alpha_{i+1}. \end{aligned} \quad (3)$$

Угол α_{i+1} есть угол, образованный прямой $C\mathcal{D}_{i+1}$ с осью x :

$$\alpha_{i+1} \begin{cases} \arctg \frac{y_{i+2} - y_i}{x_{i+2} - x_i} & \text{ПРИ } x_i \neq x_{i+2}, \\ \pi/2 & \text{ПРИ } x_i = x_{i+2}. \end{cases} \quad (4)$$

Перпендикулярно прямой AB_{i+1} проведем через точку $(x_{1,i+1}; y_{1,i+1})$ прямую EF_{i+1}

$$y = y_{1,i+1} - \frac{x_{i+2} - x_i}{y_{i+2} - y_i} (x - x_{1,i+1}) \quad \text{при } y_{i+2} \neq y_i, \quad (5)$$

$$x = x_{1,i+1} \quad \text{при } y_{i+2} = y_i$$

и через точку $(x_{2,i+1}; y_{2,i+1})$ проведем прямую KL_{i+1} , перпендикулярную прямой AB_{i+1} . Уравнение прямой KL_{i+1} имеет вид:

$$y = y_{2,i+1} - \frac{x_{i+2} - x_i}{y_{i+2} - y_i} (x - x_{2,i+1}) \quad \text{при } y_i \neq y_{i+2}, \quad (6)$$

$$x = x_{2,i+1} \quad \text{при } y_i = y_{i+2}.$$

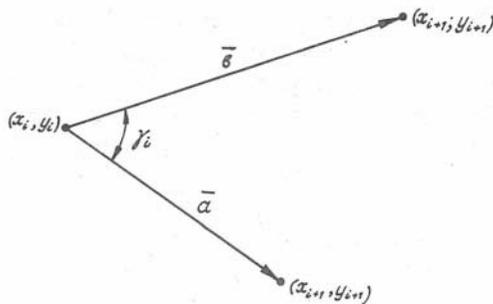


Рис.12

Точки пересечения прямой EF_{i+1} с AB_{i+1} и прямой KL_{i+1} с AB_{i+1} обозначим, соответственно, через $(x_{3,i+1}; y_{3,i+1})$ и $(x_{4,i+1}; y_{4,i+1})$. Решив соответствующие уравнения прямых, при условии, что $y_{i+2} \neq y_i$ и $x_{i+2} \neq x_i$ найдем:

$$\begin{aligned}
 x_{3,i+1} &= (x_{1,i+1} + R_{i+1} (R_{i+1} x_i + y_{1,i+1} - y_i)) / (R_{i+1}^2 + 1), \\
 y_{3,i+1} &= y_i + R_{i+1} (x_{3,i+1} - x_i), \\
 x_{4,i+1} &= (x_{2,i+1} + R_{i+1} (R_{i+1} x_i + y_{2,i+1} - y_i)) / (R_{i+1}^2 + 1), \\
 y_{4,i+1} &= y_i + R_{i+1} (x_{4,i+1} - x_i),
 \end{aligned} \tag{7}$$

где $R_{i+1} = \frac{y_{i+2} - y_i}{x_{i+2} - x_i}$. Если $y_{i+2} = y_i$ или $x_{i+2} = x_i$ (одновременно эти равенства выполняться не могут), то (7) приобретает вид:

$$x_{3,i+1} = x_{1,i+1}; \quad y_{3,i+1} = y_i; \quad x_{4,i+1} = x_{2,i+1}; \quad y_{4,i+1} = y_i, \tag{8}$$

или:

$$x_{3,i+1} = x_i; \quad y_{3,i+1} = y_{1,i+1}; \quad x_{4,i+1} = x_i; \quad y_{4,i+1} = y_{2,i+1}. \tag{8'}$$

Обозначим через $s_{1,i}$ расстояние от точки (x_i, y_i) до точки $(x_{3,i+1}; y_{3,i+1})$, а через $s_{2,i}$ расстояние от точки (x_i, y_i) до точки $(x_{4,i+1}; y_{4,i+1})$. Очевидно, что

$$\begin{aligned}
 s_{1,i} &= \sqrt{(x_i - x_{3,i+1})^2 + (y_i - y_{3,i+1})^2}, \\
 s_{2,i} &= \sqrt{(x_i - x_{4,i+1})^2 + (y_i - y_{4,i+1})^2}.
 \end{aligned} \tag{9}$$

В дальнейшем нам понадобится соединить точки $(x_{1,i+1}; y_{1,i+1})$ и $(x_{2,i}; y_{2,i})$ отрезками $[(x_{2,i}; y_{2,i}); (x_{1,i+1}; y_{1,i+1})]$, но так, чтобы они не имели общих точек. (см. рис.13). Для этого достаточно переименовать точки таким образом, чтобы за точкой $(x_{2,i}; y_{2,i})$ следовала точка $(x_{1,i+1}; y_{1,i+1})$. Об этом пойдет речь в следующем пункте.

3. Введем в рассмотрение угол γ_i , равный углу между векторами $\vec{a} = \vec{i}(x_{i+1} - x_i) + \vec{j}(y_{1,i+1} - y_i)$ и $\vec{b} = \vec{i}(x_{i+2} - x_i) + \vec{j}(y_{i+2} - y_i)$, см. рис.12. В дальнейшем нам понадобится значение $\cos \gamma_i$. Из свойства скалярного произведения находим:

$$\cos \gamma_i = \frac{(x_{i+1} - x_i)(x_{i+2} - x_i) + (y_{1,i+1} - y_i)(y_{i+2} - y_i)}{\sqrt{[(x_{i+1} - x_i)^2 + (y_{1,i+1} - y_i)^2] \cdot [(x_{i+2} - x_i)^2 + (y_{i+2} - y_i)^2]}}. \tag{10}$$

Введем новые точки $(x_{1,i+1}; u_{1,i+1})$ и $(x_{2,i+1}; u_{2,i+1})$. Если $\cos \gamma_i > 0$ и $s_{1,i} < s_{2,i}$, то

$$x_{1,i+1} = x_{1,i+1}; u_{1,i+1} = y_{1,i+1}; x_{2,i+1} = x_{2,i+1}; u_{2,i+1} = y_{2,i+1}, \quad (11)$$

в случае $s_{1,i} > s_{2,i}$

$$x_{1,i+1} = x_{2,i+1}; u_{1,i+1} = y_{2,i+1}; x_{2,i+1} = x_{1,i+1}; u_{2,i+1} = y_{1,i+1}. \quad (11')$$

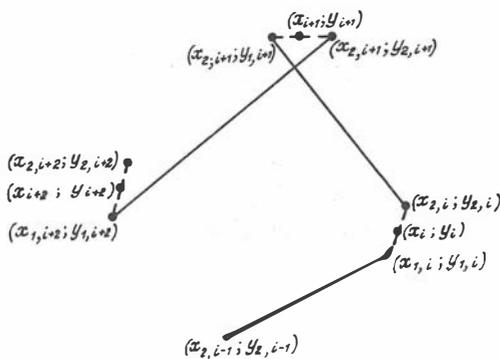


Рис.13

Если $\cos \gamma_i \leq 0$, то находим расстояние $S_{1,i}$ от точки (x_{i+2}, y_{i+2}) до точки $(x_{3,i+1}; y_{3,i+1})$ и расстояние $S_{2,i}$ от точки $(x_{i+2}; y_{i+2})$ до точки $(x_{4,i+1}; y_{4,i+1})$. Очевидно, что:

$$S_{1,i} = \sqrt{(x_{i+2} - x_{3,i+1})^2 + (y_{i+2} - y_{3,i+1})^2},$$

$$S_{2,i} = \sqrt{(x_{i+2} - x_{4,i+1})^2 + (y_{i+2} - y_{4,i+1})^2}. \quad (12)$$

Если $S_{1,i} > S_{2,i}$, то

$$x_{1,i+1} = x_{1,i+1}, u_{1,i+1} = y_{1,i+1}, x_{2,i+1} = x_{2,i+1}, u_{2,i+1} = y_{2,i+1}, \quad (13)$$

если же $S_{1,i} < S_{2,i}$, то

$$x_{1,i+1} = x_{2,i+1}, u_{1,i+1} = y_{2,i+1}, x_{2,i+1} = x_{1,i+1}, u_{2,i+1} = y_{1,i+1} \quad (13')$$

Таким образом, мы получили, что вместо старых точек $(x_{1,i+1}; y_{1,i+1})$ и $(x_{2,i+1}; y_{2,i+1})$ имеем новые точки $(x_{1,i+1}; u_{1,i+1})$ и $(x_{2,i+1}; u_{2,i+1})$, но эти новые точки упорядочены так, что за точкой $(x_{2,i}; u_{2,i})$ всегда следует точка $(x_{1,i+1}; u_{1,i+1})$, то есть отрезки $[(x_{2,i}; u_{2,i}); (x_{1,i+1}; u_{1,i+1})]$ и $[(x_{2,i+1}; u_{2,i+1}); (x_{1,i+2}; u_{1,i+2})]$ не имеют общих точек. (см. рис.14)

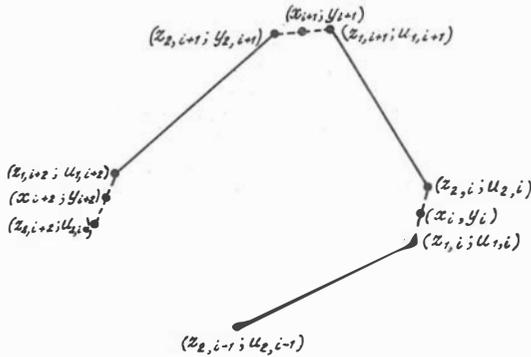


Рис.14

Заметим, что в случае замкнутой ломаной $x_{1,N} = x_{1,1}$; $x_{2,N} = x_{2,1}$; $u_{1,N} = u_{1,1}$; $u_{2,N} = u_{2,1}$, в случае незамкнутой $x_{1,1} = x_1$; $u_{1,1} = y_1$; $x_{1,N} = x_N$; $u_{1,N} = y_N$;

$$x_{2,1} = x_1; u_{2,1} = y_1; x_{2,N} = x_N; u_{2,N} = y_N.$$

4. Проведем прямую MK_i (см.рис.15) через точки $(x_{2,i}; u_{2,i})$ и $(x_{1,i+1}; u_{1,i+1})$. Она определяется уравнением:

$$y = u_{2,i} + \frac{u_{1,i+1} - u_{2,i}}{x_{1,i+1} - x_{2,i}} (x - x_{2,i}) \quad \text{ПРИ } x_{1,i+1} \neq x_{2,i}, \quad (14)$$

$$x = x_{2,i} \quad \text{ПРИ } x_{1,i+1} = x_{2,i}.$$

Обозначим угол, образованный прямой MK_i с осью x , через β_i . Очевидно, что

$$\beta_i = \begin{cases} \arctg \frac{u_{1,i+1} - u_{2,i}}{x_{1,i+1} - x_{2,i}} & \text{при } x_{1,i+1} \neq x_{2,i}, \\ \pi/2 & \text{при } x_{1,i+1} = x_{2,i}. \end{cases} \quad (15)$$

В дальнейшем нам не понадобятся старые значения точек $(x_{1,i}; y_{1,i})$ и $(x_{2,i}; y_{2,i})$, $i=1, 2, \dots, N$, поэтому эти обозначения мы будем применять для других точек. Обозначим через $(x_{2,i}; y_{2,i})$ точку, лежащую внутри отрезка $[(x_{2,i}; u_{2,i}); (x_{1,i+1}; u_{1,i+1})]$ на расстоянии δ_i от точки $(x_{2,i}; u_{2,i})$, а через $(x_{1,i+1}; y_{1,i+1})$ точку, лежащую внутри того же отрезка на расстоянии δ_{i+1} от точки $(x_{1,i+1}; u_{1,i+1})$, $i=1, \dots, N-1$.

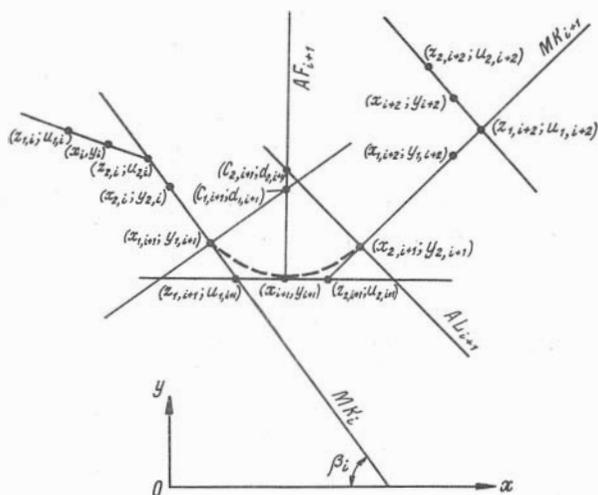


Рис. 15

Метод построения этих точек выглядит довольно просто. Для нахождения, например, точки $(x_{2,i}; y_{2,i})$ вводятся две точки $(a_1; \delta_1)$ и $(a_2; \delta_2)$, симметричные относительно точки $(x_{2,i}; u_{2,i})$ и лежащие на прямой MK_i , при помощи соотно-

шений:

$$\begin{aligned} \alpha_1 &= x_{2,i} + \delta_i \cos \beta_i, & \sigma_1 &= u_{2,i} + \delta_i \sin \beta_i, \\ \alpha_2 &= x_{2,i} - \delta_i \cos \beta_i, & \sigma_2 &= u_{2,i} - \delta_i \sin \beta_i. \end{aligned} \quad (16)$$

Находится расстояние z_1 между точками $(z_{1,i+1}; u_{1,i+1})$ и (α_1, σ_1) . Находится расстояние z_2 между точками $(z_{1,i+1}; u_{1,i+1})$ и (α_2, σ_2) . Эти расстояния сравниваются между собой. Если $z_1 > z_2$, то $x_{2,i} = \alpha_2$, $y_{2,i} = \sigma_2$, если $z_1 < z_2$, то $x_{2,i} = \alpha_1$, $y_{2,i} = \sigma_1$, и так повторяется при каждом изменении индекса i , $i = 1, \dots, N-1$. Аналогично находятся точки $(x_{1,i+1}; y_{1,i+1})$, только в предыдущем рассуждении нужно заменить $z_{1,i+1}$ на $z_{2,i}$, $u_{1,i+1}$ на $u_{2,i}$, $x_{2,i}$ на $x_{1,i+1}$, $y_{2,i}$ на $y_{1,i+1}$.

Через точку (x_{i+1}, y_{i+1}) перпендикулярно прямой CD_{i+1} проведем прямую AF_{i+1}

$$\begin{aligned} y &= y_{i+1} - \frac{x_{i+2} - x_i}{y_{i+2} - y_i} (x - x_{i+1}) & \text{при } y_{i+2} \neq y_i, \\ x &= x_{i+1} & \text{при } y_{i+2} = y_i. \end{aligned} \quad (17)$$

Через точку $(x_{1,i+1}; y_{1,i+1})$ перпендикулярно прямой MK_i проведем прямую AK_{i+1}

$$\begin{aligned} y &= y_{1,i+1} - \frac{z_{1,i+1} - z_{2,i}}{u_{1,i+1} - u_{2,i}} (x - x_{1,i+1}) & \text{при } u_{1,i+1} \neq u_{2,i}, \\ x &= x_{1,i+1} & \text{при } u_{1,i+1} = u_{2,i}. \end{aligned} \quad (18)$$

Очевидно, что прямая AF_{i+1} пересекается с прямой AK_{i+1} , иначе точка (x_{i+1}, y_{i+1}) являлась бы внутренней точкой, чего не может быть. Точку пересечения прямых AK_{i+1} и AF_{i+1} обозначим через $(c_{1,i+1}; d_{1,i+1})$. Она является решением системы уравнений (17), (18):

$$c_{1,i+1} = \frac{q_{2,i} x_{i+1} + y_{i+1} - q_{1,i} x_{1,i+1} - y_{1,i+1}}{q_{2,i} - q_{1,i}} \quad \begin{aligned} & \text{при } u_{1,i+1} \neq u_{2,i} \\ & y_{i+2} \neq y_i, \end{aligned} \quad (19)$$

$$c_{1,i+1} = x_{i+1} \quad \text{при } u_{1,i+1} \neq u_{2,i}, y_{i+2} = y_i,$$

$$c_{1,i+1} = x_{1,i+1} \quad \text{при } u_{1,i+1} = u_{2,i},$$

$$z_{1,i+1} = \sqrt{(x_{i+1} - c_{1,i+1})^2 + (y_{i+1} - d_{1,i+1})^2} \quad , \quad (24)$$

а точку $(c_{2,i+1}; d_{2,i+1})$ за центр окружности с радиусом

$$z_{2,i+1} = \sqrt{(x_{i+1} - c_{2,i+1})^2 + (y_{i+1} - d_{2,i+1})^2} \quad . \quad (25)$$

Очевидно, что окружность

$$z_{1,i+1}^2 = (x - c_{1,i+1})^2 + (y - d_{1,i+1})^2 \quad (26)$$

касается прямых MK_i и CD_{i+1} в точках $(x_{1,i+1}; y_{1,i+1})$ и $(x_{i+1}; y_{i+1})$. Окружность

$$z_{2,i+1}^2 = (x - c_{2,i+1})^2 + (y - d_{2,i+1})^2 \quad (27)$$

касается прямых CD_{i+1} и MK_{i+1} в точках $(x_{i+1}; y_{i+1})$ и $(x_{2,i+1}; y_{2,i+1})$. Значение участков окружностей, заключенные между точками $(x_{1,i+1}; y_{1,i+1})$ и $(x_{i+1}; y_{i+1})$, а также между точками $(x_{i+1}; y_{i+1})$ и $(x_{2,i+1}; y_{2,i+1})$ принимаем в качестве значений искомой (аппроксимирующей) функции, заключенных между теми же точками.

Между точками $(x_{2,i}; y_{2,i})$ и $(x_{1,i+1}; y_{1,i+1})$ за искомую функцию принимаем уравнение прямой MK_i .

5. При нахождении аппроксимирующей функции на ЭВМ, т.е. при нахождении ее дискретных значений с шагом h , следует учесть особенности расположения окружностей, о которых говорилось выше. При выводе линейной части функции на печать достаточно ввести начальную точку линейности функции и конечную, точнее точку $(x_{2,i}; y_{2,i})$ и $(x_{1,i+1}; y_{1,i+1})$. В случае замкнутой кривой первой точкой и последней выводится $(x_{2,1}; y_{2,1})$. В случае незамкнутой кривой нужно выводить в качестве первой точки не $(x_{2,1}; y_{2,1})$, а $(x_1; y_1)$ и последней вывести $(x_N; y_N)$. В противном случае будет допущена грубая ошибка. Далее, если угловой коэффициент k_{i+1} прямой CK_{i+1} , проходящей через точки $(x_{1,i+1}; y_{1,i+1})$ и $(c_{1,i+1}; d_{1,i+1})$, по модулю больше единицы, то следует находить значение y , изменяя x с шагом h от точки $(x_{1,i+1}; y_{1,i+1})$ до $(x_{i+1}; y_{i+1})$ включительно.

y находится из уравнения окружности (26):

$$y = \xi \sqrt{z_{2,i+1}^2 - (x - C_{1,i+1})^2 + d_{1,i+1}}, \quad (28)$$

где $\xi = -1$, если $d_{1,i+1} > y_{i+1}$ и $\xi = +1$, если $d_{1,i+1} < y_{i+1}$. Так как $|k_{i+1}| > 1$, то $d_{1,i+1} \neq y_{i+1}$. В случае, когда $|k_{i+1}| \leq 1$, следует находить значение x , изменяя y с шагом h . x находится из уравнения окружности (26):

$$x = \eta \sqrt{z_{2,i+1}^2 - (y - d_{1,i+1})^2 + C_{1,i+1}}, \quad (29)$$

где $\eta = 1$, если $C_{1,i+1} < x_{i+1}$ и $\eta = -1$, если $C_{1,i+1} > x_{i+1}$. Так как $|k_{i+1}| \leq 1$, то $C_{1,i+1} \neq x_{i+1}$. Для окружности (27) с центром в точке $(C_{2,i+1}; d_{2,i+1})$ рассуждения останутся аналогичными, если провести прямую через точки $(C_{2,i+1}; d_{2,i+1})$ и $(x_{2,i+1}; y_{2,i+1})$.

1. Описание блок-схем и программ "Контур-3"

Схему решения задачи можно представить следующими основными блоками.

Б л о к А — подготовка к решению задачи. Ввод исходных данных; формирование констант; проверка условия замкнутости ломаной линии; отыскание внутренних точек, выбрасывание их, перенумерация оставшихся; проверка условия расположения двух соседних точек на расстоянии, меньшем, чем заданная величина $4\delta_i$; преобразование величин δ_i в новые δ_i' таким образом, чтобы любые две соседние точки лежали на расстоянии большем, чем $4\delta_i'$; δ_i' вновь обозначаются через δ_i .

Б л о к В — определение касательных. Проводится прямая через две точки, у которых индексы отличаются на две единицы; принимается за касательную прямую к иско-

мой гладкой кривой в точке, лежащей между двумя вышеуказанными, прямая, параллельная найденной; проверяется условие замкнутости ломаной; в случае замкнутой линии первая и последняя точки совпадают, и в этих точках касательная определяется посредством второй и предпоследней точками; при незамкнутой ломаной в первой и последней точках производные не определяются.

Б л о к С — определение вспомогательных точек. Откладывается на каждой касательной в одну и другую стороны на расстоянии δ_i от заданных точек, определяющих ломаную линию, пара вспомогательных для нахождения точек касания окружностей, при помощи которых ломаная линия будет сглажена; упорядочивание вспомогательных точек в последовательность, удобную для решения задачи; в случае незамкнутой линии в окрестностях первой и последней точек вспомогательные не находятся.

Б л о к Д — построение точек касания окружностей и линейных участков гладкой кривой. Каждые две соседние вспомогательные точки, лежащие на разных касательных, соединяются отрезками; на каждом из таких отрезков находятся две точки, лежащие на расстоянии δ_i от вспомогательных; построенные точки принимаются за начало и конец участка линейности и за точки касания окружностей.

Б л о к Е — построение окружностей. Находятся центры сглаживающих окружностей и строятся последние; в окрестности каждой из точек, задающих ломаную, получается две окружности, причем точка является общей, в которой касательная к одной окружности и касательная к другой совпадают; при незамкнутой ломаной в окрестности первой и последней точках нет построения окружностей.

Б л о к F — выдача результатов. В случае незамкнутой линии первой и последней точками гладкой кривой выводятся соответствующие точки ломаной; на линей-

ных участках кривой вводятся начальная точка линейности и последняя, то-есть точки касания окружностей; при вводе нелинейных участков аппроксимирующей кривой учитывается расположение прямых, соединяющих центры окружностей с вспомогательными точками: а) если прямая имеет угловой коэффициент по абсолютной величине больше 1, то выводятся точки окружности на заданном участке нелинейности с шагом h по оси x ; б) если угловой коэффициент прямой по абсолютной величине не превосходит 1, то выводятся точки окружности с шагом h по оси y . На каждом нелинейном участке в качестве первой точки вводится точка касания окружности.

Б л о к Я — конец задачи.

О п и с а н и е р а б о т н о п е р а т о р н о й с х е м ы

Программа состоит из ряда самостоятельных блоков. Б л о к А — подготовка к решению задачи. В состав блока входят следующие операторы:

- F_1 — формирование числа $\pi = 3,14159265$;
- F_2 — формирование числа $N = N_1$, где N_1 — количество всех точек, включая и внутренние;
- P_3 — проверка условия замкнутости ломаной линии;
- A_4 — вычисление значений $X_{1,N-1} = x_N - x_{N-1}$; $Y_{1,N-1} = y_N - y_{N-1}$
 $X_{2,N-1} = x_2 - x_{N-1}$; $Y_{2,N-1} = y_2 - y_{N-1}$;
- P_5 — проверка того, является ли точка (x_N, y_N) внутренней;
- F_6 — формирование $j = 1$;
- F_7 — формирование $x_i = x_{i+1}$; $y_i = y_{i+1}$;
- P_8 — проверка условия $i \leq N-2$;
- F_9 — формирование $i = i + 1$;
- F_{10} — формирование $x_{N-1} = x_1$;
- F_{11} — формирование $N = N-1$;

- F_{72} - формирование $i=1$;
 A_{13} - вычисление $X_{1,i} = x_{i+1} - x_i$; $Y_{1,i} = y_{i+1} - y_i$;
 $X_{2,i} = x_{i+2} - x_i$; $Y_{2,i} = y_{i+1} - y_i$;
 P_{14} - проверка на существование внутренних точек .
 F_{15} - формирование $j = i + 1$;
 F_{16} - формирование $x_j = x_{j+1}$, $y_j = y_{j+1}$, т.е. перенумерация точек , если существуют внутренние ;
 P_{17} - проверка $j < N-1$;
 F_{18} - формирование $j = j + 1$;
 F_{19} - формирование $N = N - 1$;
 P_{20} - проверка условия $i < N-2$;
 F_{21} - формирование $i = i + 1$;
 F_{22} - формирование $i = 1$;
 A_{23} - вычисление расстояния $X_{1,i}$ между двумя соседними точками , $X_{1,i} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$;
 P_{24} - проверка условия $\delta_i > X_{1,i} / 4$;
 F_{25} - формирование $\delta_i = X_{1,i} / 4$; $\delta_{i+1} = X_{1,i} / 4$;
 P_{26} - проверка условия $i < N-1$;
 F_{27} - формирование $i = i + 1$;

Операторная схема блока А имеет вид :

$$\begin{array}{cccccccccccccccc}
 F_1 & F_2 & P_{3+12} & A_4 & P_{5+12} & F_6 & P_7 & P_{8+10} & F_9 & F_{10} & F_{11} & P_{12} & P_{13} & A_{13} \\
 P_{14+20} & F_{15} & P_{16} & P_{17+19} & F_{18} & P_{19} & P_{20+22} & F_{21} & P_{22} & P_{23} & P_{24+26} \\
 F_{25} & P_{26+28} & F_{27}
 \end{array}$$

Операторная схема работает следующим образом. Операторы F_1 и F_2 формируют числа $\pi = 3,14159265$ и $N = N_1$, где N_1 - количество всех заданных точек ломаной линии, включая и внутренние. Заметим, что число N_1 вводится. Оператор P_3 проверяет условие замкнутости ломаной линии. Если линия замкнута, то управление передается оператору A_4 , если незамкнута, то управление передается оператору F_{12} , который формирует число $i=1$. Оператор A_4 вычисляет разность компонент $(N-1)$ -й

точки и N -й точки и передает управление оператору P_5 , который проверяет, является ли точка (x_N, y_N) внутренней. Если точка (x_N, y_N) не является внутренней, то управление передается оператору F_{12} . Если же указанная точка является внутренней, то управление передается оператору F_8 . Операторы $F_6, F_7, P_8, F_9, F_{10}$ реализуют выбрасывание внутренней точки (x_N, y_N) , если она таковой является, и перенумеровывают оставшиеся точки в той же последовательности. В этом случае оператор F_{11} уменьшает число N на единицу и передает управление оператору F_{22} . Операторы $A_{13}, P_{14}, F_{15}, F_{16}, P_{17}, F_{18}, F_{19}, F_{20}, F_{21}$ реализуют выбрасывание внутренних точек на оставшихся участках ломаной, если внутренних точек нет, то управление передается оператору F_{22} , который формирует число $i = 1$ и передает управление оператору A_{23} . Оператор A_{23} вычисляет расстояние между двумя соседними точками и передает управление оператору P_{24} . Оператор P_{24} проверяет, больше ли заданное число δ_i четверти этого расстояния или меньше (δ_i вводится). В зависимости от этого управление передается оператору F_{25} или оператору P_{26} . Оператор F_{25} обеспечивает формирование новых δ_i и δ_{i+1} , равных четверти вышеупомянутого расстояния и передает управление оператору P_{26} . Оператор P_{26} передает управление оператору F_{28} , если проверены все расстояния между точками, в противном случае передает управление оператору F_{27} , который образует цикл. После этого работа блока А закончена. Происходит переход на оператор F_{28} блока В.

Б л о к В – определение касательных. В состав блока входят следующие операторы:

F_{28} – формирование $i = 1$;

P_{29} – проверка $x_{i+2} \neq x_i$;

A_{30} – нахождение угла $\alpha_{i+1} = \arctg \frac{y_{i+2} - y_i}{x_{i+2} - x_i}$;

A_{31} – вычисление угла $\alpha_{i+1} = \pi / 2$.

Операторная схема блока В имеет вид:

$${}^{26}F_{28} \quad {}^{49}P_{29+31} \quad A_{30}^{32} \quad {}^{29}A_{31}$$

Операторная схема работает следующим образом. Оператор F_{28} формирует число $i=1$. Оператор P_{29} проверяет условие $x_{i+2} \neq x_i$. Если условие выполнено, то управление передается оператору A_{30} . В противном случае управление передается оператору A_{31} . Операторы A_{30} и A_{31} вычисляют углы наклона касательных.

Б л о к С — определение вспомогательных точек.
 В состав блока входят следующие операторы:

A_{32} — вычисление $x_{3,i+1}$, $y_{3,i+1}$, $x_{4,i+1}$, $y_{4,i+1}$ по формулам (3).

P_{33} — проверка условия $x_{i+2} \neq x_i$.

P_{34} — проверка условия $y_{i+2} \neq y_i$.

A_{35} — вычисление $R_{i+1} = \frac{y_{i+2} - y_i}{x_{i+2} - x_i}$, а также значений

$x_{3,i+1}$, $y_{3,i+1}$, $x_{4,i+1}$, $y_{4,i+1}$ по формуле (7).

A_{36} — вычисление $x_{3,i+1}$, $y_{3,i+1}$, $x_{4,i+1}$, $y_{4,i+1}$ по формуле (8).

A_{37} — вычисление $x_{3,i+1}$, $y_{3,i+1}$, $x_{4,i+1}$, $y_{4,i+1}$ по формуле (8').

A_{38} — вычисление $s_{1,i}$, $s_{2,i}$ по формуле (9).

A_{39} — вычисление $\cos \gamma'_0$ по формуле (10).

F_{40} — формирование $A = \cos \gamma'_0$.

F_{41} — вычисление $S_{1,i}$, $S_{2,i}$ по формуле (12).

P_{42} — проверка условия $A_i > 0$.

P_{43} — проверка условия $s_{1,i} < s_{2,i}$.

F_{44} — формирование $x_{1,i+1}$, $y_{1,i+1}$, $x_{2,i+1}$, $y_{2,i+1}$ по формулам (11).

F_{45} — формирование $x_{1,i+1}$, $y_{1,i+1}$, $x_{2,i+1}$, $y_{2,i+1}$ по формулам (11').

P_{46} — проверка условия $S_{1,i} < S_{2,i}$.

F_{47} — формирование $x_{1,i+1}$, $y_{1,i+1}$, $x_{2,i+1}$, $y_{2,i+1}$ по формулам (13').

F_{48} — формирование $x_{1,i+1}$, $y_{1,i+1}$, $x_{2,i+1}$, $y_{2,i+1}$ по формулам (13).

F_{49} — проверка условия $i < N-2$.

F_{50} — формирование $i = i+1$.

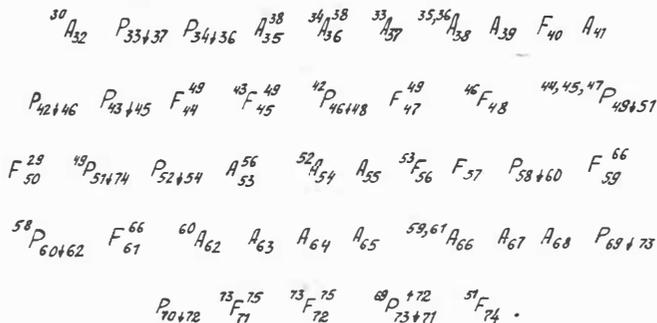
- P_{51} - проверка замкнутости ломаной линии.
 P_{52} - проверка условия $x_{N-1} = x_2$.
 A_{53} - вычисление $\alpha_1 = \pi/2$.
 A_{54} - вычисление $R_1 = \frac{y_{N-1} - y_2}{x_{N-1} - x_2}$.
 A_{55} - вычисление $\alpha_1 = \arctg(R_1)$.
 F_{56} - формирование $i = 0$.
 F_{57} - формирование $x_{1,1}, y_{1,1}, x_{2,1}, y_{2,1}$ по формуле (3).
 P_{58} - проверка условия $x_{N-1} = x_2$.
 F_{59} - формирование $x_{3,1} = x_2, y_{3,1} = y_{1,1}, x_{4,1} = x_2, y_{4,1} = y_{2,1}$.
 P_{60} - проверка условия $y_{N-1} = y_2$.
 F_{61} - формирование $x_{3,1} = x_{1,1}, y_{3,1} = y_{N-1}, x_{4,1} = x_{2,1}, y_{4,1} = y_{N-1}$.
 A_{62} - вычисление $x_{3,1} = \frac{x_{2,1} + R_1(R_1 x_2 + y_{1,1} - y_2)}{R_1^2 + 1}$.
 A_{63} - вычисление $y_{3,1} = y_2 + R_1(x_{3,1} - x_2)$.
 A_{64} - вычисление $x_{4,1} = \frac{x_{2,1} + R_1(R_1 x_2 + y_{1,1} - y_2)}{R_1^2 + 1}$.
 A_{65} - вычисление $y_{4,1} = y_2 + R_1(x_{4,1} - x_2)$.
 A_{66} - вычисление расстояний от точки (x_{N-1}, y_{N-1}) до $(x_{3,1}, y_{3,1})$, от точки (x_{N-1}, y_{N-1}) до $(x_{4,1}, y_{4,1})$, от точки (x_2, y_2) до $(x_{3,1}, y_{3,1})$ от точки (x_2, y_2) до $(x_{4,1}, y_{4,1})$.
 A_{67} - вычисление $X_{1,N} = x_1 - x_{N-1}, X_{2,N} = x_2 - x_{N-1}, Y_{1,N} = y_1 - y_{N-1}, Y_{2,N} = y_2 - y_{N-1}$.
 A_{68} - вычисление $A_{N-1} = \frac{X_{1,N} X_{2,N} + Y_{1,N} Y_{2,N}}{\sqrt{(X_{1,N}^2 + Y_{1,N}^2)(X_{2,N}^2 + Y_{2,N}^2)}}$.
 P_{69} - проверка условия $A_{N-1} > 0$.
 P_{70} - проверка условия $s_{1,N-1} < s_{2,N-1}$.
 F_{71} - формирование $z_{1,N} = z_{1,1} = x_1, u_{1,N} = u_{1,1} = y_{1,1}, z_{2,N} = z_{2,1} = x_{2,1}, u_{2,N} = u_{2,1} = y_{2,1}$.
 F_{72} - формирование $z_{1,N} = z_{1,1} = x_{2,1}, z_{2,N} = z_{2,1} = x_{1,1}, u_{1,N} = u_{1,1} = y_{2,1}, u_{2,N} = u_{2,1} = y_{1,1}$.

P_{73} - проверка условия $S_{1, N-1} < S_{2, N-1}$.

F_{74} - формирование $x_{3,1} = x_{2,1} = x_1$, $u_{1,1} = u_{2,1} = y_1$,

$x_{3,N} = x_{2,N} = x_N$, $u_{3,N} = u_{2,N} = y_N$.

Операторная схема блока С имеет вид:



Операторная схема работает следующим образом. Оператор A_{32} находит вспомогательные точки по формулам (3) и передает управление условному оператору P_{33} . Если условие последнего выполняется, то управление передается условному оператору P_{34} . В противном случае управление передается оператору A_{37} . При выполнении условия оператора P_{34} управление передается оператору A_{35} , который определяет по формулам (7) дополнительные точки, необходимые для упорядочивания (ориентации) вспомогательных. Если условие оператора P_{34} не выполняется, то управление передается оператору A_{36} . Операторы A_{36} и A_{37} вычисляют те же точки (необходимые для ориентации), но соответственно по формулам (8) и (8'). Операторы A_{35} , A_{36} , A_{37} передают управление оператору A_{38} . Последний вычисляет расстояния $s_{1,i}$, $s_{2,i}$ по формуле (9) и передает управление оператору A_{39} , который находит значение $\cos \gamma_i$ по формуле (10). Оператор A_{39} передает управление оператору F_{40} , который формирует число $A_i = \cos \gamma_i$ и передает управление оператору A_{41} . Опера-

тор A_{47} вычисляет расстояния $S_{1,i}$ и $S_{2,i}$ по формулам (12) и передает управление условному оператору P_{42} . Операторы $P_{42}, P_{43}, F_{44}, F_{45}, F_{46}, F_{47}, F_{48}, P_{49}, F_{50}$ проделывают процедуру упорядочивания вспомогательных точек. Условный оператор P_{49} и оператор F_{50} организовывают цикл. Если условие оператора P_{49} выполнено, то управление передается оператору F_{50} , который формирует число $i = i + 1$ и передает управление оператору P_{51} . В случае невыполнения условия оператора P_{49} (цикл закончился) управление передается условному оператору P_{51} . Последний проверяет, замкнута ли ломаная линия. Если линия замкнута, то работают операторы $P_{52} - F_{73}$. Эти операторы выполняют действия, аналогичные операторам $P_{29} - F_{48}$, но только для точек с индексами $N-1, N-2$. Это очень наглядно, если заменить точки с индексами $i, i+1, i+2$ в операторах $P_{29} - F_{48}$ на точки с индексами $N-1, N-2$ соответственно. Если условие оператора P_{51} не выполняется (линия незамкнута), то управление передается оператору F_{74} , в котором заканчивается упорядочивание вспомогательных точек. Происходит переход на оператор F_{75} блока Д.

Б л о к Д — построение точек касания окружностей и линейных участков гладкой кривой.

В состав блока входят следующие операторы:

F_{75} — формирование $i = 1$.

P_{76} — проверка условия $Z_{1,i+1} = Z_{2,i}$.

A_{77} — вычисление $\alpha_i = \pi/2$.

A_{78} — вычисление $\alpha_i = \arctg \frac{u_{1,i+1} - u_{2,i}}{z_{1,i+1} - z_{2,i}}$.

P_{79} — проверка условия $i < N-1$.

F_{80} — формирование $i = i + 1$.

F_{81} — формирование $i = 1$.

A_{82} — вычисление $a_1 = z_{2,i} + \delta_i \cos \alpha_i$.

A_{83} — вычисление $b_1 = u_{2,i} + \delta_i \sin \alpha_i$.

A_{84} — вычисление $a_2 = z_{2,i} - \delta_i \cos \alpha_i$.

A_{85} — вычисление $b_2 = u_{2,i} - \delta_i \sin \alpha_i$.

A_{86} — вычисление $e_1 = \sqrt{(z_{1,i+1} - a_1)^2 + (u_{1,i+1} - b_1)^2}$.

Заметим, что вышеупомянутые отрезки лежат на прямых MK_i . Укажем несколько подробней схему работы операторов $A_{92} - F_{101}$. Операторы $A_{92} - A_{95}$ находят точки (a_1, b_1) и (a_2, b_2) , лежащие в одну и другую стороны от точки $(z_{2,i}, u_{2,i})$ на расстоянии δ_i по прямой MK_i . Оператор A_{95} передает управление оператору A_{96} . Операторы $A_{96} - A_{97}$ находят расстояния от точки $(z_{1,i+1}, u_{1,i+1})$ до (a_1, b_1) и до (a_2, b_2) . Оператор A_{97} передает управление оператору F_{98} , который сравнивает найденные расстояния между собой.

Условный оператор F_{98} передает управление оператору F_{99} , если условие выполнено, в противном случае управление передается оператору F_{90} . Оператор F_{99} или F_{90} (в зависимости от выполнения или невыполнения условия оператора F_{98}) находит точку начала i -го линейного участка строящейся линии. В качестве такой точки принимается ближайшая из точек (a_1, b_1) , (a_2, b_2) к вспомогательной точке $(z_{1,i+1}, u_{1,i+1})$. Найденная точка обозначается через $(x_{2,i}, y_{2,i})$.

Операторы $A_{91} - F_{99}$ аналогично операторам $A_{92} - F_{90}$ находят точку $(x_{1,i+1}, y_{1,i+1})$, которая будет являться концом i -го линейного участка строящейся линии. Чтобы не вводить лишних символов, мы использовали старые обозначения $(x_{2,i}, y_{2,i})$ и $(x_{1,i}, y_{1,i})$ в качестве новых. Так как мы строим гладкую линию, то примем в качестве точек касания окружностей, с помощью которых ломаная сглаживается, найденные точки.

Операторы F_{100}, F_{101} организуют цикл для нахождения точек. По окончании цикла происходит переход на оператор F_{102} блока Е.

Б л о к Е - построение окружностей. В состав блока входят следующие операторы:

F_{102} - формирование $i = 1$.

F_{103} - проверка условия $u_{1,i+1} \neq u_{2,i}$.

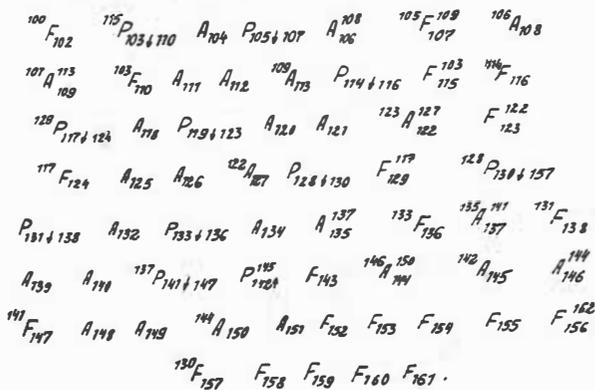
F_{104} - вычисление $q_{1,i} = \frac{z_{1,i+1} - z_{2,i}}{u_{1,i+1} - u_{2,i}}$.

F_{105} - проверка условия $y_{i+2} \neq y_i$.

- A_{106} - вычисление $q_{2,i} = \frac{x_{i+2} - x_i}{y_{i+2} - y_i}$.
- F_{107} - формирование $C_{1,i+1} = x_{i+1}$.
- A_{108} - вычисление $C_{1,i+1} = \frac{q_{2,i} x_{i+1} + y_{i+1} - q_{2,i} x_{i+2} - y_{i+2}}{q_{2,i} - q_{1,i}}$.
- A_{109} - вычисление $d_{1,i+1} = y_{i+1} - q_{1,i} (C_{1,i+1} - x_{i+1})$.
- F_{110} - формирование $C_{1,i+1} = x_{1,i+1}$.
- A_{111} - вычисление $q_{2,i} = \frac{x_{i+2} - x_i}{y_{i+2} - y_i}$.
- A_{112} - вычисление $d_{1,i+1} = y_{i+1} - q_{2,i} (C_{1,i+1} - x_{i+1})$.
- A_{113} - вычисление $z_{1,i+1}$ по формуле (24).
- F_{114} - проверка условия $i < N-2$.
- F_{115} - формирование $i = i+1$.
- F_{116} - формирование $i = 1$.
- F_{117} - проверка условия $y_{i+2} \neq y_i$.
- A_{118} - вычисление $q_{2,i} = \frac{x_{i+2} - x_i}{y_{i+2} - y_i}$.
- F_{119} - проверка условия $u_{1,i+2} \neq u_{2,i+1}$.
- A_{120} - вычисление $q_{3,i} = \frac{z_{1,i+2} - z_{2,i+1}}{u_{1,i+2} - u_{2,i+1}}$.
- A_{121} - вычисление $C_{2,i+1} = \frac{q_{3,i} x_{2,i+1} + y_{2,i+1} - q_{3,i} x_{1,i+1} - y_{1,i+1}}{q_{3,i} - q_{2,i}}$.
- A_{122} - вычисление $d_{2,i+1} = y_{i+1} - q_{2,i} (C_{2,i+1} - x_{i+1})$.
- F_{123} - формирование $C_{2,i+1} = x_{2,i+1}$.
- F_{124} - формирование $C_{2,i+1} = x_{i+1}$.
- A_{125} - вычисление $q_{3,i} = \frac{z_{1,i+2} - z_{2,i+1}}{u_{1,i+2} - u_{2,i+1}}$.
- A_{126} - вычисление $d_{2,i+1} = y_{2,i+1} - q_{3,i} (C_{2,i+1} - x_{2,i+1})$.
- A_{127} - вычисление $z_{2,i+1}$ по формуле (25).
- A_{128} - проверка условия $i < N-1$.
- F_{129} - формирование $i = i+1$.
- F_{130} - проверка условия $x_1 = x_N$ и $y_1 = y_N$.
- F_{131} - проверка условия $y_{N-1} = y_2$.
- A_{132} - вычисление $a_1 = (x_{N-1} - x_2) / (y_{N-1} - y_2)$.
- F_{133} - проверка условия $u_{1,N} \neq u_{2,N-1}$.
- A_{134} - вычисление $\sigma_1 = (z_{1,N} - z_{2,N-1}) / (u_{1,N} - u_{2,N-1})$.
- A_{135} - вычисление $C_{1,N} = (\sigma_1 x_{1,N} + y_{1,N} - a_1 x_1 - y_1) / (\sigma_1 - a_1)$.
- F_{136} - формирование $C_{1,N} = x_{1,N}$.

- A_{137} - вычисление $d_{1,N} = y_1 - a_1 (C_{1,N} - x_1)$.
 F_{138} - формирование $C_{1,N} = x_1$.
 A_{139} - вычисление $\sigma_1 = (x_{1,N} - x_{2,N-1}) / (u_{1,N} - u_{2,N-1})$.
 A_{140} - вычисление $d_{1,N} = y_{1,N} - \sigma_1 (C_{1,N} - x_{1,N})$.
 P_{141} - проверка условия $y_{N-1} \neq y_2$.
 P_{142} - проверка условия $u_{2,1} \neq u_{1,2}$.
 F_{143} - формирование $C_{2,N} = x_{2,1}$.
 A_{144} - вычисление $d_{2,N} = y_1 - a_1 (C_{2,N} - x_1)$.
 A_{145} - вычисление $\sigma_1 = (x_{2,1} - x_{1,2}) / (u_{2,1} - u_{1,2})$.
 A_{146} - вычисление $C_{2,N} = (\sigma_1 x_{2,1} + y_{2,1} - a_1 x_1 - y_1) / (\sigma_1 - a_1)$.
 F_{147} - формирование $C_{2,N} = x_1$.
 A_{148} - вычисление $\sigma_1 = (x_{2,1} - x_{1,2}) / (u_{2,1} - u_{1,2})$.
 A_{149} - вычисление $d_{2,N} = y_{2,1} - \sigma_1 (C_{2,N} - x_{2,1})$.
 A_{150} - вычисление $z_{1,N} = (x_N - C_{1,N})^2 + (y_N - d_{1,N})^2$.
 A_{151} - вычисление $z_{2,N} = (x_N - C_{2,N})^2 + (y_N - d_{2,N})^2$.
 F_{152} - формирование $x_{2,N} = x_{2,1}$.
 F_{153} - формирование $z_{2,N} = z_{2,1}$.
 F_{154} - формирование $u_{2,N} = u_{2,1}$.
 F_{155} - формирование $y_{2,N} = y_{2,1}$.
 F_{156} - формирование $k = N - 1$.
 F_{157} - формирование $x_{2,1} = x_1$.
 F_{158} - формирование $y_{2,1} = y_1$.
 F_{159} - формирование $x_{1,N} = x_N$.
 F_{160} - формирование $y_{1,N} = y_N$.
 F_{161} - формирование $k = N - 2$.

- Операторная схема блока E имеет вид:



Эта схема работает следующим образом. Оператор F_{102} формирует число $i=1$. Управление передается условному оператору P_{103} , который проверяет условие наклона прямой MK_i к оси OY под углом, не равным $\pi/2$, тем самым избавляет в дальнейшем от деления на "0". Оператор P_{103} передает управление оператору A_{104} , если условие выполнено, оператору F_{110} , если условие не выполнено. Оператор A_{104} вычисляет угловой коэффициент прямой MK_i и передает управление оператору P_{105} . Оператор P_{105} проверяет условие наклона к оси Ox касательной в точке (x_{i+1}, y_{i+1}) под углом, не равным "0" и передает управление оператору A_{106} , если условие выполнено, в противном случае управление передается оператору F_{117} . Оператор A_{106} находит угловой коэффициент касательной в точке (x_{i+1}, y_{i+1}) и передает управление оператору A_{108} . Операторы F_{107} , A_{108} , A_{109} , F_{110} , A_{111} , A_{112} находят центр $(C_{1,i+1}, d_{1,i+1})$ окружности, которая сглаживает участок ломаной линии от конца i -го линейного участка строящейся гладкой кривой до точки (x_{i+1}, y_{i+1}) .

Заметим, что не все операторы из этой группы работают при заданном i , т.к. работа их полностью зависит от условных операторов P_{103} и P_{105} .

Один из операторов A_{109} или A_{112} передает управление оператору A_{113} , который вычисляет радиус $r_{1,i+1}$ окружности. Управление передается условному оператору P_{114} . В случае выполнения условия оператора P_{114} , управление передается оператору F_{115} , который формирует число $i=i+1$, и, передав управление на P_{103} , организует цикл для нахождения $(C_{1,i+1}, d_{1,i+1})$. Если условие оператора P_{114} не выполняется, то управление передается оператору F_{116} , который формирует число $i=1$ и начинается новый цикл.

Группа операторов P_{114} - A_{126} выполняет действия, необходимые для нахождения координат $(C_{2,i+1}, d_{2,i+1})$ центра окружности, которая сглаживает участок ломаной линии от точки (x_{i+1}, y_{i+1}) до начала $(i+1)$ -го линейного участка строящейся гладкой кривой. Затем управление передается оператору A_{127} , который вычисляет радиус указанной окружности и передает управление условному оператору P_{128} .

Если условие P_{128} выполнено, то управление передается оператору F_{129} , который формирует число $i = i + 1$, и, передав управление на условный оператор P_{117} , организует цикл нахождения точек $(C_{2, i+1}, d_{2, i+1})$, а также $e_{2, i+1}$. В случае невыполнения условия оператора P_{129} управление передается на условный оператор P_{130} . Последний проверяет условие замкнутости ломаной линии. Если ломаная замкнута, то управление передается оператору F_{131} . Группа операторов $P_{131} - A_{149}$ находит центры двух окружностей, сглаживающих ломаную от конца $(x_{1, N}, y_{1, N})$ N -го линейного участка строящейся кривой до точки (x_1, y_1) и от этой точки до начала 1-го линейного участка. Управление передается оператору A_{150} . Операторы A_{150}, A_{151} находят радиусы $e_{1, N}$ и $e_{2, N}$ только что упомянутых окружностей. В случае замкнутой ломаной точки $(x_{2, N}, y_{2, N})$, $(x_{2, N}, u_{2, N})$ совпадают соответственно с точками $(x_{2, 1}, y_{2, 1})$, $(x_{2, 1}, u_{2, 1})$. Операторы $F_{152} - F_{155}$ приравнивают эти точки и управление передается оператору F_{156} .

Оператор F_{156} формирует число $\kappa = N - 1$ и передает управление оператору F_{162} блока F .

Если ломаная незамкнута, то оператор P_{130} передает управление оператору F_{157} . В случае незамкнутой ломаной точки $(x_{2, 1}, y_{2, 1})$ и $(x_{1, N}, y_{1, N})$ не участвуют в решении задачи, но обозначение этих точек удобно для организации цикла при выдаче результатов, а точки (x_1, y_1) и (x_N, y_N) вносятся в качестве первой и последней точек строящейся кривой и неудобны их обозначения. Ввиду этого операторы $F_{157} - F_{160}$ значение точек (x_1, y_1) и (x_N, y_N) придают соответственно точкам $(x_{2, 1}, y_{2, 1})$ и $(x_{1, N}, y_{1, N})$. Далее управление передается оператору F_{161} , который формирует число $\kappa = N - 2$ и передает управление оператору F_{162} блока F .

Б л о к F - выдача результатов. В состав блока входят следующие операторы:

F_{162} - формирование $i = 1$.

P_{163} - проверка условия $C_{1, i+1} \neq z_{1, i+1}$.

A_{164} - вычисление $y_{1, i+1} = (d_{1, i+1} - u_{1, i+1}) / (C_{1, i+1} - z_{1, i+1})$.

- F_{165} - формирование $y_{1,i+1} = 10^8$.
 P_{166} - проверка условия $C_{2,i+1} \neq x_{2,i+1}$.
 F_{167} - формирование $y_{2,i+1} = 10^8$.
 A_{168} - вычисление $y_{2,i+1} = (d_{2,i+1} - a_{2,i+1}) / (C_{2,i+1} - x_{2,i+1})$.
 P_{169} - проверка условия $i < k$.
 F_{170} - формирование $i = i + 1$.
 F_{171} - формирование $i = 1$.
 F_{172} - формирование $R = x_{2,i}$.
 F_{173} - формирование $f = y_{2,i}$.
 B_{174} - вывод на печать R, f .
 F_{175} - формирование $R = x_{1,i+1}$.
 F_{176} - формирование $f = y_{1,i+1}$.
 B_{177} - вывод на печать R, f .
 P_{178} - проверка условия $|y_{1,i+1}| \leq 1$.
 P_{179} - проверка условия $y_{1,i+1} > \bar{y}_{i+1}$.
 F_{180} - формирование $H = -h$.
 F_{181} - формирование $H = h$.
 A_{182} - вычисление $l = \text{entier} \left\lfloor \frac{y_{i+1} - y_{1,i+1}}{h} \right\rfloor$.
 F_{183} - формирование $j = 0$.
 P_{184} - проверка условия $C_{1,i+1} > x_{i+1}$.
 F_{185} - формирование $a_1 = -1$.
 F_{186} - формирование $j = j + 1$.
 F_{187} - формирование $a_1 = 1$.
 A_{188} - вычисление $f = y_{2,i+1} + jH$.
 A_{189} - вычисление $R = a_1 x \sqrt{v_{1,i+1} - (f - d_{1,i+1})^2} + C_{1,i+1}$.
 B_{190} - вывод на печать R, f .
 P_{191} - проверка условия $j < l$.
 P_{192} - проверка условия $x_{1,i+1} < x_{i+1}$.
 F_{193} - формирование $H = h$.
 F_{194} - формирование $H = -h$.
 A_{195} - вычисление $l = \text{entier} \left\lfloor \frac{x_{i+1} - x_{1,i+1}}{h} \right\rfloor$.
 F_{196} - формирование $j = 0$.
 P_{197} - проверка условия $d_{1,i+1} > y_{i+1}$.
 F_{198} - формирование $a_1 = -1$.
 F_{199} - формирование $a_1 = 1$.
 F_{200} - формирование $j = j + 1$.

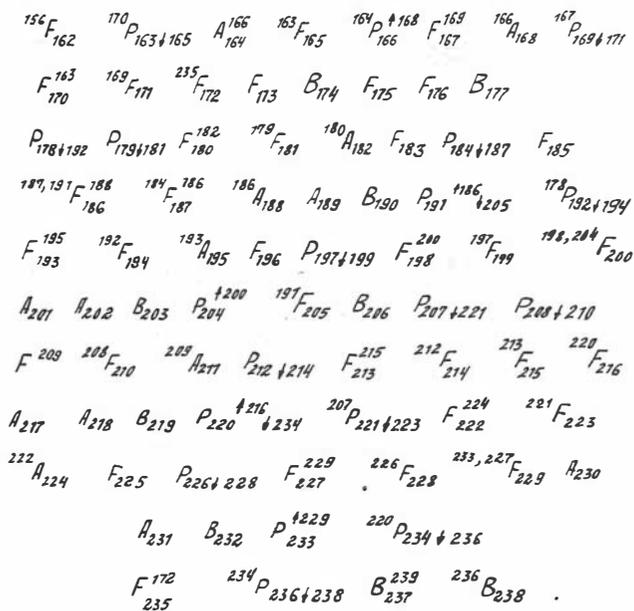
- A_{201} - вычисление $R = x_{1,i+1} + jH$.
 A_{202} - вычисление $f = a_1 \sqrt{z_{1,i+1} - (R - C_{1,i+1})^2} + d_{3,i+1}$.
 B_{203} - вывод на печать R, f .
 P_{204} - проверка условия $j < L$.
 F_{205} - формирование $R = x_{i+1}, f = y_{i+1}$.
 B_{206} - вывод на печать R, f .
 P_{207} - проверка условия $|y_{2,i+1}| \leq 1$.
 P_{208} - проверка условия $y_{i+1} < y_{2,i+1}$.
 F_{209} - формирование $H = h$.
 F_{210} - формирование $H = -h$.
 A_{211} - вычисление $l = \text{entiez} \left| \frac{y_{i+1} - y_{2,i+1}}{h} \right|$.
 P_{212} - проверка условия $C_{2,i+1} > x_{i+1}$.
 F_{213} - формирование $a_1 = -1$.
 F_{214} - формирование $a_1 = 1$.
 F_{215} - формирование $j = 0$.
 F_{216} - формирование $j = j + 1$.
 A_{217} - вычисление $f = y_{i+1} + jH$.
 A_{218} - вычисление $R = a_1 \sqrt{z_{2,i+1} - (f - d_{2,i+1})^2} + C_{2,i+1}$.
 B_{219} - вывод на печать R, f .
 P_{220} - проверка условия $j < L$.
 P_{221} - проверка условия $x_{i+1} < x_{2,i+1}$.
 F_{222} - формирование $H = h$.
 F_{223} - формирование $H = -h$.
 A_{224} - вычисление $l = \text{entiez} \left| \frac{x_{i+1} - x_{2,i+1}}{h} \right|$.
 F_{225} - формирование $j = 0$.
 P_{226} - проверка условия $d_{2,i+1} > y_{i+1}$.
 F_{227} - формирование $a_1 = -1$.
 F_{228} - формирование $a_1 = 1$.
 F_{229} - формирование $j = j + 1$.
 A_{230} - вычисление $R = x_{i+1} + jH$.
 A_{231} - вычисление $f = a_1 \sqrt{z_{2,i+1} - (R - C_{2,i+1})^2} + d_{2,i+1}$.
 B_{232} - вывод на печать R, f .
 P_{233} - проверка условия $j < L$.
 P_{234} - проверка условия $i < K$.
 F_{235} - формирование $i = i + 1$.

P_{236} - проверка условия $x_1 = x_N$ и $y_1 = y_N$.

B_{237} - вывод на печать $x_{2,N}$, $y_{2,N}$.

B_{238} - вывод на печать $x_{2,N-1}$, $y_{2,N-1}$.

Операторная схема блока F имеет вид:



Операторная схема работает следующим образом. Оператор F_{162} формирует число $i=1$ и передает управление оператору P_{163} . Группа операторов $P_{163} - A_{168}$ находит угловые коэффициенты $Y_{1, i+1}$, $Y_{2, i+1}$ прямых, первая из которых проходит через вспомогательную точку $(x_{1, i+1}, y_{1, i+1})$ и через центр $(C_{1, i+1}, d_{1, i+1})$ одной окружности, вторая через вспомогательную точку $(x_{2, i+1}, y_{2, i+1})$ и через центр $(C_{2, i+1}, d_{2, i+1})$ другой окружности. Управление передается оператору P_{169} . Если условие оператора P_{169} не

выполняется, то управление передается оператору F_{171} . В противном случае управление передается оператору F_{170} . Последний увеличивает число i на единицу и передает управление оператору P_{165} . Тем самым образуется цикл по i , пока не найдутся все упомянутые угловые коэффициенты.

Оператор F_{171} формирует число $i=1$ и передает управление оператору F_{172} . Операторы F_{172} , F_{173} находят координаты начальной точки первого линейного участка строящейся кривой. Происходит передача управления оператору B_{174} , который выводит найденные точки на печать. Управление передается оператору F_{175} . Операторы F_{175} , F_{176} находят координаты конечной точки вышеуказанного участка кривой. Управление передается оператору B_{177} , который выводит найденные точки на печать. Происходит передача оператору P_{178} . Если условие оператора P_{178} выполняется, то управление передается оператору P_{179} . Оператор P_{179} проверяет, какая из точек $(x_{1,i+1}, y_{1,i+1})$, (x_{i+1}, y_{i+1}) лежит выше на плоскости XOY . Если первая выше, то управление передается оператору F_{180} , который формирует отрицательный шаг $H=-h$. В противном случае оператор F_{181} формирует положительный шаг $H=h$. Управление передается оператору A_{182} , который находит количество точек, расположенных на куске кривой от $(x_{1,i+1}, y_{1,i+1})$ до (x_{i+1}, y_{i+1}) , ординаты которых меняются с шагом H . Группа операторов $F_{183}-P_{191}$ организует выдачу на печать этих точек. Заметим, что на указанном куске кривой первой точкой выводится точка касания $(x_{1,i+1}, y_{1,i+1})$.

В случае невыполнения условия оператора P_{178} , управление передается оператору P_{192} , который проверяет, какая из точек $(x_{1,i+1}, y_{1,i+1})$, (x_{i+1}, y_{i+1}) лежит левее по оси OX . Если первая левее, то управление передается на F_{193} , который формирует положительный шаг $H=h$, иначе управление передается оператору F_{194} и формируется отрицательный шаг $H=-h$. Затем работает оператор A_{195} и находит количество точек, расположенных на куске кривой от $(x_{1,i+1}, y_{1,i+1})$ до (x_{i+1}, y_{i+1}) , абсциссы которых меняются с шагом H . Группа операторов $F_{196}-P_{204}$ организует выдачу на печать этих точек. Как и в группе опе-

раторов $F_{183} - P_{191}$ первой точкой вводится точка $(x_{1,i+1}, y_{1,i+1})$.

Группа операторов $F_{205} - P_{233}$ аналогично операторам $F_{173} - P_{204}$ находит точки куска строящейся кривой от $(x_{1,i+1}; y_{1,i+1})$ до $(x_{2,i+1}; y_{2,i+1})$ и вводит их на печать, причем $(x_{1,i+1}, y_{1,i+1})$ вводится первой.

Итак, найден ряд точек кривой от $(x_{2,i}, y_{2,i})$ до $(x_{2,i+1}, y_{2,i+1})$ исключая последнюю. Она будет выведена в качестве первой при очередном значении i .

Оператор P_{234} проверяет, все ли такие участки кривой рассмотрены. Если не все, то управление передается на F_{235} , который увеличивает i на единицу и передает управление оператору F_{172} . Тем самым организовывается цикл.

Если условие оператора P_{234} не выполняется, то управление передается на P_{236} . Он проверяет замкнутость кривой. Если кривая замкнута, то работает оператор B_{237} . В противном случае работает оператор B_{238} . Тот или другой передает управление на A_{239} - конец задачи.

2. И н с т р у к ц и я к п р о г р а м м е "К о н т у р - 3"

Для счета на ЭВМ требуется сделать следующее.

1. Пробить алгоритмическую схему на перфокартн. После перфорации получим колоду из 172 перфокарт, причем на 172-ой пробивается контрольная сумма (нулевая).

2. Получить контрольную сумму ($K\Sigma$). Если $K\Sigma$ не совпадает, то колоду перфокарт с алгоритмической схемой следует продублировать и вновь получить $K\Sigma$.

3. После этого желательно получить программу на машинном языке. В этом случае колода перфокарт с алгоритмической схемой транслируется на машине и программа выдается на перфокартн.

4. Для счета после трансляции или по готовой программе нужно подготовить числа. Последние компануются и вводятся строго в следующем порядке.

- Перфокарта № 1, на которой пробивается число N_1 : 1-ая строка - величина N_1 , 2-ая строка - $K\Sigma$.
- Перфокарты № 2(a) - 2(...), на которых пробивается массив $x(x_1, x_2, \dots, x_{N_1})$: 1-ая строка - число x_1 , 2-ая строка - число x_2 и т.д. Очевидно, что если $N_1 \leq 12$, то будем иметь только одну перфокарту № 2(a) из совокупности № 2(a) - 2(...).
- Перфокарта № 3 с контрольной суммой массива x .
- Перфокарты № 4(a) - 4(...), на которых пробивается массив $y(y_1, y_2, \dots, y_{N_1})$: 1-ая строка - число y_1 , 2-ая строка - число y_2 и т.д.
- Перфокарта № 5 с контрольной суммой массива y .
- Перфокарты № 6(a) - 6(...), на которых пробивается массив $\delta(\delta_1, \delta_2, \dots, \delta_{N_1})$: 1-ая строка - число δ_1 , 2-ая строка - число δ_2 и т.д.
- Перфокарта № 7 с контрольной суммой массива δ .
- Перфокарта № 8; пробивается число h : 1-ая строка - величина h , 2-ая строка - $K\Sigma$.
- Перфокарта № 9; пробивается число ε : 1-ая строка - величина ε , 2-ая строка - $K\Sigma$.

Заметим, что все пробивается в десятичном коде.

В результате счета выдаются новые координаты точек линии в виде последовательности

$$\tilde{x}_1, \tilde{y}_1, \tilde{x}_2, \tilde{y}_2, \tilde{x}_3, \tilde{y}_3, \dots, \tilde{x}_j, \tilde{y}_j,$$

где \tilde{x}_j, \tilde{y}_j - координаты последней выводимой на печать точки. При этом в качестве первой точки \tilde{x}_1, \tilde{y}_1 новой таблицы выдается начальная точка первого линейного участка - случай замкнутой линии; для незамкнутой первая и последняя точки новой таблицы совпадают с первой и последней точками старой таблицы.

$$x_1 = \tilde{x}_1; \quad y_1 = \tilde{y}_1; \quad x_{N_1} = \tilde{x}_j; \quad y_{N_1} = \tilde{y}_j.$$

Особое внимание нужно обратить на то, что при вводе исходных данных в случае замкнутой линии должно выполняться условие $x_1 = x_{N_1}$, $y_1 = y_{N_1}$.

ТАБЛИЦА РАСПРЕДЕЛЕНИЯ ПАМЯТИ

	НАЧАЛО	КОНЕЦ	ДЛИНА	ПРИМЕЧАНИЯ
ПРОГРАММА	0063	4162	4100	
ЧИСЛОВЫЕ КОНСТАНТЫ	4163	4205	0023	
ФОРМИРУЕМЫЕ КОНСТАНТЫ	4206	4612	0405	
СКАЛЯРЫ	4613	4677	0065	
РАБОЧИЕ ЯЧЕЙКИ	4700	4713	0014	
МАССИВЫ				
1	xxxx	xxxx	xxxx	НЕ РАСПРЕДЕЛЕН
2	xxxx	xxxx	xxxx	НЕ РАСПРЕДЕЛЕН
3	xxxx	xxxx	xxxx	НЕ РАСПРЕДЕЛЕН
4	xxxx	xxxx	xxxx	НЕ РАСПРЕДЕЛЕН
..
30	xxxx	xxxx	xxxx	НЕ РАСПРЕДЕЛЕН
31	xxxx	xxxx	xxxx	НЕ РАСПРЕДЕЛЕН
32	xxxx	xxxx	xxxx	НЕ РАСПРЕДЕЛЕН
ПЕРВОЕ РАБОЧЕЕ ПОЛЕ ИС	НАЧАЛО	КОНЕЦ	ДЛИНА	
	4714	7303	2370	
РАБОЧЕЕ ПОЛЕ ИС	НАЧАЛО	КОНЕЦ	ДЛИНА	
	7304	7500	0174	

АДРЕСА ПРОГРАММНЫХ ОСТАНОВК

Алгоритмическая программа транслируется ≈ 15 минут (М-20, М-220). Время счета зависит от количества точек, определяющих ломаную линию, а также и от h . Чем меньше h , тем больше время счета. В наших примерах счет длился порядка 5 сек.

Т е с т о в ы е п р и м е р ы

А. Случай замкнутой ломаной.

Исходные данные следующие:

1. Число $N_1 = 11$;
2. Координаты (x_i, y_i) , $i = 1, \dots, 11$, первоначальных точек ломаной, которые в нашем случае заданы в виде отдельных массивов $x(x_1, x_2, \dots, x_n)$, $y(y_1, y_2, \dots, y_n)$.

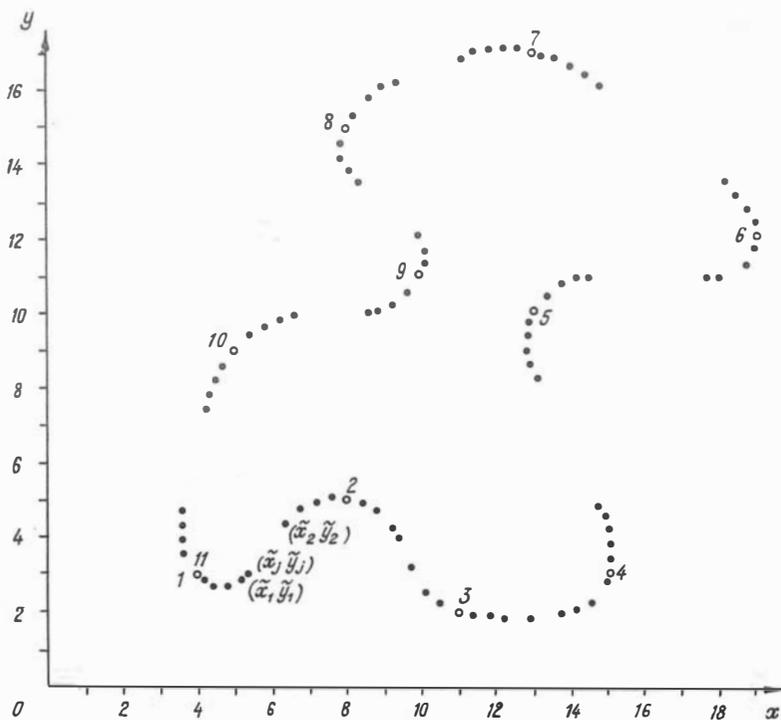


Рис. 16

i	1	2	3	4	5	6	7	8	9	10	11
x_i	4	8	11	15	13	19	13	8	10	5	4
y_i	3	5	2	3	10	12	17	15	11	9	3

3. Массив $\delta(\delta_1, \delta_2, \dots, \delta_n)$:

i	1	2	3	4	5	6	7	8	9	10	11
δ_i	1	1	1	1	1	1	1	1	1	1	1

4. Шаг $h = 0,4$;

5. Величина $\varepsilon = 0,1$.

В результате счета были получены новые точки (x_j, y_j) , $j = 1, 2, \dots$, в виде таблицы А. Если нанести эти точки на плоскость (x, y) , то получим рис. (16).

Б. Случай незамкнутой ломаной.

Исходные данные: $N_j = 11$;

i	1	2	3	4	5	6	7	8	9	10	11
x_i	69	64	70	80	90	100	110	120	125	120	110
y_i	0	10	18	15	17	7	13	16	10	8	0
δ_i	4	4	4	4	4	4	4	4	4	4	4

$h = 0,5$; $\varepsilon = 0,5$

В результате счета была получена таблица Б.

По данным этой таблицы построен рис. (17).

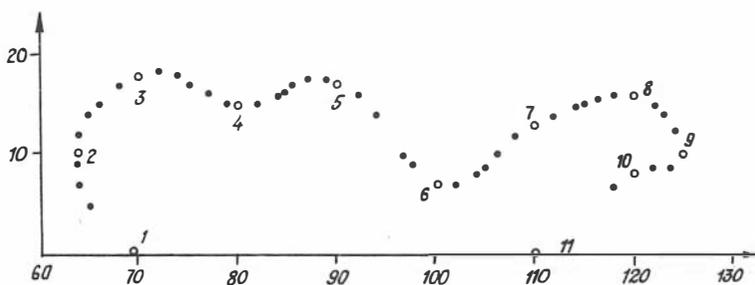


Рис.17

Таблица
Тестовая выдача на печать (пример А;

X (2 столбец)	Y (4 столбец)
+++01523377842	+++01297351465
+++01677627209	+++01472749299
+++01757627209	+++01500940881
+++01800000000	+++01500000000
+++01880000000	+++01466787836
+++01936596774	+++01393196638
+++02100624578	+++01257420209
+++02108624578	+++01204459639
+++02114000000	+++01190296829
+++02122000000	+++01180757024
+++02129556427	+++01183357421
+++02141633456	+++01202496790
+++02149633456	+++01287703628
...
+++01960000000	+++02104753828
+++01880000000	+++02100646799
+++01659702524	+++01987707571
+++01579702524	+++01965916524
+++01500000000	+++01900000000
+++01451641648	+++01820000000
+++01422173956	+++01740000000
+++01362162110	+++01477513285
+++01359190174	+++01397513285
+++01388157976	+++01317513285
+++01440000000	+++01272002424
+++01520000000	+++01293493434

читать слева направо по строке)

X (6 столбец)

+++01637627209

+++01717627209

+++01797627208

+++01840000000

+++01920000000

+++01966245781

+++02104624578

+++02110000000

+++02118000000

+++02126000000

+++02137633456

+++02145633456

+++02150000000

... ..

+++01920000000

+++01855576116

+++01619702524

+++01539702524

+++01473095857

+++01434840296

+++01417837889

+++01356933317

+++01369181386

+++01400000000

+++01480000000

+++01523377842

Y (8 столбец)

+++01436790670

+++01492287371

+++01500322651

+++01489308569

+++01424500185

+++01320133339

+++01224161057

+++01200000000

+++01183924110

+++01180735873

+++01192156205

+++01227743642

+++01300000000

... ..

+++02102049303

+++02100284071

+++01980877278

+++01940732025

+++01860000000

+++01780000000

+++01722486714

+++01437513285

+++01357513285

+++01300000000

+++01270562530

+++01297351465

Таблица
Тестовая выдача на печать (пример Б;

X (2 столбец)	Y (4 столбец)
+++02690000000	+-00000000000
+++02649602028	+++01594115281
+++02644837378	+++01694115281
+++02641765589	+++01794115281
+++02640176122	+++01894115281
+++02639969673	+++01994115280
+++02640427016	+++02105000000
+++02642195316	+++02115000000
+++02645256591	+++02125000000
+++02649769847	+++02134999999
+++02656022123	+++02145000000
+++02661393099	+++02152354319
+++02671393099	+++02163531098
...
+++03123000000	+++02141130692
+++03124211607	+++02124375919
+++03124743873	+++02114375919
+++03124976934	+++02104375919
+++03124500000	+++01896831195
+++03123654096	+++01868599785
+++03121966085	+++01869711824
+++03120966086	+++01847996427
+++03120000000	+++01800000000
+++03119000000	+++01730111570
+++03118000000	+++01653353728
+++03110000000	+-00000000000

читать слева направо по строке)

X (6 столбец)

Y (8 столбец)

+++02652738701

+++01544115281

+++02646988207

+++01644115281

+++02643107003

+++01744115280

+++02640794522

+++01844115281

+++02639902459

+++01944115281

+++02640000000

+++02100000000

+++02641156108

+++02110000000

+++02643556602

+++02120000000

+++02647317697

+++02130000000

+++02652653141

+++02140000000

+++02656131652

+++02145150314

+++02666393099

+++02158498593

+++02676393100

+++02167731825

... ..

... ..

+++03123058071

+++02140343280

+++03124521543

+++02119375919

+++03124893031

+++02109375919

+++03125000000

+++02100000000

+++03124000000

+++01872377100

+++03122466085

+++01871449894

+++03121466085

+++01852008111

+++03120466085

+++01826982485

+++03119500000

+++01765873450

+++03118500000

+++01692634705

+++03117837150

+++01640154044

3. ПРОГРАММА "КОНТУР - 3" (на Альфа - яэнке)

Лист 1

1 начало веществе $\pi, h, H, \varepsilon, a_1, \sigma_1, a_2, \sigma_2, f, R$;
 2 целое i, j, N, N_1, K, l ; ввод (N_1) ; начало
 3 веществе массиве $y, x, y_1, x_1, z_1, u_1, y_2, x_2, z_2, u_2, A,$
 4 $X_1, X_2, Y_1, Y_2, a_3, u_3, a_4, y_4, \delta, \alpha, k, s_1, s_2, S_1, S_2 [1 : N_1]$;
 5 ввод ($x, y, \delta, h, \varepsilon$) ;
 6. $\pi := 3.14159265$; $N := N_1$;
 7 если $x[1] = x[N]$ и $y[1] = y[N]$ то $\{ X_1[N-1] := x[N] -$
 8 $x[N-1]$; $Y_1[N-1] := y[N] - y[N-1]$; $X_2[N-1] := x[2] -$
 9 $x[N-1]$; $Y_2[N-1] := y[2] - y[N-1]$;
 10 если $abs (X_1[N-1] \times Y_2[N-1] - X_2[N-1] \times Y_1[N-1]) \leq \varepsilon$

Лист 2

88

11 то для $i := 1, \dots, N-2$ цикл $\{ x[i] := x[i+1]$; $y[i] := y[i+1]$
 12 $\}$; $x[N-1] := x[1]$; $N := N-1$ $\}$;
 13 $i := 1$; ВВ : $X_1[i] := x[i+1] - x[i]$; $Y_1[i] := y[i+1] - y[i]$;
 14 $X_2[i] := x[i+2] - x[i]$; $Y_2[i] := y[i+2] - y[i]$;
 15 если $abs (X_1[i] \times Y_2[i] - X_2[i] \times Y_1[i]) \leq \varepsilon$ то $\{$
 16 для $j := i+1, \dots, N-1$ цикл $\{ x[j] := x[j+1]$; $y[j] := y[j+1]$ $\}$;
 17 $N := N-1$; на ВВ $\}$ иначе если $i < N-2$ то $\{ i := i+1$; на ВВ $\}$;
 18 для $i := 1, \dots, N-1$ цикл $\{ X_1[i] := sqrt ((x[i+1] - x[i]$
 19 $)^2 + (y[i+1] - y[i])^2)$; если $\delta[i] > X_1[i] / 4$ то $\{$
 20 $\delta[i] := X_1[i] / 4$; $\delta[i+1] := X_1[i] / 4$ $\}$ $\}$;

21 $\text{dim } i := 1, \dots, N-2$ $\text{umkl} \{ \text{eclm } x[i+2] \neq x[i] \text{ mo}$
 22 $\alpha[i+1] := \text{arctg}((y[i+2] - y[i]) / (x[i+2] - x[i]))$ umkl
 23 $\alpha[i+1] := \pi/2$; $x1[i+1] := x[i+1] - \delta[i+1] \times \cos(\alpha[i+1])$;
 24 $y1[i+1] := y[i+1] - \delta[i+1] \times \sin(\alpha[i+1])$; $x2[i+1] := x[i+1] +$
 25 $\delta[i+1] \times \cos(\alpha[i+1])$; $y2[i+1] := y[i+1] + \delta[i+1] \times \sin(\alpha[i+1]$
 26 $); \text{eclm } x[i+2] \neq x[i] \text{ mo } \{ \text{eclm } y[i+2] \neq y[i] \text{ mo } \{$
 27 $k[i+1] := (y[i+2] - y[i]) / (x[i+2] - x[i])$;
 28 $x3[i+1] := (x1[i+1] + k[i+1]) \times (k[i+1] \times x[i] + y1[i+1] -$
 29 $y[i]) / (k[i+1] + 2)$; $y3[i+1] := y[i] + k[i+1] \times$
 30 $(x3[i+1] - x[i])$; $x4[i+1] := (x2[i+1] + k[i+1]) \times$

31 $(k[i+1] \times x[i] + y2[i+1] - y[i]) / (k[i+1] + 2)$;
 32 $y4[i+1] := y[i] + k[i+1] \times (x4[i+1] - x[i])$ $\}$ umkl
 33 $x3[i+1] := x1[i+1]$; $y3[i+1] := y[i]$; $x4[i+1] := x2[i+1]$;
 34 $y4[i+1] := y[i]$ $\}$ umkl $\{ x3[i+1] := x[i]$; $y3[i+1] := y1[i+1]$;
 35 $x4[i+1] := x[i]$; $y4[i+1] := y2[i+1]$;
 36 $sl[i] := \text{sqrt}((x[i] - x3[i+1]) + (y[i] - y3[i+1]) + 2)$;
 37 $se[i] := \text{sqrt}((x[i] - x4[i+1]) + 2 + (y[i] - y4[i+1]) + 2)$;
 38 $X1[i+1] := x[i+1] - \alpha[i]$; $X2[i+1] := x[i+2] - \alpha[i]$;
 39 $Y1[i+1] := y[i+1] - y[i]$; $Y2[i+1] := y[i+2] - y[i]$;
 40 $A[i] := (X1[i+1] \times X2[i+1] + Y1[i+1] \times Y2[i+1]) /$

41 $sqzt((X1[i+1]+2 + Y1[i+1]+2) * (X2[i+1]+2 + Y2[i+1]+2))$;
 42 $S1[i] := sqzt((x[i+2] - x3[i+1]+2 + (y[i+2] - y3[i+1]+2))$;
 43 $S2[i] := sqzt((x[i+2] - x4[i+1]+2 + (y[i+2] - y4[i+1]+2))$;
 44 если $A[i] > 0$ то {если $S1[i] < S2[i]$ то { $z1[i+1] :=$
 45 $x1[i+1]$; $u1[i+1] := y1[i+1]$; $z2[i+1] := x2[i+1]$;
 46 $u2[i+1] := y2[i+1]$ } иначе { $z1[i+1] := x2[i+1]$;
 47 $u1[i+1] := y2[i+1]$; $z2[i+1] := x1[i+1]$; $u2[i+1] := y1[i+1]$ } }
 48 иначе {если $S1[i] < S2[i]$ то { $z1[i+1] := x2[i+1]$;
 49 $u1[i+1] := y2[i+1]$; $z2[i+1] := x1[i+1]$; $u2[i+1] := y1[i+1]$ }
 50 иначе { $z1[i+1] := x1[i+1]$; $u1[i+1] := y1[i+1]$;

51 $z2[i+1] := x2[i+1]$; $u2[i+1] := y2[i+1]$ } } } ;
 52 если $x[1] = x[N]$ & $y[1] = y[N]$ то {если
 53 $x[N-1] = x[2]$ то $\alpha[1] := \pi/2$ иначе { $k[1] :=$
 54 $(y[N-1] - y[2]) / (x[N-1] - x[2])$; $\alpha[1] := arctg(k[1])$ } ;
 55 $x1[1] := x[1] - \delta[1] * \cos(\alpha[1])$; $y1[1] := y[1] - \delta[1] *$
 56 $\sin(\alpha[1])$; $x2[1] := x[1] + \delta[1] * \cos(\alpha[1])$;
 57 $y2[1] := y[1] + \delta[1] * \sin(\alpha[1])$;
 58 если $x[N-1] = x[2]$ то { $x3[1] := x[2]$; $y3[1] := y1[1]$;
 59 $x4[1] := x[2]$; $y4[1] := y2[1]$ } иначе {если $y[N-1] = y[2]$
 60 то { $x3[1] := x1[1]$; $y3[1] := y[N-1]$; $x4[1] := x2[1]$;

61 $y_4[1] := y[N-1]$ } иначе { $x_3[1] := (x_1[1] + k[1]) * (x_2[1] * k[1] +$
 62 $y_1[1] - y[2]) / (k[1] + 2 + 1)$; $y_3[1] := y[2] + k[1] * (x_3[1] -$
 63 $x[2])$; $x_4[1] := (x_2[1] + k[1] * (x[2] * k[1] + y_2[1] - y[2]))$
 64 $/(k[1] + 2 + 1)$; $y_4[1] := y[2] + k[1] * (x_4[1] - x[2])$ } } ;
 65 $s_1[N-1] := \text{sqrt}((x[N-1] - x_3[1])^2 + (y[N-1] - y_3[1])^2)$;
 66 $s_2[N-1] := \text{sqrt}((x[N-1] - x_4[1])^2 + (y[N-1] - y_4[1])^2)$;
 67 $S_1[N-1] := \text{sqrt}((x[2] - x_3[1])^2 + (y[2] - y_3[1])^2)$;
 68 $S_2[N-1] := \text{sqrt}((x[2] - x_4[1])^2 + (y[2] - y_4[1])^2)$;
 69 $X_1[N] := x[1] - x[N-1]$; $X_2[N] := x[2] - x[N-1]$;
 70 $Y_1[N] := y[1] - y[N-1]$; $Y_2[N] := y[2] - y[N-1]$;

71 $A[N-1] := (X_1[N] * X_2[N] + Y_1[N] * Y_2[N]) /$
 72 $\text{sqrt}((X_1[N]^2 + Y_1[N]^2) * (X_2[N]^2 + Y_2[N]^2))$;
 73 если $A[N-1] > 0$ то { если $s_1[N-1] < s_2[N-1]$ то {
 74 $z_1[N] := z_1[1] := x_1[1]$; $u_1[N] := u_1[1] := y_1[1]$;
 75 $z_2[N] := z_2[1] := x_2[1]$; $u_2[N] := u_2[1] := y_2[1]$ } иначе {
 76 $z_1[N] := z_1[1] := x_2[1]$; $z_2[N] := z_2[1] := x_1[1]$;
 77 $u_1[N] := u_1[1] := y_2[1]$; $u_2[N] := u_2[1] := y_1[1]$ } } иначе {
 78 если $S_1[N-1] < S_2[N-1]$ то { $z_1[N] := z_1[1] := x_2[1]$;
 79 $z_2[N] := z_2[1] := x_1[1]$; $u_1[N] := u_1[1] := y_2[1]$;
 80 $u_2[N] := u_2[1] := y_1[1]$ } иначе { $z_1[N] := z_1[1] := x_1[1]$;

81 $x_1[N] := x_1[i+1] := y_1[i+1]; \quad z_2[N] := z_2[i+1] := x_2[i+1];$
 82 $x_2[N] := x_2[i+1] := y_2[i+1] \}$ иначе { $x_1[i+1] := z_2[i+1] := x[i+1];$
 83 $x_1[i+1] := x_2[i+1] := y[i+1]; \quad z_1[N] := z_2[N] := x[N];$
 84 $x_1[N] := x_2[N] := y[N] \}$;
 85 для $i := 1, \dots, N-1$ цикл { если $x_1[i+1] = z_2[i]$ то
 86 $\alpha[i] := \pi/2$ иначе $\alpha[i] := \arctg((x_1[i+1] - u_2[i])/$
 87 $(x_1[i+1] - z_2[i]))$ };
 88 для $i := 1, \dots, N-1$ цикл { $a_1 := z_2[i] + \delta[i] \times \cos(\alpha[i]);$
 89 $s_1 := u_2[i] + \delta[i] \times \sin(\alpha[i]); \quad a_2 := z_2[i] - \delta[i] \times \cos(\alpha[i]);$
 90 $s_2 := u_2[i] - \delta[i] \times \sin(\alpha[i]);$

91 $s_1[i+1] := \text{sqrt}((z_1[i+1] - a_1)^2 + (u_1[i+1] - s_1)^2);$
 92 $s_2[i+1] := \text{sqrt}((z_1[i+1] - a_2)^2 + (u_1[i+1] - s_2)^2);$
 93 если $s_1[i+1] > s_2[i+1]$ то { $x_2[i] := a_2; \quad y_2[i] := s_2$ } иначе
 94 { $x_2[i] := a_1; \quad y_2[i] := s_1$ }; $a_1 := z_1[i+1] + \delta[i+1] \times \cos(\alpha[i]);$
 95 $s_1 := u_1[i+1] + \delta[i+1] \times \sin(\alpha[i]); \quad a_2 := z_1[i+1] - \delta[i+1] \times \cos(\alpha[i]);$
 96 $s_2 := u_1[i+1] - \delta[i+1] \times \sin(\alpha[i]); \quad s_1[i+1] := \text{sqrt}(($
 97 $z_2[i] - a_1)^2 + (u_2[i] - s_1)^2); \quad s_2[i+1] := \text{sqrt}((z_2[i] - a_2)^2 +$
 98 $(u_2[i] - s_2)^2); \quad \text{если } s_1[i+1] > s_2[i+1]$ то { $x_1[i+1] := a_2;$
 99 $y_1[i+1] := s_2$ } иначе { $x_1[i+1] := a_1; \quad y_1[i+1] := s_1$ }};
 100 для $i := 1, \dots, N-2$ цикл { если $x_1[i+1] \neq u_2[i]$ то {

101 $X_1[i] := (x_1[i+1] - z_2[i]) / (u_1[i+1] - u_2[i]);$
 102 если $y[i+2] \neq y[i]$ то $\{X_2[i] := (x[i+2] - x[i]) / (y[i+2] -$
 103 $y[i]); x_3[i+1] := (X_2[i] \times x[i+1] + y[i+1] - X_1[i]) \times x_1[i+1]$
 104 $- y_1[i+1]) / (X_2[i] - X_1[i])\}$ иначе $x_3[i+1] := x[i+1];$
 105 $y_3[i+1] := y_1[i+1] - X_1[i] \times (x_3[i+1] - x_1[i+1])\}$ иначе
 106 $\{x_3[i+1] := x_1[i+1]; X_2[i] := (x[i+2] - x[i]) / (y[i+2] - y[i]);$
 107 $y_3[i+1] := y[i+1] - X_2[i] \times (x_3[i+1] - x[i+1])\};$
 108 $s_1[i+1] := (x[i+1] - x_3[i+1]) \div 2 + (y[i+1] - y_3[i+1]) \div 2\};$
 109 для $i := 1, \dots, N-2$ цикл $\{\text{если } y[i+2] \neq y[i]$ то
 110 $\{X_2[i] := (x[i+2] - x[i]) / (y[i+2] - y[i]);$

111 если $u_1[i+2] \neq u_2[i+2]$ то $\{X_1[i] := (z_1[i+2] - x_2[i+1]) / ($
 112 $u_1[i+2] - u_2[i+1]); x_4[i+1] := (X_1[i] \times x_2[i+1] + y_2[i+1] -$
 113 $X_2[i] \times x[i+1] - y[i+1]) / (X_1[i] - X_2[i])\}$ иначе
 114 $x_4[i+1] := x_2[i+1]; y_4[i+1] := y[i+1] - X_2[i] \times (x_4[i+1] - x[i+1])\}$
 115 иначе $\{x_4[i+1] := x[i+1]; X_1[i] := (x_1[i+2] - z_2[i+1]) / (u_1[i+2]$
 116 $- u_2[i+1]); y_4[i+1] := y_2[i+1] - X_1[i] \times (x_4[i+1] - x_2[i+1])\};$
 117 $s_2[i+1] := (x[i+1] - x_4[i+1]) \div 2 + (y[i+1] - y_4[i+1]) \div 2\};$
 118 если $x[i] = x[N]$ и $y[i] = y[N]$ то $\{\text{если } y[N-1] \neq y[2]$
 119 то $\{a_1 := (x[N-1] - x[2]) / (y[N-1] - y[2]); \text{если}$
 120 $u_1[N] \neq u_2[N-1]$ то $\{a_1 := (z_1[N] - z_2[N-1]) / (u_1[N] -$

121 $u_2 [N-1]); \quad x_3 [N] := (e_1 \times x_1 [N] + y_1 [N] - a_1 \times x [1] - y [1]) /$
 122 $(e_1 - a_1) \}$ иначе $x_3 [N] := x_1 [N];$
 123 $y_3 [N] := y [1] - a_1 \times (x_3 [N] - x [1]); \}$ иначе $\{ x_3 [N] :=$
 124 $x [1]; \quad e_1 := (z_1 [N] - z_2 [N-1]) / (u_1 [N] - u_2 [N-1]);$
 125 $y_3 [N] := y_1 [N] - e_1 \times (x_3 [N] - x_1 [N]); \}$;
 126 если $y [N-1] \neq y [2]$ то $\{ \text{если}$ $u_2 [1] \neq u_1 [2]$ то $\{$
 127 $e_1 := (z_2 [1] - z_1 [2]) / (u_2 [1] - u_1 [2]); \quad x_4 [N] :=$
 128 $(e_1 \times x_2 [1] + y_2 [1] - a_1 \times x [1] - y [1]) / (e_1 - a_1) \}$ иначе
 129 $x_4 [N] := x_2 [1]; \quad y_4 [N] := y [1] - a_1 \times (x_4 [N] - x [1]) \}$ иначе
 130 $\{ x_4 [N] := x [1]; \quad e_1 := (z_2 [1] - z_1 [2]) / (u_2 [1] - u_1 [2]);$

131 $y_4 [N] := y_2 [1] - e_1 \times (x_4 [N] - x_2 [1]); \}$;
 132 $s_1 [N] := (x [N] - x_3 [N]) \uparrow_2 + (y [N] - y_3 [N]) \uparrow_2;$
 133 $s_2 [N] := (x [N] - x_4 [N]) \uparrow_2 + (y [N] - y_4 [N]) \uparrow_2; \quad x_2 [N] :=$
 134 $x_2 [1]; \quad z_2 [N] := z_2 [1]; \quad u_2 [N] := u_2 [1]; \quad y_2 [N] := y_2 [1];$
 135 $K := N-1 \}$ иначе $\{ x_2 [1] := x [1]; \quad y_2 [1] := y [1]; \quad x_1 [N] :=$
 136 $x [N]; \quad y_1 [N] := y [N]; \quad K := N-2 \}$;
 137 для $i := 1, \dots, K.$ цикл $\{ \text{если}$ $x_3 [i+1] \neq z_1 [i+1]$ то
 138 $y_1 [i+1] := (y_3 [i+1] - u_1 [i+1]) / (x_3 [i+1] - z_1 [i+1])$ иначе
 139 $y_1 [i+1] := 10 \uparrow_5; \quad \text{если}$ $x_4 [i+1] \neq z_2 [i+1]$ то $y_2 [i+1] :=$
 140 $(y_4 [i+1] - u_2 [i+1]) / (x_4 [i+1] - z_2 [i+1])$ иначе $y_2 [i+1] := 10 \uparrow_5 \}$;

161 если $j < l$ то на KB }} иначе {если $x[i+1] < x_2[i+1]$
162 то $H := h$ иначе $H := -h$; $l := \text{entier}(a \cos(x[i+1] -$
163 $x_2[i+1]) / h)$; $j := 0$;
163' если $y_4[i+1] = y[i+1]$ то {если $y_2[i+1] > y[i+1]$ то $a_1 := 1$ иначе $a_1 := -1$ }
164 иначе {если $y_4[i+1] > y[i+1]$ то $a_1 := -1$ иначе $a_1 := 1$ }; если $l \neq 0$ то {
165 $BC : j := j + 1$; $R := x[i+1] + j \times H$; $f := a_1 \times \text{sqrt}(s_2[i+1] - (R -$
166 $x_4[i+1])^2) + y_4[i+1]$; вывод (R, f) ;
167 если $j < l$ то на BC }}};
168 если $x[l] = x[N]$ & $y[l] = y[N]$ то { $R := x_2[N]$;
169 $f := y_2[N]$; вывод (R, f) } иначе { $R := x_2[N-1]$;
170 $f := y_2[N-1]$; вывод (R, f) ; $R := x[N]$; $f := y[N]$;

171 вывод (R, f) }} конец *

§4. МЕТОД ПОСТРОЕНИЯ КОНТУРНОЙ ЛОМАННОЙ ЛИНИИ

Для изображения с помощью графопостроителя графика в виде ломаной линии, состоящей из отрезков прямых, достаточно иметь метод построения отрезка прямой, соединяющего две заданные точки. Графопостроители с шагово-импульсными при-

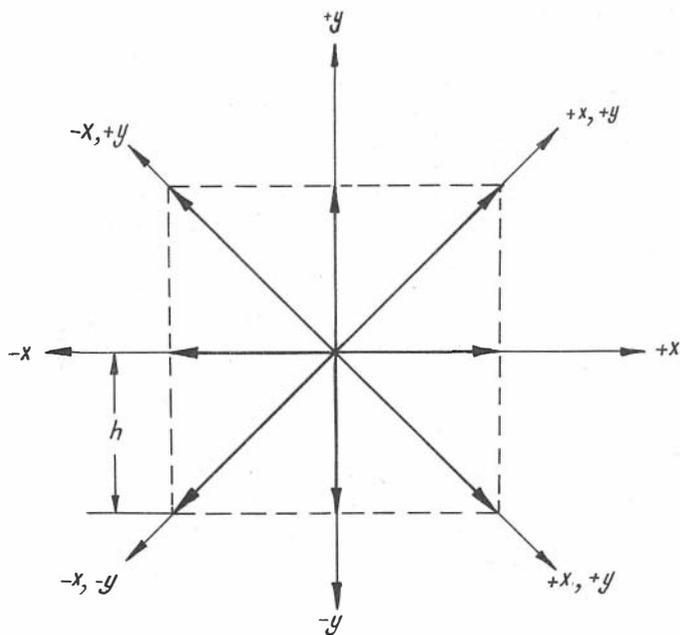


Рис. 18

водами выполняют единичные перемещения (шаги) величиной h по осям x и y или под углом 45° к этим осям. Построение с помощью графопостроителя отрезка прямой сводится к аппроксимации этого отрезка ломаной, состоящей из отрезков прямых, соответствующих элементарным единичным перемещениям шагово-импульсного привода. При этом необходимо определить

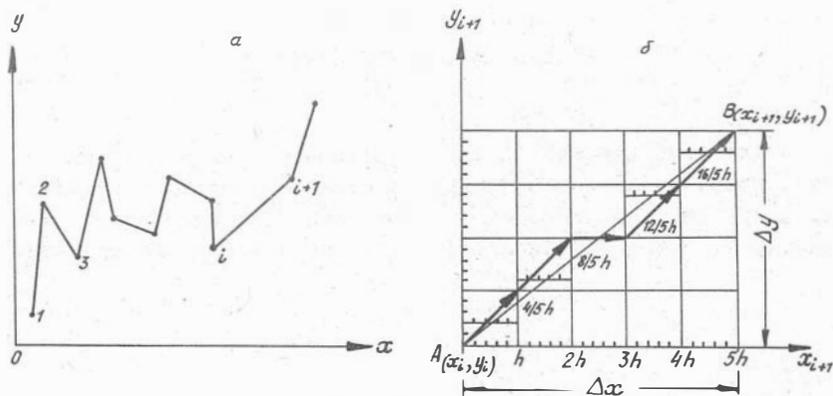


Рис. 19

последовательность перемещений, каждое из которых должно обеспечивать наименьшее отклонение от заданной прямой. За величину, характеризующую степень приближения, принимаем половину размера шага h .

Координаты точек x_i , y_i графика выразим в числах, кратных величине шага h .

$$x_i = E \left(\frac{|X_i|}{h} + 0,5 \right) \cdot \frac{X_i}{|X_i|},$$

$$y_i = E \left(\frac{|Y_i|}{h} + 0,5 \right) \cdot \frac{Y_i}{|Y_i|}.$$

Тогда приращения координат при переходе от одной точки к другой будут целочисленными

$$\Delta x_i = x_{i+1} - x_i,$$

$$\Delta y_i = y_{i+1} - y_i.$$

Единичные перемещения могут быть или вдоль какой-либо из осей, или одновременно по оси x и y (комбинированное перемещение). Обозначим операторы этих перемещений следующим образом: h^x и h^{-x} - для перемещений вдоль оси x в обоих направлениях, h^y и h^{-y} - для перемещений вдоль оси y в обоих направлениях, $h^x h^y$, $h^{-x} h^y$, $h^x h^{-y}$, $h^{-x} h^{-y}$ - для единичных комбинированных перемещений.

Введем следующие обозначения:

$$\lambda^y = \frac{|\Delta y_i|}{h} \quad - \text{ полное число шагов по } y;$$

$$\lambda^x = \frac{|\Delta x_i|}{h} \quad - \text{ полное число шагов по } x,$$

n - текущее число шагов по x ,

m - текущее число шагов по y .

Уравнение прямой, проходящей через точки i , $i+1$ в системе координат xOy имеет вид

$$y = \alpha x, \quad \alpha = \left| \frac{\Delta y_i}{\Delta x_i} \right|.$$

Так как при $\alpha > 1$ при единичном перемещении по оси y и при одном комбинированном перемещении приращение по y одинаково, а по x разное, будем рассматривать отклонения от прямой $y = \alpha x$ по x . Аналогично, при $\alpha < 1$ рассматриваем отклонение от прямой $y = \alpha x$ по y .

Рассмотрим вначале случай, когда $\alpha > 1$, т.е. мы можем совершать либо комбинированные перемещения, либо единичные перемещения вдоль оси y . Допустим, нужно сделать первый шаг (рис.20а). Если сделано комбинированное перемещение, расстояние от конца отрезка OO' до прямой $y = \alpha x$: $O'C' < \frac{1}{2}h$, а расстояние от конца отрезка OB (единичного перемещения) $BC' > \frac{1}{2}h$. Следовательно, нужно сделать комбинированное перемещение OO . Если вначале $m = n = 0$, то теперь $m = n = 1$. Перейдем в систему координат $x'O'y'$, полученную из xOy

переносом начала в т. O' . Если сделать комбинированное перемещение, то $CD > \frac{1}{2}h$, а если сделать перемещение вдоль оси y , $CB' < \frac{1}{2}h$, значит нужно сделать перемещение вдоль оси y . Теперь $m=2$, а $n=1$, т.к. перемещений

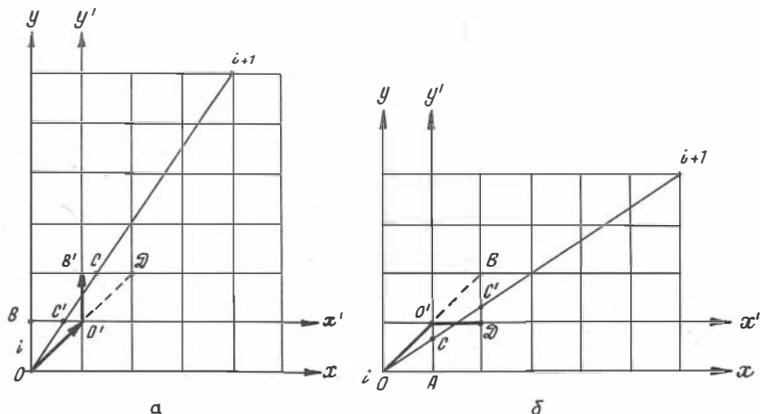


Рис. 20

по x не производилось. Переходя в систему координат с началом в точке B' и продолжая подобные рассуждения, снова сделаем некоторое перемещение. Описанный процесс повторяется до тех пор, пока не придем в точку $i+1$.

Аналогичные рассуждения проводятся, когда $\alpha < 1$.

В этом случае можно производить либо комбинированные перемещения, либо перемещения вдоль оси x . Допустим, нужно сделать первый шаг. Если сделано комбинированное перемещение OO' , расстояние от точки O до прямой $y = \alpha x \div O'C < \frac{1}{2}h$. Если сделано единичное перемещение вдоль оси x , то расстояние от точки A до прямой $y = \alpha x \div AC > \frac{1}{2}h$. Следовательно, целесообразнее сделать комбинированное перемещение. Если вначале $m=n=0$, то теперь $m=n=1$. Перейдем в систему координат с началом в т. $O' - x'o'y'$. Если сделать комбинированное перемещение, то $BC' > \frac{1}{2}h$, а если единичное - то $DC' < \frac{1}{2}h$, значит, в этом случае лучше сделать единичное перемещение $O'D$ (рис 20б). Теперь $n=2$, $m=1$, т.к. перемещений по y не производилось. Перейдем в новую систему координат с началом в т. D .

Продолжая аналогичные рассуждения, снова сделаем некоторое перемещение. Описанный процесс повторяется до тех пор, пока не придем в точку $i+1$.

Приведем аналогичные рассуждения для общего случая, считая вначале $\alpha > 1$. Уравнение данной прямой, проходящей через точки i и $i+1$, в системе координат xoy с центром в точке i , $y = \alpha x$.

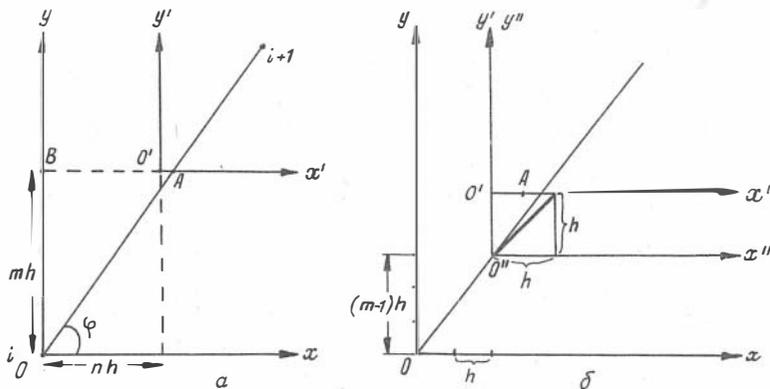


Рис. 20

Рассмотрим систему координат $x'o'y'$, смещенную в направлении оси y на расстояние mh и по оси x на расстояние nh . Отрезок $O'A$, отсекаемый прямой $y = \alpha x$ на оси $O'x'$, равен (рис. 21 а)

$$O'A = AB - BO' = \frac{mh}{\operatorname{tg} \varphi} - nh = h \left(\frac{m}{\alpha} - n \right).$$

Предположим, система координат $x''o''y''$ смещена относительно xoy на $(m-1)h$ по y и на nh по x . Предположим, сделан еще один шаг по y . Тогда величина отрезка $O'A$ по вышеприведенной формуле равна $h \left(\frac{m}{\alpha} - n \right)$. Если $h \left(\frac{m}{\alpha} - n \right) \leq \frac{1}{2} h$ или, сокращая на h , $\frac{m}{\alpha} - n \leq \frac{1}{2}$, будем делать перемещение на h по оси y , в против -

ном случае, т.е. если $\left(\frac{m}{\alpha} - n\right) > \frac{1}{2}$ (как показано на рис. 21δ), сделаем комбинированное перемещение. С каждым новым шагом m по y будем сравнивать его величину с $\frac{1}{2}h$. В случае, когда $m = \frac{1}{2}h$, аппроксимация отрезка $i, i+1$ закончена.

Пусть теперь $\alpha < 1$. Допустим, система $x'o'y'$ смещена относительно xoy на mh по y и на nh по x . Тогда (рис. 22а)

$$O'A' = AB - O'B = nh \operatorname{tg} \varphi - mh = h(\alpha n - m).$$

Предположим, система координат $x''o''y''$ смещена относительно xoy на $(n-1)h$ по x и на mh по y .

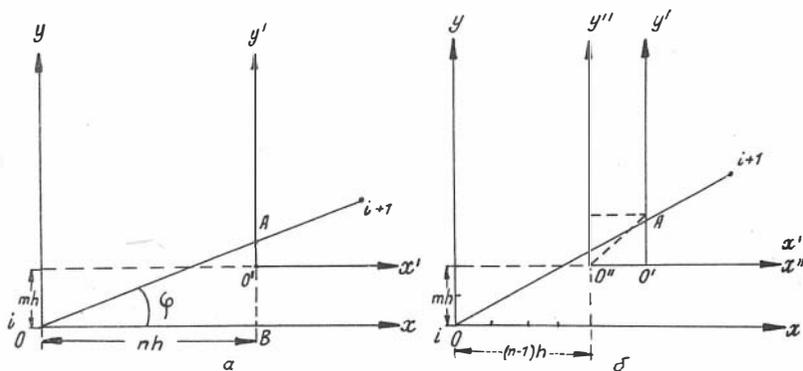


Рис.22

Рассмотрим теперь отрезок $O'A$ в системе $x'o'y'$, смещенной на nh по x и на mh по y относительно xoy . Если $O'A \leq \frac{1}{2}h$, т.е. $h(\alpha n - m) \leq \frac{1}{2}h$, или $(\alpha n - m) \leq \frac{1}{2}$, сделаем перемещение по оси x на h . В противном случае делаем комбинированное перемещение (рис. 22б) и с каждым новым шагом h по оси x сравниваем его с $\frac{1}{2}h$. В случае совпадения аппроксимация отрезка $i, i+1$ закончена.

Все вышеизложенное применимо в случаях, когда приращения отрезка по обеим осям имеют знак "+". Но, вообще говоря, приращения по любой оси могут иметь и знак "-". Поэтому прежде, чем давать сигнал на какое-либо перемещение, нужно установить знак, который имеют приращения Δx и Δy . В зависимости от знака перемещения могут быть $h^x h^y$, $h^x h^{-y}$, $h^{-x} h^y$, $h^{-x} h^{-y}$, $h^x h^{-x}$, $h^y h^{-y}$.

1. Описание

Функциональной блок-схем

В блоке 1 происходит вычисление полных перемещений по x и по y (Δx , Δy), полного количества шагов по x и по y (γ^x и γ^y) и тангенса угла наклона заданного отрезка прямой α . Затем проверяется условие $\alpha \geq 1$ (блок 2). Если $\alpha \geq 1$, то переходим к проверке $m = \gamma^y$ (блок 3). Пусть $m \neq \gamma^y$. Тогда, считая $m = m + 1$, проверяем условие $\frac{m}{\alpha} - n \leq \frac{1}{2}$ (блок 4). При выполнении

этого условия перемещение будет производиться по оси y , а направление (знак) его определяется в блоке 5. Если условие $\frac{m}{\alpha} - n \leq \frac{1}{2}$ не выполняется, перемещение будет

комбинированным, а знак его определяется в блоке 6, в котором также подсчитывается n - число шагов по x . Так как в результате комбинированного перемещения произойдет перемещение и по оси x , то после него $n = n + 1$. После любого перемещения возвращаемся к проверке $m = \gamma^y$, т.е. проверяем, не исчерпано ли полное число шагов по y . Если нет, то опять идем на блок 4. Если $m = \gamma^y$, это означает конец работы.

В случае, когда условие $\alpha \geq 1$ не выполняется, переходим к проверке $n = \gamma^x$ (блок 7). Пусть $n \neq \gamma^x$. Тогда, считая $n = n + 1$, проверяем условие $\alpha \cdot n - m \leq \frac{1}{2}$ (блок 8). При выполнении этого условия перемещение будет производиться по оси x , а направление перемещения определяется в блоке 9. Если условие $\alpha \cdot n - m \leq \frac{1}{2}$ не вы-

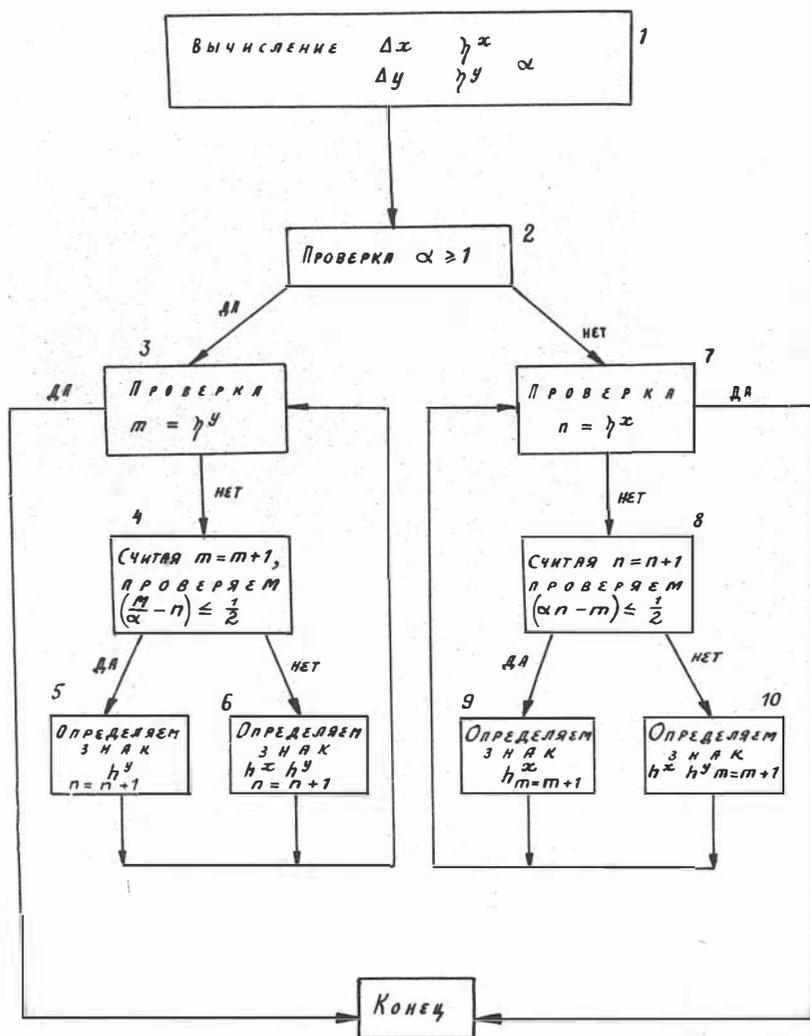


Рис. 23 Функциональная блок-схема

полняется, то перемещение будет комбинированным, а знак его определяется в блоке 10, в котором также подсчитывается число шагов по y , которое после комбинированного перемещения становится $m = m + 1$. После перемещения возвращаемся к проверке $n = \gamma^x$, т.е. проверяем, не исчерпано ли полное число шагов по x . Когда $n = \gamma^x$ - конец работы.

2. О п и с а н и е р а б о т н о п е р а т о р н о й с х е м ы а л г о р и т м а

Оператор A_1 подсчитывает Δx , Δy . Оператор A_2 вычисляет $\gamma^x \gamma^y$. Оператор A_3 подсчитывает α . Затем оператор P_4 проверяет условие $\alpha \geq 1$. При выполнении этого условия работает оператор P_5 , проверяющий условие $m = \gamma^y$. Пока это условие не выполняется, после P_5 работает A_6 , который подсчитывает $m = m + 1$. Затем оператор P_7 проверяет условие $\left(\frac{m}{\alpha} - n\right) \leq \frac{1}{2}$. При выполнении этого условия оператор P_8 проверяет условие $\Delta y > 0$, т.е. устанавливает знак Δy . При выполнении условия $\Delta y > 0$ оператором Φ_9 дается сигнал на перемещение h^y . При невыполнении, т.е. если $\Delta y < 0$, оператором Φ_{10} дается сигнал на перемещение h^{-y} . Операторы Φ_9 и Φ_{10} передают управление на P_5 . В том случае, когда условие $\left(\frac{m}{\alpha} - n\right) \leq \frac{1}{2}$ не выполняется (проверка в P_7), после оператора P_7 работает оператор P_{11} , проверяющий условие $\Delta y > 0$. Если $\Delta y > 0$, то оператор P_{12} проверяет знак Δx . Когда $\Delta x > 0$, то оператор Φ_{13} дает сигнал на перемещение $h^x h^y$. Если $\Delta x < 0$, то оператор Φ_{14} дает сигнал на перемещение $h^{-x} h^y$. В том случае, когда $\Delta y < 0$ (проверка в P_{11}), то знак Δx проверяет оператор P_{15} . Если $\Delta x > 0$, то оператор Φ_{16} дает сигнал на перемещение $h^x h^{-y}$, если же $\Delta x < 0$, то Φ_{17} дает сигнал на перемещение $h^{-x} h^{-y}$. От всех операторов, дающих сигнал на комбинированное перемещение, т.е. от Φ_{13} , Φ_{14} , Φ_{16} , Φ_{17} управление передается на оператор A_{18} , который подсчитывает n . От него - выход на оператор P_5 . Когда $m = \gamma^y$ (проверка P_5), работа

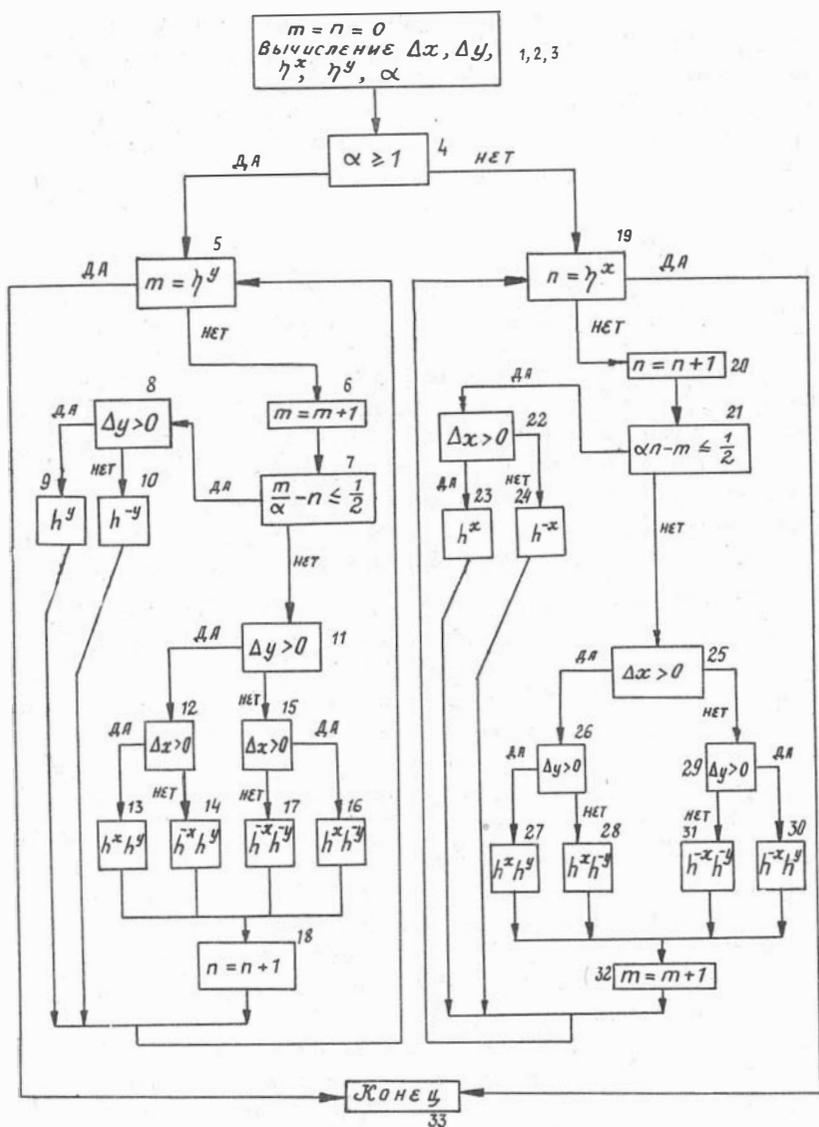


Рис. 24. Блок-схема аппроксимации отрезка прямой

заканчивается (оператор Я₃₃).

Другая часть операторной схемы работает, если условие $\alpha \geq 1$ не выполняется (проверка Р₄). В этом случае после оператора Р₄ работает оператор Р₁₉, который проверяет условие $n = \gamma^x$. Пока это условие не выполнено, работает А₂₀, который подсчитывает n . Затем оператор Р₂₁ проверяет условие $(\alpha \cdot n - m) \leq \frac{1}{2}$. При выполнении этого условия оператор Р₂₂ проверяет условие $\Delta x > 0$. В случае, когда $\Delta x > 0$, оператор Ф₂₃ дает сигнал на перемещение h^x , когда $\Delta x < 0$, оператор Ф₂₄ дает сигнал на перемещение h^{-x} . И от Ф₂₃ и от Ф₂₄ управление передается на Р₁₉. В том случае, когда условие $(\alpha \cdot n - m) \leq \frac{1}{2}$ не выполняется, после оператора Р₂₁ работает оператор Р₂₅, который проверяет знак Δx . Если $\Delta x > 0$, то оператор Р₂₆ проверяет знак Δy . Когда $\Delta y > 0$, то оператор Ф₂₇ дает сигнал на перемещение $h^x h^y$, когда $\Delta y < 0$, то Ф₂₈ дает сигнал на перемещение $h^x h^{-y}$. Когда $\Delta x < 0$ (проверка Р₂₆), то знак Δy проверяет оператор Р₂₉. Если $\Delta y > 0$, то оператор Ф₃₀ дает сигнал на перемещение $h^{-x} h^y$, в противном случае оператор Ф₃₁ дает сигнал на перемещение $h^{-x} h^{-y}$. От всех операторов, дающих сигнал на комбинированное перемещение, т.е. от Ф₂₇, Ф₂₈, Ф₃₀, Ф₃₁ управление передается на оператор А₃₂, который подсчитывает m . От А₃₂ выход на Р₁₉. Когда $n = \gamma^x$, (проверка Р₁₉) работа заканчивается (оператор Я₃₃).

3. Структура операторной схемы алгоритма

- А₁ - вычисление $\Delta x, \Delta y$.
- А₂ - вычисление γ^x, γ^y .
- А₃ - вычисление α .
- Р₄ - проверка $\alpha \geq 1$.
- Р₅ - проверка $m = \gamma^y$.
- А₆ - подсчет $m = m + 1$.
- Р₇ - проверка: $(m/\alpha - n) \leq \frac{1}{2}$.
- Р₈ - проверка $\Delta y > 0$.

- Φ_9 - дается сигнал на перемещение h^y .
 Φ_{10} - дается сигнал на перемещение h^{-y} .
 P_{11} - проверка: $\Delta y > 0$.
 P_{12} - проверка: $\Delta x > 0$.
 Φ_{13} - сигнал на перемещение $h^x h^y$.
 Φ_{14} - сигнал на перемещение $h^{-x} h^y$.
 P_{15} - проверка: $\Delta x > 0$.
 Φ_{16} - сигнал на перемещение $h^x h^{-y}$.
 Φ_{17} - сигнал на перемещение $h^{-x} h^{-y}$.
 A_{18} - подсчет $n = n + 1$.
 P_{19} - проверка: $n = \eta^x$.
 A_{20} - подсчет $n = n + 1$.
 P_{21} - проверка: $(\alpha \cdot n - m) \leq \frac{1}{2}$.
 P_{22} - проверка: $\Delta x > 0$.
 Φ_{23} - сигнал на перемещение h^x .
 Φ_{24} - сигнал на перемещение h^{-x} .
 P_{25} - проверка: $\Delta x > 0$.
 P_{26} - проверка: $\Delta y > 0$.
 Φ_{27} - сигнал на перемещение $h^x h^y$.
 Φ_{28} - сигнал на перемещение $h^x h^{-y}$.
 P_{29} - проверка: $\Delta y > 0$.
 Φ_{30} - сигнал на перемещение $h^{-x} h^y$.
 Φ_{31} - сигнал на перемещение $h^{-x} h^{-y}$.
 A_{32} - подсчет: $m = m + 1$.
 Λ_{33} - конец.

$A_1 A_2 A_3 P_{4+18} \quad 10, 9, 18 P_5^{+33} \quad A_6 P_{7+11} P_{8+10} \quad \Phi_9^5 \quad \Phi_{10}^5$
 $P_{11+15} P_{12+14} \Phi_{13}^{18} \quad 12 \Phi_{14}^{18} \quad 11 P_{15+17} \quad \Phi_{16}^{18} \quad 15 \Phi_{17}^{18} \quad 13, 14, 16 \quad A_{18}^5$
 $32, 24, 23, 4 P_{19}^{+33} \quad A_{20} \quad P_{21+25} \quad P_{22+24} \quad \Phi_{23}^{19} \quad 22 \Phi_{24}^{19} \quad 21 P_{25+29} \quad P_{26+28}$
 $\Phi_{27}^{32} \quad 26 \Phi_{28}^{32} \quad 25 P_{29+31} \quad \Phi_{30}^{32} \quad 29 \Phi_{31}^{32} \quad 27, 28, \quad 30 A_{32}^5 \quad 58 \Phi_{33}^5$

4. Описание работ операторной схем программы УГ (управление графопостроителем)

Программа УГ выполняет 8 различных операций. В зависимости от той операции, которую должна выполнить программа УГ, ее работа начинается с одного из операторов с H_1 по H_8 . Оператор H_1 начинает работу программы УГ, когда нужно провести отрезок из точки стояния карандаша по заданным Δx , Δy . Если нужно провести отрезок до заданной точки x_k , y_k , то программа начнет работать с оператора H_2 . Эти два режима проведения отрезков прямых линий являются основными для программы УГ. Другие ее функции следующие. Она может дать сигнал на подъем и спуск карандаша – в этих случаях работа начинается соответственно с операторов H_3 и H_4 . Оператор H_5 начинает режим "уход в 0,0", т.е. подъем и перемещение карандаша в точку 0,0. Оператор H_6 является началом в случае, когда требуется изменить масштаб. Изменение масштаба производится через изменение K_x , K_y , которые соответственно означают число шагов графопостроителя в принятой единице измерения по осям x и y (при единице измерения по осям x и y , равной 1 мм, $K_x = K_y = 10$ для графопостроителя с шагом 0,1 мм).

Если требуется по ходу работ изменить угол поворота осей θ , то программа начинается с оператора H_7 . Задаются Δx , Δy , по которым определяются $\sin \theta$ и $\cos \theta$, используемые в дальнейших расчетах. Значение угла не подсчитывается, а поворот считается от оси x против часовой стрелки. Оператор H_8 является началом работы, когда программа выполняет операцию обнуления, т.е. восстановления исходных K_x , K_y , $\sin \theta$, $\cos \theta$ и, кроме того, присвоение начала координат точке стояния карандаша в данный момент.

Пусть требуется провести отрезок из точки стояния карандаша по заданным Δx , Δy . Обращение к программе происходит через оператор H_1 . За ним следует оператор A_9 , который засылает в оператор K_{12} ключ K_1 . Оператор A_{10} , рабо -

тающий после A_9 , выбирает строку информации из обращения, а оператор A_{11} выбирает из этой строки информацию, по которой в памяти находит Δx , Δy и угол θ . Затем стоит оператор K_{12} , в который записывается нужный ключ. В зависимости от ключа далее путь разветвляется. В рассматриваемом случае в него записан ключ K_1 . Этот ключ ведет программу к оператору P_{13} , который проверяет: $\cos \theta = 1$? Если поворота осей не было, то $\cos \theta = 1$, и после оператора P_{13} работает оператор A_{15} , подсчитывающий x_k, y_k . К координатам точки, в которой находится карандаш, x_{k-1}, y_{k-1} добавляются $\Delta x, \Delta y$, т.е. вычисляются координаты точки, в которую карандаш должен прийти. Если же был поворот осей, т.е. $\cos \theta \neq 1$, то после оператора P_{13} работает оператор A_{14} , подсчитывающий $\Delta x, \Delta y$ с учетом θ , а после A_{14} — оператор A_{15} . Оператор A_{16} переводит полученные в A_{15} координаты новой точки стояния карандаша в целое число шагов графопостроителя \tilde{x}_k, \tilde{y}_k .

Оператор A_{17} вычисляет Δx по $\tilde{x}_k, \tilde{x}_{k-1}$ и записывает его в ячейку, соответствующую Δ ведущему. Оператор P_{18} анализирует знак Δx . В зависимости от знака заносится сигнал для графопостроителя „+ x” ведущий — оператором A_{19} или „- x” ведущий — оператором A_{20} . После любого из этих операторов работает оператор A_{21} , который по $\tilde{y}_k, \tilde{y}_{k-1}$ вычисляет Δy и записывает его в ячейку, соответствующую Δ ведомому. Оператор P_{22} анализирует знак Δy . В зависимости от знака заносится сигнал для графопостроителя „+ y” ведомый — оператором A_{23} или „- y” ведомый — оператором A_{24} .

Оператор A_{25} заносит координаты новой точки стояния карандаша $x_k, y_k, \tilde{x}_k, \tilde{y}_k$ (вместо прежних $x_{k-1}, y_{k-1}, \tilde{x}_{k-1}, \tilde{y}_{k-1}$). Далее оператор P_{26} выясняет, действительно ли Δx является ведущим, т.е. $\Delta x > \Delta y$? Если окажется, что $\Delta x < \Delta y$, значит, ведущим будет перемещение по y и оператор A_{27} производит перестановку Δ ведущего и Δ ведомого и соответственно сигналов S ведущего и S' ведомого. После оператора A_{27} , а также после оператора P_{26} , когда $\Delta x > \Delta y$, работает оператор A_{28} , который вычисляет отношение Δ ведомого к Δ ведущему.

Оператор A_{29} записывает 0 в ячейку R . В дальнейшем

в ячейку R записывается ошибка, которая появляется в результате выражения координат нужной точки в целых числах шагов графопостроителя. Ошибка эта постепенно растет, но данная программа не дает ей превзойти 0,5 шага. Оператор K_{30} — это установка начального значения счетчика шагов графопостроителя. Число шагов определяется по Δ ведущему. Оператор K_{31} вычитает из счетчика шагов 1. Оператор P_{32} проверяет, не появилось ли в счетчике отрицательное число, т.е. есть ли еще шаги. Если шагов нет, то требуемое по программе перемещение произведено, работа закончена. Оператор Y_{33} — конец работ.

Но пока шаги есть, после оператора P_{32} работает оператор A_{34} , который вычисляет накапливающуюся ошибку и записывает ее в ячейку R . Ошибка подсчитывается из условия, что будет сделан шаг только по ведущей. Оператор A_{35} записывает ведущий сигнал на место входного сигнала, т.е. на управление графопостроителем. Оператор A_{36} проверяет, не превзошла ли ошибка 0,5 шага. Если ошибка меньше 0,5 шага, то после оператора P_{36} работает оператор A_{39} , который осуществляет исполнение шага. Если же ошибка превзошла 0,5 шага или равна 0,5 шага, то после оператора P_{36} работает оператор A_{37} , который вычитает из ошибки единицу, т.е. предполагает, что будет сделан шаг и по ведущей и по ведомой. Новое значение ошибки записывается в ячейку R . Оператор A_{38} совмещает ведущий и ведомый сигналы и этот совмещенный сигнал также идет на оператор A_{39} , который дает команду непосредственно графопостроителю на исполнение шага, т.е. на перемещение карандаша соответственно сформированному сигналу. Оператор A_{40} пропускает несколько холостных циклов с целью задержки времени, необходимого для срабатывания графопостроителя. После A_{40} происходит переход на оператор K_{31} для осуществления очередного шага. Когда отрезок кончается, т.е. шагов больше нет, после P_{32} происходит переход на оператор Y_{33} . Этот оператор состоит из 2-х операторов: оператора A_{41} , который прибавляет единицу ко 2-ой команде адреса и тем самым получает команду возврата (по условиям программы), и оператора A_{42} , который производит уход в программу.

Пусть теперь требуется провести отрезок из точки стояния карандаша до точки с заданными координатами x_k , y_k . Обращение к УГ в этом случае начнется с оператора H_2 , а после него оператор A_{43} засылает в оператор K_{12} ключ K_2 . Работают операторы A_{10} и A_{11} , выбирающие строку информации и координаты заданной точки x_k , y_k , а затем оператор K_{12} направляет программу по ключу K_2 . По этому ключу программа переходит на оператор A_{16} , переводящий координаты x_k , y_k в целое число шагов графопостроителя. Дальше путь программы ничем не отличается от описанного выше, когда отрезок проводится по заданным Δx , Δy .

Подъем и спуск пера графопостроителя осуществляется непосредственно командой, подаваемой операторами A_{44} (подъем) и A_{45} (спуск), которые работают соответственно за операторами H_3 и H_4 . После операторов A_{44} и A_{45} оператор A_{46} осуществляет задержку времени для срабатывания графопостроителя. На этом работа заканчивается, происходит переход к Я33.

Операция "уход в 0,0" осуществляется следующим образом. После оператора H_5 работает оператор A_{48} , который заносит 0 в x_k , y_k , а затем оператор A_{49} производит подъем пера. Дальше работает оператор A_{16} и по схеме осуществляется перемещение карандаша в точку 0,0.

Для изменения K_x , K_y после оператора H_6 оператор A_{50} засылает в оператор K_{12} ключ K_3 . После A_{50} работают операторы A_{10} , A_{11} , дающие информацию о требуемом изменении K_x , K_y и после K_{12} по ключу K_3 происходит переход на оператор A_{51} , который меняет K_x , K_y , после чего - переход к Я33, конец работы.

Изменение угла поворота осей θ начинается с обращения к оператору H_7 . После него оператор A_{52} засылает в K_{12} ключ K_4 . Затем операторы A_{10} , A_{11} дают информацию о требуемом изменении и после K_{12} по ключу K_4 происходит переход на оператор A_{53} , вычисляющий по заданным в информации Δx и Δy значения $\sin \theta$ и $\cos \theta$, а дальше - уход на Я33.

И, наконец, операция обнуления. Начало - оператор H_8 . Дальше идет оператор A_{54} , который выполняет требуемое за -

дание: засылка 0 в x_{k-1} , y_{k-1} , \tilde{x}_{k-1} , \tilde{y}_{k-1} , восстановление исходных K_x , K_y и угла θ . На этом работа заканчивается оператором Я33.

Ст р у к т у р а о п е р а т о р н о й с х е м ы п р о г р а м м ы У Г.

- Н₁ - перемещение по заданным Δx , Δy (начальный оператор).
- Н₂ - перемещение до точки x_k , y_k (начальный оператор).
- Н₃ - подъем } начальные операторы.
- Н₄ - спуск }
- Н₅ - уход в 0,0 (начальный оператор).
- Н₆ - изменение K_x , K_y (начальный оператор).
- Н₇ - изменение угла θ (начальный оператор).
- Н₈ - обнуление (начальный оператор).
- А₉ - засылка ключа К₁.
- А₁₀ - выборка строки информации.
- А₁₁ - выборка адресных чисел из строки информации.
- К₁₂ - оператор-ключ.
- Р₁₃ - проверка: $\cos \theta = 1$?
- А₁₄ - вычисление Δx , Δy с учетом θ .
- А₁₅ - вычисление x_k , y_k .
- А₁₆ - вычисление \tilde{x}_k , \tilde{y}_k .
- А₁₇ - вычисление Δ ведущего.
- Р₁₈ - проверка: $\Delta x > 0$?
- А₁₉ - запоминание сигнала „+x” ведущий.
- А₂₀ - запоминание сигнала „-x” ведущий.
- А₂₁ - вычисление Δ ведомого.
- Р₂₂ - проверка: $\Delta y > 0$?
- А₂₃ - запоминание сигнала „+y” ведомый.
- А₂₄ - запоминание сигнала „-y” ведомый.
- А₂₅ - замена x_{k-1} , y_{k-1} , \tilde{x}_{k-1} , \tilde{y}_{k-1} на x_k , y_k , \tilde{x}_k , \tilde{y}_k .
- Р₂₆ - проверка: $\Delta x > \Delta y$?
- А₂₇ - перестановка Δ ведущего и Δ ведомого, сигнал - лов S ведущего, S ведомого.

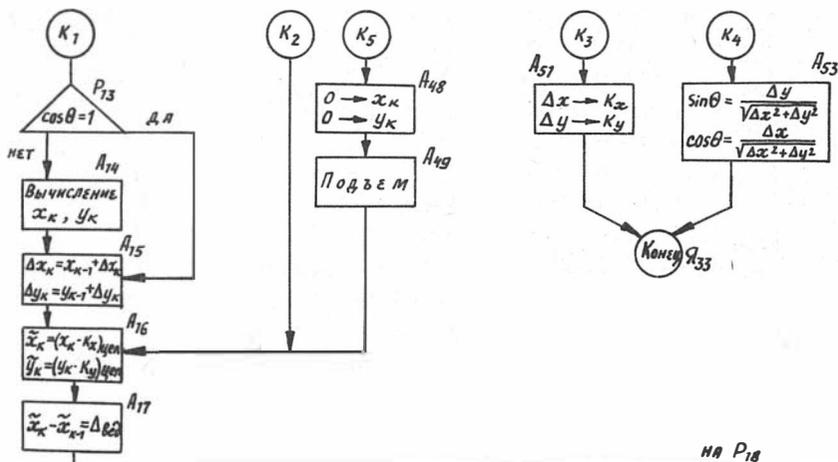
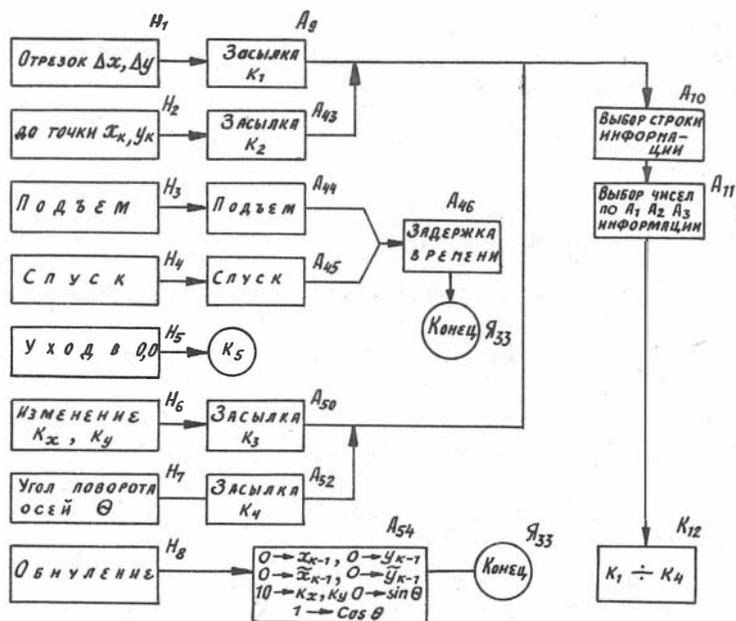


Рис. 25 Блок-схема программы УР (управление графовоэдром)

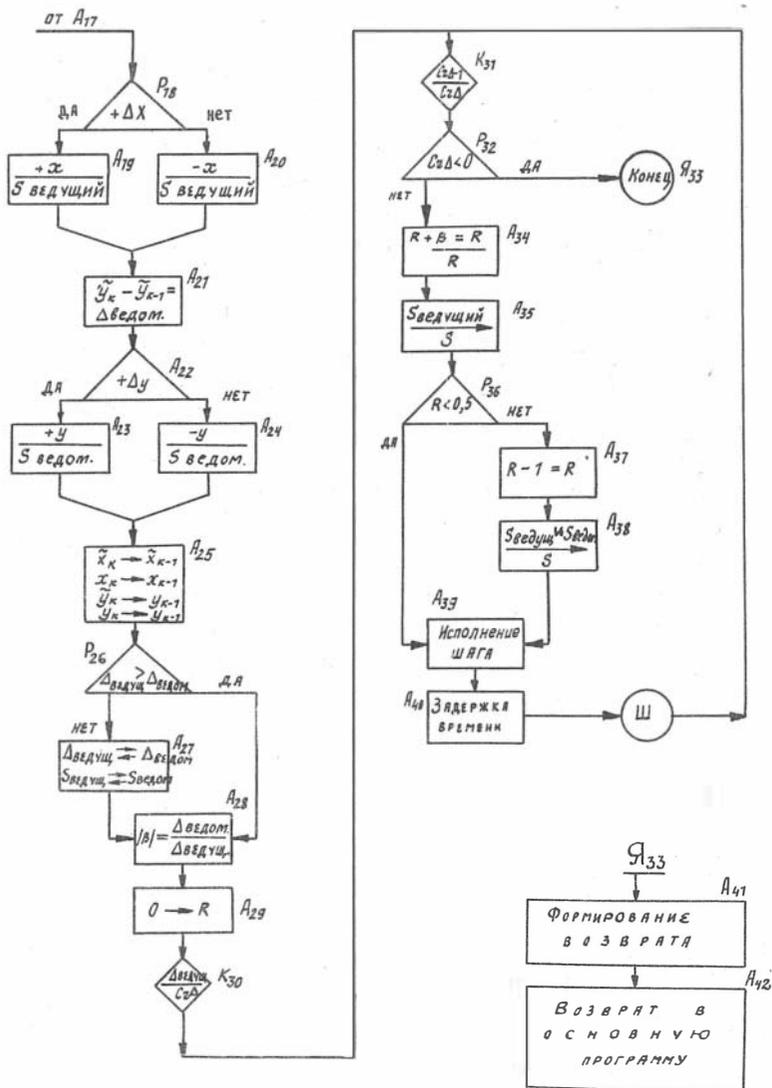


Рис. 26 Блок-схема программы УР (продолжение)

- A_{28} - вычисление $|\beta| = \frac{\Delta \text{ ведомое}}{\Delta \text{ ведущее}}$.
 A_{29} - засылка 0 в ячейку R .
 K_{30} - установка начального значения счетчика шагов графопостроителя.
 K_{31} - вычитание 1 из счетчика K_{30} .
 P_{32} - проверка: счетчик шагов меньше 0?
 A_{33} - конец работ.
 A_{34} - подсчет накапливающейся ошибки.
 A_{35} - засылка сигнала S ведущ. на вход графопостроителя.
 P_{36} - проверка: ошибка меньше 0,5 шага?
 A_{37} - вычитание 1 из ошибки R .
 A_{38} - засылка на вход графопостроителя совмещенного сигнала S' ведомый и S' ведущий.
 A_{39} - команда на исполнение шага.
 A_{40} - задержка времени на исполнение шага.
 A_{41} - вычисление команды возврата.
 A_{42} - уход в программу. (УП)
 A_{43} - засылка ключа K_2 .
 A_{44} - команда на подъем пера графопостроителя.
 A_{45} - команда на спуск пера графопостроителя.
 A_{46} - задержка времени на исполнение команды.
 A_{48} - занесение 0 в x_k, y_k .
 A_{49} - команда на подъем и перемещение пера в 0,0.
 A_{50} - засылка ключа K_3 .
 A_{51} - засылка $\Delta x, \Delta y$ в K_x, K_y .
 A_{52} - засылка ключа K_4 .
 A_{53} - вычисление $\sin \theta, \cos \theta$.
 A_{54} - обнуление.

$H_1^9, H_2^{43}, H_3^{44}, H_4^{45}, H_5^{48}, H_6^{50}, H_7^{52}, H_8^{54}, A_9^{43,50,52}, A_{10}^{13,16,51,53}, A_{11}, A_{12}^{13,16,51,53}, P_{13} \neq 15$
 $A_{14}^{13}, A_{15}^{12,15,49}, A_{16}^{17}, A_{17}, P_{18}^{42}, A_{19}^{21}, A_{20}^{18}, A_{21}^{19,20}, P_{22}^{24}, A_{23}^{25}, A_{24}^{22}, A_{25}^{22}$
 $A_{26}^{23,29}, P_{26}^{26}, A_{27}^{26,27}, A_{28}, A_{29}, K_{30}^{30,40}, K_{37}, P_{32}^{34}, A_{33}^{46,51,53,54}, A_{35}^{32}, A_{34}^{34}$
 $A_{35}, P_{36}, A_{39}, A_{37}, A_{38}, A_{39}^{36}, A_{40}^{31}, A_{41}^{33}, A_{42}^{40}, A_{43}^{2,10}, A_{44}^{3,46}, A_{45}^{4,45}$
 $A_{46}^{44,45,33}, A_{48}^5, A_{49}^{16}, A_{50}^{6,10}, A_{51}^{12,33}, A_{52}^{7,10}, A_{53}^{12,33}, A_{54}^{8,33}$

5. И Н С Т Р У К Ц И Я

Так как в комплекте машин до сего времени графопостроитель не предусматривался, на рисунке приведена логическая схема подключения графопостроителя к ЭВМ типа М-220.

Графопостроитель управляется машиной в реальном масштабе времени. Для передачи на графопостроитель сигналов обработки шага или подъема или спуска пера использована команда "67" (сдвиг циклический) с небольшой модификацией. Модификация заключается в том, что в А2 команды 67 должна быть "1" П-го разряда, т.е. команда имеет вид:

67 С 2000 0000,

где С - адрес управляющего слова.

Для управления перемещениями пера а также для управления подъемом или спуском в управляющем слове должны быть единички в соответствующих разрядах (задействованы только шесть разрядов с 25 + 30, назначение каждого видно из рисунка).

Модифицированная команда 67 используется в данном случае только для передачи управляющего слова через Р1 на регистр ГП, а с него на ГП. Обычный сдвиг кода, который производится этой командой, происходит как обычно, но никакого отношения к графопостроителю не имеет.

Формированием разрядов в управляющем слове и передачей их на ГП занимается блок УГ - управление графопостроителем.

Программа "Управление графопостроителем (УГ)" выполнена в виде самостоятельной СП. Обращение к подпрограмме состоит из 2-х строк

	КОП	А ₁	А ₂	А ₃
x	016	$x + 1$	6100+6107	6110
$x+1$	—	x	—	y_k

В первой строке в КОПе стоит "016",

А₁ - адрес следующей строки,

- A_2 - адрес одного из восьми входов в программу
- 6100 - обнуление координат (листа)
 - 6101 - вччерчивание отрезка $\Delta x, \Delta y$
 - 6102 - вччерчивание луча (ход до x_k, y_k)
 - 6103 - подъем пера
 - 6104 - спуск пера
 - 6105 - уход в начало координат (0,0)
 - 6106 - изменение K_x, K_y
 - 6107 - задание угла поворота осей θ_x, θ_y

A_3 - адрес команды возврата (6110).

Во второй строке обращения указываются адреса параметров (x, y)

В случае обнуления координат, подъема, спуска, ухода в (0,0) x, y - произвольны.

Вччерчивание отрезков: $x = \Delta x, y = \Delta y$ (в мм)

Вччерчивание луча: $x = x_k, y = y_k$ (в мм)

Изменение K_x, K_y : $x = K_x, y = K_y$ (число шагов)

Поворот осей: $x = \theta_x, y = \theta_y$ (числа) $\Gamma\Pi$ в 1 мм)

После выполнения заданного действия УГ передает управление в ячейку $x+2$

Программа располагается в МОЗУ на фиксированном месте (6100 ÷ 6357).

Рабочие ячейки 0001 ÷ 0003.

Время действия зависит от характеристик графопостроителя.

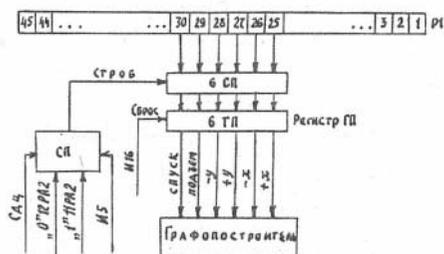


рис.26^а Схема подключения графопостроителя к ЭВМ

6. ПРОГРАММА "УПРАВЛЕНИЕ ГРАФОПОСТРОИТЕЛЕМ"

6100	056	0000	6170	0000	на обнуление
1.	016	6250	6220	6232	на $\Delta x, \Delta y$
2.	016	6265	6220	6232	на x_k, y_k
3.	032	0000	6350	7000	на подъем
4.	032	0000	6352	7000	на спуск
5.	056	0000	6204	0000	уход в 0,0
6.	016	6211	6220	6232	изменение
7.	016	6234	6220	6232	изменять угол поворота осей θ
6110	056	0000	6354	0000	на выход из УГ
1.	000	0000	0000	0000	команда возврата
2.	000	0010	0000	0000	+x
3.	000	0004	0000	0000	-x
4.	000	0001	0000	0000	+y
5.	000	0002	0000	0000	-y
6.	000	0020	0000	0000	подъем
7.	000	0040	0000	0000	спуск
6120	000	0100	0000	0000	
1.	000	0200	0000	0000	
2.	001	0000	0000	0000	x_k } координаты очередной
3.	002	0000	0000	0000	y_k } точки
4.	000	0003	0000	0000	x_{k-1} } координаты точки
5.	000	0004	0000	0000	y_{k-1} } стояния карандаша
6.	000	0005	0000	0000	
7.	000	0006	0000	0000	
6130	000	0007	0000	0000	
1.	000	0010	0000	0000	\sim } координаты точки
2.	000	0011	0000	0000	x_{k-1} } стояния карандаша
3.	000	0012	0000	0000	y_{k-1} } в шагах ГП
4.	000	0013	0000	0000	ведущий сигнал
5.	000	0014	0000	0000	ведомый сигнал
6.	000	0015	0000	0000	выходной сигнал
7.	000	0016	0000	0000	R - остаток
6140	000	0001	0000	0000	
1.	000	0010	0000	0000	
2.	000	0010	0000	0000	
3.	000	0100	0000	0000	$\sin \theta$
4.	101	4000	0000	0000	$\cos \theta$
5.	104	5000	0000	0000	k_x } коэффициенты осей
6.	104	5000	0000	0000	k_y } (число шагов ГП в ед.
7.	000	0000	0000	0000	длинн)
6150	000	0000	0000	0000	
1.	101	4000	0000	0000	
2.	100	4000	0000	0000	
3.	104	5000	0000	0000	
4.	110	4000	0000	0000	
5.	000	0000	0000	0000	
6.	000	0000	0000	0000	
7.	000	0000	0000	0000	

6160	000	0000	0000	0000
1.	000	0000	0000	0000
2.	000	0000	0000	0000
3.	000	0000	0000	0000
4.	000	0000	0000	0000
5.	000	0000	0000	0000
6.	000	0000	0000	0000
7.	000	0000	0000	0000
6170	000	0000	0000	6124
1.	000	0000	0000	6125
2.	000	0000	0000	6132
3.	000	0000	0000	6133

Обнуление

$0 \rightarrow x_{k-1}$
 $0 \rightarrow y_{k-1}$
 $0 \rightarrow \tilde{x}_{k-1}$
 $0 \rightarrow \tilde{y}_{k-1}$

4.	000	6153	0000	6145
5.	000	6153	0000	6146
6.	000	0000	0000	6143
7.	000	6151	0000	6144
6200	000	0000	0000	0000
1.	000	0000	0000	0000
2.	000	0000	0000	0000
3.	056	0000	6354	0000
4.	000	0000	0000	6122
5.	000	0000	0000	6123
6.	016	6207	6103	6354
7.	056	0000	6265	6354

$10 \rightarrow K_x$
 $10 \rightarrow K_y$
 $0 \rightarrow \sin \theta$
 $1 \rightarrow \cos \theta$

на вход из программ
 $0 \rightarrow x_k$ Уход в 0,0
 $0 \rightarrow y_k$
 подъем
 исполнение ухода в 0,0

6210	000	0000	0000	0000
1.	000	6122	0000	6145
2.	056	6123	6354	6146
3.	000	0000	0000	0000
4.	000	0000	0000	0000
5.	000	0000	0000	0000
6.	056	0000	6354	0000
7.	000	0000	0000	0000
6220	072	0000	6111	0000
1.	400	0000	0000	6137
2.	072	0000	6137	0000
3.	400	0000	0000	6140

Изменение K_x, K_y

$x_k \rightarrow K_x$
 $y_k \rightarrow K_y$

на вход из программ
 блок заделки информации
 выборка строки информации
 $[A_2]$

4.	054	0064	6137	0001
5.	072	0000	0001	0000
6.	400	0000	0000	6122
7.	054	0114	6137	0001
6230	072	0000	0001	0000
1.	400	0000	0000	6123
2.	000	0000	0000	0000
3.	000	0000	0000	0000
4.	005	6122	6122	0002
5.	005	6123	6123	0003
6.	001	0002	0003	0003
7.	044	0003	0003	0003

$[A_1]$

$[A_3]$

возврат

Δx^2
 Δy^2
 $\Delta x^2 + \Delta y^2$
 $\sqrt{\Delta x^2 + \Delta y^2}$

Изменение угла

6240	002	0000	6123	0002
1.	003	0000	0003	6143
2.	076	6151	6245	6144
3.	004	0002	0003	6143
4.	004	6122	0003	6144
5.	056	0000	6354	0000
6.	000	0000	0000	0000
7.	000	0000	0000	0000
6250	000	0000	0000	0000
1.	015	6144	6151	0000
2.	036	0000	6261	0000
3.	005	6122	6144	0001

$\sin \theta$

$\cos \theta$

на выход из программы

Исполнение $\Delta x, \Delta y$

$\cos \theta = 1?$

4.	005	6123	6143	0002
5.	005	6122	6143	0003
6.	005	6123	6144	0004
7.	002	0001	0002	6122
6260	001	0003	0004	6123
1.	001	6122	6124	6122
2.	001	6123	6125	6123
3.	056	0000	6265	0000
4.	144	4000	0000	0000
5.	005	6122	6145	0002
6.	041	0002	6264	0002
7.	002	0002	6264	0002

$x_k = x \cdot \cos \theta - y \cdot \sin \theta$

$y_k = x \cdot \sin \theta + y \cdot \cos \theta$

Исполнение x_k, y_k

\tilde{x}_k - целое число шагов

6270	005	6123	6146	0003
1.	041	0003	6264	0003
2.	002	0003	6264	0003
3.	002	0002	6132	6126
4.	036	6113	6276	6134
5.	000	6112	0000	6134
6.	002	0003	6133	6127
7.	036	6115	6301	6135
6300	000	6114	0000	6135
1.	000	0002	0000	6132
2.	000	6122	0000	6124
3.	000	0003	0000	6133

\tilde{y}_k - целое число шагов

$\tilde{x}_k - \tilde{x}_{k-1} \rightarrow \Delta \text{ведущ.}$

сигнал „- X”, если $-\Delta x$

сигнал „+ X”, если $+\Delta x$

$\tilde{y}_k - \tilde{y}_{k-1} \rightarrow \Delta \text{ведом.}$

сигнал „- Y”

сигнал „+ Y”

$\tilde{x}_k \rightarrow \tilde{x}_{k-1}$

$x_k \rightarrow x_{k-1}$

$\tilde{y}_k \rightarrow \tilde{y}_{k-1}$

4.	000	6123	0000	6125
5.	003	6127	6126	0000
6.	036	6126	6314	0001
7.	000	6127	0000	6126
6310	000	0001	0000	6127
1.	000	6134	0000	0001
2.	000	6135	0000	6134
3.	000	0001	0000	6135
4.	003	0000	6126	0000
5.	076	0000	6354	0000
6.	004	6127	6126	6140
7.	003	6140	0000	6140

$y_k \rightarrow y_{k-1}$

$\Delta \text{ведущ.} > \Delta \text{ведом.} ?$

реверс Δ ведущего
и соответствующих сигналов

$\beta = \frac{\Delta \text{ведом.}}{|\beta| \Delta \text{ведущ.}}$

6320	000	0000	0000	6137	$0 \rightarrow R$ (погрешность)
1.	003	6126	0000	6141	$ \Delta \text{ведущ} \rightarrow$ счетчик Δ очередной шаг счетчик $\Delta-1$, ведущ. сигнал $R + \beta \rightarrow R$ $R < 0,5 ?$
2.	003	6141	6151	6141	
3.	036	6134	6354	6136	
4.	001	6137	6140	6137	
5.	002	6137	6152	0000	
6.	036	0000	6331	0000	
7.	002	6137	6151	6137	
6330	075	6134	6135	6136	$R-1$ положение ведущ. и ведом. сигн.
1.	000	0000	0000	6336	исполнение элементарн. шага ГП
2.	067	6136	2000	0000	
3.	000	0000	0000	0000	
4.	052	0000	7740	0000	временная задержка
5.	112	7770	6335	0001	
6.	000	0000	0000	0000	за следующим шагом
7.	056	0000	6322	0000	
6340	000	0000	0000	0000	
1.	000	0000	0000	0000	
2.	000	0000	0000	0000	
3.	000	0000	0000	0000	
4.	000	0000	0000	0000	
5.	000	0000	0000	0000	
6.	000	0000	0000	0000	
7.	000	0000	0000	0000	
6350	067	6116	2000	0000	исполнение подъема карандаша
1.	056	0000	6353	0000	исполнение спуска карандаша
2.	067	6117	2000	0000	
3.	112	7770	6353	0001	
4.	000	0000	0000	0000	
5.	013	6111	6356	6357	
6.	000	0000	0001	0000	
7.	070	0000	0000	0000	
6360	000	0000	0000	0000	конец обращения к УГ $\mathcal{E}+1$ (формиров. команды возвр. уход в основную программу)
1.	000	0000	0000	0000	
2.	000	0000	0000	0000	
3.	000	0000	0000	0000	

**ЭЛЕМЕНТЫ АВТОМАТИЧЕСКОГО
ОФОРМЛЕНИЯ ГРАФИКОВ
В ГЕОЛОГИИ И ГЕОФИЗИКЕ**

Часть II

§4. ОПИСАНИЕ БЛОК-СХЕМЫ ПРОГРАММЫ "ГРАФИК"

Вывод геолого-геофизической информации из ЦВМ на ГП имеет ряд особенностей, в частности, вывод данных в несоизмеримых единицах, неизвестный заранее диапазон выводимых чисел. Это вызывает трудности при размещении графика в пределах листа, в подборе масштаба. Еще большие трудности возникают при выводе серии кривых на один лист. Кроме того, возникает потребность в оформлении выводимых данных: нанесение осей или сетки, масштабирование, подпись осей, различные нестандартные подписи, нанесение символов и т.д.

Предлагаемый комплекс программ "График" разработан специально для оформления выводимых на ГП графиков и выполняет значительную часть названных выше операций.

Блок-схема программы "График" состоит из нескольких блоков.

Б л о к " К о н т р о л ь в ы ч е р ч и в а н и я к р и в ы х ". При выводе на график семейства кривых может оказаться, что формат, подобранный для одной кривой, не подходит для другой, т.к. кривые могут значительно отличаться друг от друга. Некоторые точки кривых могут выходить за пределы выбранного формата. Описываемый блок и выполняет контроль за такими точками. Если очередная точка вышла за пределы листа, то карандаш ГП остается на месте до тех пор, пока кривая возвратится в заданные пределы. Тогда производится подъем карандаша, переход его в новую точку и вычерчивание кривой продолжается. Этот блок также преобразует реальные за-

данные значения величин x и y в абсолютные перемещения карандаша ГП в пределах листа с учетом их расположения относительно координатных осей.

Кроме того, этот блок управляет и такой операцией процесса вычерчивания кривых как вывод точки на график в виде того или иного символа, соединение этих точек линией.

Б л о к "В ы б о р ф о р м а т а и с м е н а л и с т а". Служит для автоматического распределения места на носителе для разных выдач одной задачи и для разных задач. Предлагается несколько стандартных форматов листа, линейные размеры которого кратны базовому. За базовый формат принят максимальный.

В конкретной задаче формат задается оператором.

Б л о к "К а д р и р о в а н и е". Служит для выбора рациональных значений в начале отсчета координатных осей путем анализа выводимых данных с тем, чтобы на график попадали отрезки осей, на которых находятся выводимые функции. Таким образом, основная задача блока — рациональное использование листа при размещении графика.

Б л о к "М а с ш т а б и р о в а н и е". Предназначен для подбора подходящего масштаба для каждой оси с учетом конкретного расположения осей и найденного значения начала координат.

Б л о к и "К о о р д и н а т н а я о с ь" и "В ы б о р ш а г а о с е й". Служат для вычерчивания по отдельности каждого луча координатных осей, начиная от выбранного блока кадрирования начала координат. На осях наносятся метки в соответствии с шагом осей, который может быть либо выбран, либо задан по номеру из числа предусмотренных стандартных.

Б л о к "К о о р д и н а т н а я с е т к а" предусмотрен для вычерчивания прямоугольной равномерной сетки при заданных шагах по осям x и y .

Б л о к "Р а м к а" вычерчивает рамку принятого фор-

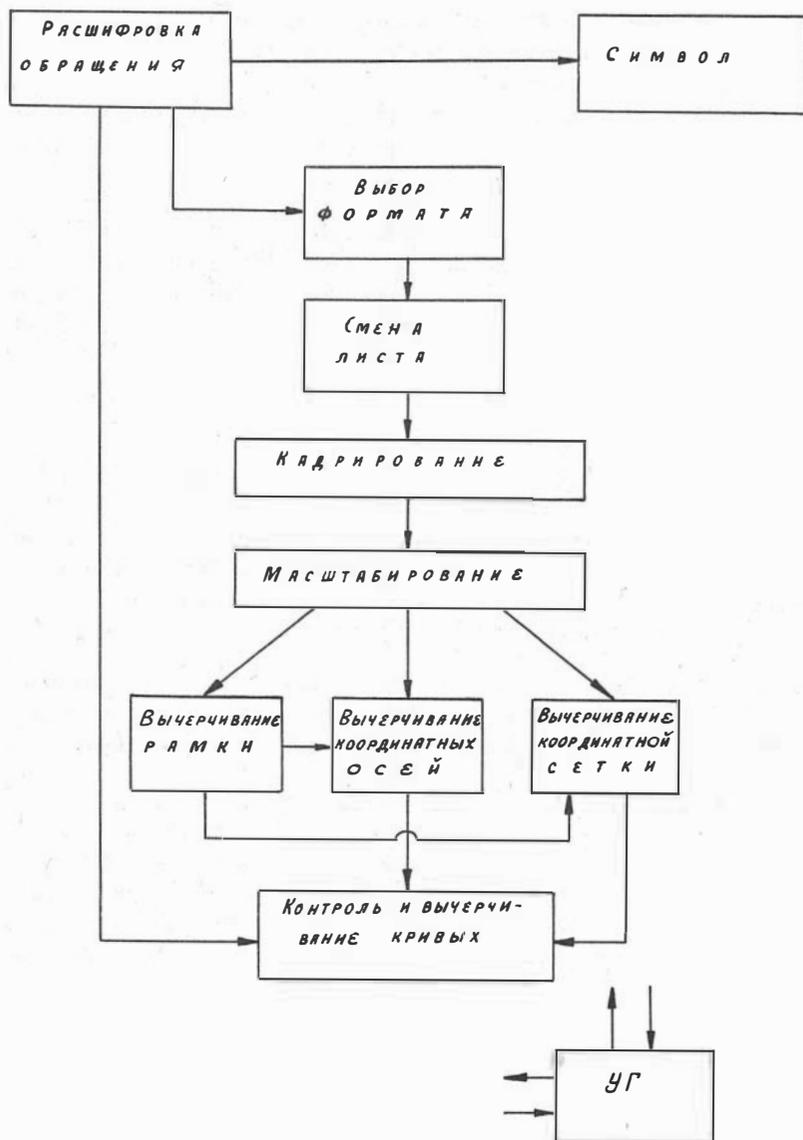


Рис. 27. Блок-схема программы «График»

мата.

Б л о к и "С и м в о л" и "Т е к с т". Служат для вычерчивания символов, исполнения надписей в требуемом месте с нужным углом наклона.

Б л о к "У п р а в л е н и е г р а ф о п о с т р о - и т е л е м" (УГ). Главная задача блока состоит в преобразовании в единично-импульсные коды числовых значений координат точек траекторий перемещения пера графопостроителя и в управлении последним.

ОПИСАНИЕ РАБОТЫ ОБЩЕЙ БЛОК-СХЕМЫ ПРОГРАММЫ "ГРАФИК"

Программа "График" после расшифровки команд обращения к ней может работать по одному из трех вариантов. В случае, когда требуется нанесение символов или текста, программа сразу приступает к этой работе (после расшифровки - блок "Символ"). Если задана смена листа, то после расшифровки работают последовательно блоки "Выбор формата", "Смена листа", "Кадрирование", "Масштабирование", "Вычерчивание рамки", осей или сетки, после чего - вычерчивание заданной кривой. Третий вариант предусмотрен для случаев, когда нет текста и нет смены листа. В этом случае после расшифровки команд обращения следует сразу блок "вычерчивание кривых" по заданному массиву числовых значений x и y .

Во всех вариантах работы программы используется программа УГ.

1. ОПИСАНИЕ БЛОКА "ВЫБОР ФОРМАТА И СМЕНА ЛИСТА" ПРОГРАММЫ "ГРАФИК"

О п и с а н и е р а б о т ы о п е р а т о р н о й с х е м ы

Подпрограмма предназначена для распределения места при построении графика по данным ЦВМ с помощью графопостроителя.

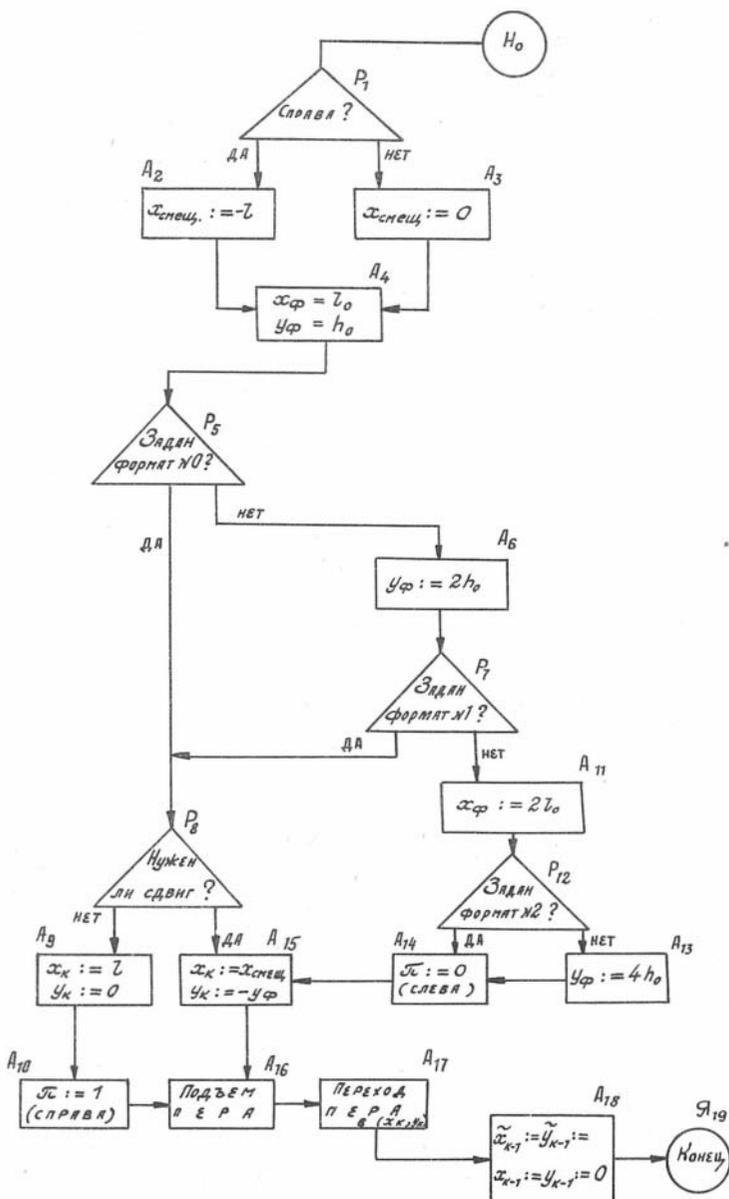


Рис. 28. Блок-схема выбора формата и смены листа

Начало программ - фиктивный оператор H_0 . Оператор P_1 проверяет, был ли предыдущий лист в правой стороне. Признаком, определяющим положение листа, является величина π . $\pi = 1$, когда лист находится справа и $\pi = 0$, когда лист находится слева. Оператор A_2 работает после P_1 , когда $\pi = 1$. Он запоминает, что должно быть смещение левого нижнего угла нового листа по оси x , равное l , где l - размер предыдущего листа по оси x . Если $\pi = 0$, то после P_1 работает оператор A_3 , запоминающий x смещения = 0. Оператор A_4 - выбор формата. Он подготавливает параметры формата 0, т.е. задает $x_\phi = l_0$, $y_\phi = h_0$. Оператор P_5 решает, действительно ли требуется нулевой формат.

Ноформата	Размер ($x_\phi \times y_\phi$)
0	200 × 150 мм
1	200 × 300 мм
2	400 × 300 мм
3	400 × 600 мм
—	—

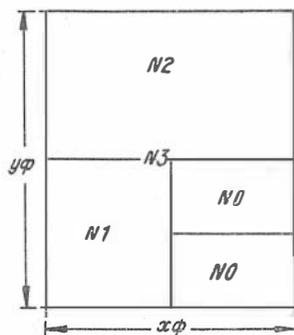
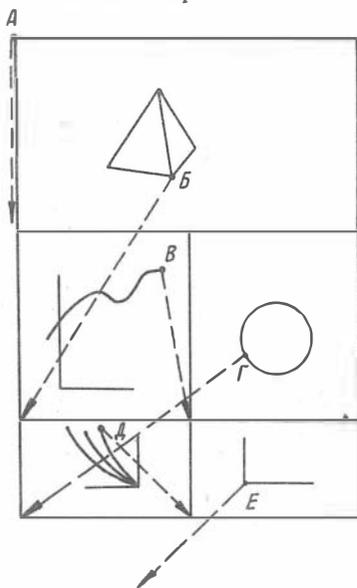


Рис. 29

Если да, то выбранные x_ϕ , y_ϕ окончательны, происходит переход на оператор P_8 . Если нет, то после оператора P_5 работает оператор A_6 , который подготавливает параметры формата № 1, для чего принимает $y_\phi = 2h_0$.

Оператор P_7 проверяет, является ли заданный формат первым. Если да, то выбранные $x_\phi = l_0$, $y_\phi = 2h_0$ окончательны, происходит переход на оператор P_8 , если нет, то оператор A_{11} подготавливает параметры формата № 2, которые равны $x_\phi = 2l_0$, $y_\phi = 2h_0$. Оператор P_{12} выясняет, действительно ли требуется формат № 2. Если да, то работает оператор A_{14} , если нет, то оператор A_{13} вычисляет $y_\phi = 4h_0$, что соответствует формату № 3. После оператора A_{13} , как и

после P_{12} , работает оператор A_{14} , который заносит 0 в ячейку π , так как форматы $\mathbb{P} 2$ и $\mathbb{P} 3$ могут располагаться только с левой стороны листа.



СХЕМЫ ПЕРЕХОДА К СЛЕДУЮЩЕМУ ЛИСТУ

А - НАЧАЛО ЗАДАЧИ

Б - ПЕРЕХОД К СЛЕДУЮЩЕМУ ЛИСТУ ВНИЗ

В - ПЕРЕХОД ВПРАВО

Г - ПЕРЕХОД ВЛЕВО ВНИЗ

Д - ПЕРЕХОД ВПРАВО

Е - ПЕРЕХОД ВЛЕВО ВНИЗ

Рис. 30

Оператор P_8 , который работает после операторов P_5 и P_7 , т.е. когда задан формат либо $\mathbb{P} 0$, либо $\mathbb{P} 1$, уточняет расположение предыдущего формата. Он проверяет: нужен ли сдвиг? Если да, то оператор A_{15} задает смещение карандаша $x_k = x_{\text{смещения}}$, $y_k = -y_f$. Если нет, то смещения задаются оператором A_9 и соответственно равны $x_k = l$, $y_k = 0$. После оператора A_9 работает оператор A_{10} , который заснляет в π единицу, т.к. теперь лист будет справа.

После оператора A_{10} , а также A_{15} работает оператор A_{16} , дающий команду на поднятие пера. Оператор A_{17} через программу УГ с заданными x_k, y_k осуществляет требуемое смещение. Оператор A_{18} производит засылку 0 в $\tilde{x}_{k-1}, \tilde{y}_{k-1}$,

x_{k-1} , y_{k-1} , т.е. принимает за начало координат листа точку, куда переместился карандаш. После этого работа заканчивается. Оператор Я₁₉ - конец работы.

Структура операторной схемы блока "В набор формата и смена листа"

$H_0 P_{13} A_2^{14} A_3^{15} A_4^{16} A_5^{17} A_6 P_{111}^{12} A_{115} A_9 A_{10}^{16} A_{11} P_{12}^{11} A_{13}^{12} A_{14}^8 A_{15}^8 A_{16}^{10} A_{17} A_{18} Я_{19}$.

- H₀ - начало (фиктивный блок).
- P₁ - проверка условия: был ли предыдущий лист в правой стороне?
- A₂ - занесение $X_{\text{смещ.}} = -l$ для смещения влево (если выполнено P₁).
- A₃ - занесение $X_{\text{смещ.}} = 0$ (если условие P₁ не выполнено)
- A₄ - занесение $x_{\varphi} = l_0$, $y_{\varphi} = h_0$ для формата № 0.
- P₅ - анализ номера формата $\mathbb{P}_{\Phi} = 0?$
- A₆ - удвоение высоты формата $y_{\varphi} = 2h_0$ для формата № 1.
- P₇ - анализ номера $\mathbb{P}_{\Phi} = 1?$
- P₈ - проверка возможности перехода вправо $\pi = 1?$
- A₉ - занесение $x_k = l_0$, $y_k = 0$ для перехода вправо.
- A₁₀ - занесение $1 \rightarrow \pi$ для запоминания, что лист справа.
- A₁₁ - удвоение длины $x_{\varphi} = 2l_0$ для формата № 2.
- P₁₂ - анализ $\mathbb{P}_{\Phi} = 2?$
- A₁₃ - удвоение высоты для формата № 3.
- A₁₄ - занесение $0 \rightarrow \pi$ (лист слева).
- A₁₅ - занесение $X_k = X_{\text{смещ.}}$, $Y_k = -Y_{\Phi}$ для перехода вниз или влево-вниз.
- A₁₆ - обращение к блоку подъема пера.
- A₁₇ - обращение к блоку обработки перемещения пера в точку $X_k Y_k$.
- A₁₈ - присвоение точке стояния пера абсолютной координаты $0, 0$ ($\tilde{x}_{k-1} = 0$, $x_{k-1} = 0$; $\tilde{y}_{k-1} = 0$, $y_{k-1} = 0$).
- Я₁₉ - конец данного блока, переход к следующему блоку.

2. ОПИСАНИЕ БЛОКА "КАДРИРОВАНИЕ" ПРОГРАММЫ "ГРАФИК"

О п и с а н и е р а б о т н
о п е р а т о р н о й с х е м ы

Основная задача подпрограммы – рациональный выбор начала системы координат при автоматическом вычерчивании графика.

Начало работы – оператор H_0 . Оператор A_1 отыскивает в данных задачи X_{max} , X_{min} , Y_{max} , Y_{min} . Оператор Φ_2 формирует настройку программы на ось x , т.е. сначала программа будет кадрировать ось x (находить рациональное расположение точки x_0). В общем случае начало координат располагается в середине листа. Но в зависимости от данных может оказаться более удобным расположить начало координат в другом месте. Программа "Кадрирование" обеспечивает возможность смещения x_0 , y_0 на $1/4$ формата листа вправо и влево от оси y , вверх и вниз от оси x и смещение одновременно от оси x и от оси y также в пределах $1/4$ формата листа, т.е. начало координат может быть в 9 различных точках (рис.31а).

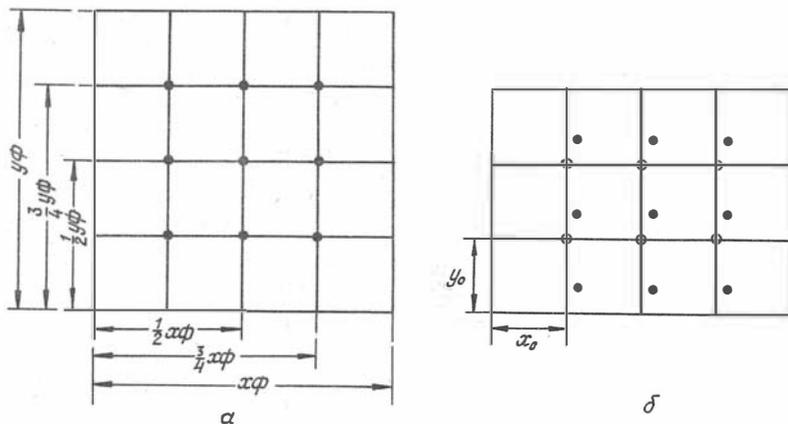


Рис. 31

Оператор K_3 заносит 0 в счетчик циклов. Всего будет 2 цикла - по оси x и по оси y . Оператор A_4 располагает x_0 в середине листа, т.е. фиксирует x формата = x_0 , x начала координат = 0. Оператор P_5 проверяет, автоматическое смещение или нет, что указывается в обращении к программе. Если смещение автоматическое, то работает оператор P_6 , который проверяет: $X_{max} < 0$? Если да, то это значит, что все значения x находятся влево от 0 и оператор A_7 засылает в x_0 значение $3/4 x$ формата, т.е. x_0 сместится вправо от серединн листа на $1/4$ формата.

Дальше работает оператор P_8 . Он анализирует отношение величин $\mathcal{D}X = |X_{max} - X_{min}|$, т.е. всего размаха графика по оси x к величине $|X_{min}|$. В некоторых случаях $\mathcal{D}X \ll X_{min}$ и тогда имеет смнсл расположить в начале координат не 0, а какую-то другую величину. Оператор P_8 проверяет: $\frac{\mathcal{D}X}{|X_{min}|} < \frac{1}{2}$? . Если да, то оператор A_9 производит подбор значения x , которое разумно расположить в x_0 . Если нет, то после оператора P_8 , как и после оператора A_9 , работает оператор P_{18} . Если после проверки оператора P_6 получится ответ $X_{max} > 0$, то за P_6 работает оператор P_{10} .

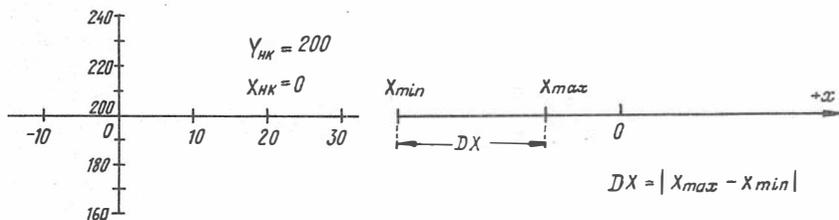


Рис.32

Оператор P_{10} проверяет: $X_{min} < 0$? . Если да, это значит, что значения X находятся по обе стороны от 0 и начало координат, помещенное в середину листа, перемещать не нужно. После оператора P_{10} работает оператор P_{18} . Если $X_{min} > 0$, это значит, что все значения X находятся вправо от 0.

Оператор A_{11} заснлает в x_0 значение $1/4 x$ формата, т.е. x_0 сместится влево от серединн листа на $1/4$ формата.

Оператор P_{12} проверяет: $\frac{\partial x}{|x_{max}|} < \frac{1}{2}$? Если да, то

оператор A_{13} производит подбор значения x , которое следует поместить в x_0 , а если нет, то после оператора P_{12} , как и после A_{13} , происходит переход на оператор P_{18} . Операторн $P_6 \div A_{13}$ работают в случае, когда смещение автоматическое. Если же смещение не автоматическое, то после оператора P_5 работает оператор P_{14} , который проверяет, задано ли смещение вообще. Если да, то оператор P_{15} проверяет, куда задано смещение. Если влево, то после P_{15} работает A_{16} , заснлающий в x_0 значение $1/4 x$ формата для смещения x_0 влево. Если вправо, то после P_{15} работает A_{17} , заснлающий в x_0 значение $3/4 x$ формата для смещения x_0 вправо. После операторов A_{16} , A_{17} , а также после P_{14} в случае, когда смещения нет, происходит переход на оператор P_{18} .

Оператор P_{18} проверяет, задана ли сетка. Если да, то оператор A_{19} уточняет x_0 до величинн, кратной шагу сетки. Если нет, то работает оператор K_{20} , который к счетчику циклов добавляет 1. Оператор K_{20} работает и после P_{18} , если сетка не предусмотрена. После K_{20} оператор P_{21} проверяет, какое значение в счетчике циклов. Если оно равно 1, значит, прокадрирована лишь одна ось. Оператор Φ_{22} формирует настройку на ось y и происходит переход на оператор A_4 , начиная с которого цикл повторяется, но для оси y . Если же в счетчике циклов число больше 1, значит, обе оси прокадрированн, работа окончена. Оператор Ψ_{23} - конец работн, уход в программу, из которой шло обращение к этому блоку.

Ст р у к т у р а о п е р а т о р н о й с х е м н п о д п р о г р а м м н " К а д р и р о в а н и е "

$H_0 A_1 \Phi_2 K_3^{22} A_4 P_5 \div 14 P_6 \div 10 A_7 P_8 \div 18 A_9^{18} P_{10}^{6,18} A_{11} P_{12} \div 18 A_{13}^{18} P_{14} \div 18 P_{15} \div 17$

$A_{16}^{18} A_{17}^{15} P_{18} \div 20^{8,9,10,12,13,14,16} A_{19}^{8} K_{20} P_{21}^{12,3} \Phi_{22}^y \Psi_{23}^{21,9}$

- H_0 - фиктивный оператор начала.
 A_1 - анализ данных точек (нахождение $X_{max}, X_{min}, Y_{max}, Y_{min}$).
 Φ_2 - формирование настройки на ось x .
 K_3 - занесение 0 в счетчик циклов.
 A_4 - занесение в x_0 - x формата/2 и 0 в начало координат.
 P_5 - проверка: автоматическое смещение или оно задано.
 P_6 - проверка: $X_{max} < 0$?
 A_7 - смещение начала координат вправо, т.е. $x_0 = \frac{3}{4} x_p$.
 P_8 - проверка отношения диапазона данных чисел $\frac{\Delta X}{K / X_{min}}$.
 A_9 - подбор начала координат $x_{нк}$.
 P_{10} - проверка: $X_{min} < 0$?
 A_{11} - смещение начала координат влево, т.е. занесение $x_0 = 1/4 x$ формата.
 P_{12} - проверка: $\frac{\Delta X}{|X_{max}|} < \frac{1}{2}$?
 A_{13} - подбор начала координат $x_{нк}$.
 P_{14} - проверка: смещение автоматическое?
 P_{15} - проверка: смещение влево?
 A_{16} - занесение в x_0 $1/4 x$ формата для смещения влево.
 A_{17} - занесение в x_0 $3/4 x$ формата для смещения вправо.
 P_{18} - проверка: задана ли координатная сетка?
 A_{19} - уточнение смещения до величины, кратной шагу сетки.
 K_{20} - прибавление 1 к счетчику циклов.
 P_{21} - проверка: счетчик циклов больше 1?
 Φ_{22} - формирование настройки для оси y .
 A_{23} - конец работн.

3. ОПИСАНИЕ БЛОКА "МАСШТАБИРОВАНИЕ" ПРОГРАММЫ "ГРАФИК"

О п и с а н и е р а б о т н о п е р а т о р н о й с х е м н

Целью настоящей подпрограммы является выбор масштаба. Масштаб выбирается из ряда чисел $(1, 2, 5) \times 10^k$, где k - целое число из условий конкретного расположения осей и нача-

ла координат по каждой оси (M_x , M_y).

Начало работы - фиктивный оператор H_0 . Оператор K_1 записывает 0 в регистр адреса оси (РА оси), который является счетчиком осей. 0 в регистре адреса соответствует настройке на ось x . Оператор P_2 выясняет, в какую сторону от начала координат расположена большая часть кривой, для чего он проверяет: $|x_{max}| > |x_{min}|$.

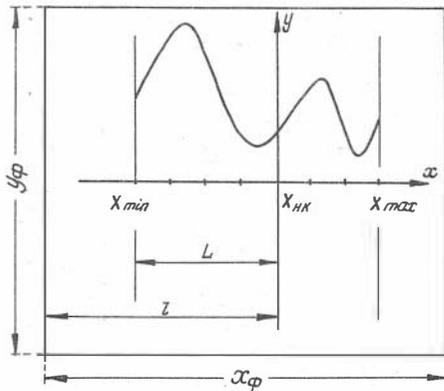


Рис.34

занять по оси x линейный размер $x_\phi - x_0$. Эта величина и записывается оператором A_6 в ячейку z . Если же знак „-“, то соответственно в ячейку z записывается величина x_0 оператором A_7 .

После операторов A_6 и A_7 работает оператор A_8 , который является блоком подбора масштаба по заданным величинам L и z . Работа этого блока рассматривается ниже. После оператора A_8 оператор A_9 отправляет полученное в A_8 значение масштаба по оси x в ячейку M_x . Оператор K_{10} добавляет к регистру адреса осей 1. Оператор P_{11} проверяет: $PA > 1$? Если нет, то происходит переход на оператор P_2 , начиная с которого повторяется тот же цикл для оси y . Если $PA > 1$,

Если да, то оператор A_3 записывает величину $|x_{max} - x_{HK}|$ в ячейку L , а если нет, то в ячейку L оператором A_4 записывается величина $|x_{min} - x_{HK}|$. L имеет размерность вычерчиваемой кривой по оси x . После операторов A_3 и A_4 оператор P_5 проверяет знак, который имеет большая часть кривой. Если знак „+“, то кривая на графике может

работа закончена, масштаб выбран по обеим осям. Оператор Я₁₂ - конец работн.

Блок подбора масштаба работает следующим образом. Н₁₃ - начальный оператор, вход в блок. Оператор А₁₄ засылает исходные данные, т.е. в ячейку М - единицу, в ячейку К - „0”, в ячейку 10^К - „1”, в регистр адреса М (РАМ) - „0”. Оператор Р₁₅ проверяет: $L \cdot M > Z$? Если да, то масштаб требуется уменьшить, т.к. иначе график не поместится на отведенном для него участке. Оператор А₁₆ уменьшает масштаб, т.е. выбирает из ряда масштабных чисел предидущее меньшее число. Оператор Р₁₇ снова проверяет: $L \cdot M > Z$? Если да, то оператор А₁₆ снова уменьшает масштаб.

Если после оператора Р₁₅ было нет, т.е. $L \cdot M < Z$, то оператор А₁₈ увеличивает масштаб, т.е. выбирает из ряда масштабных чисел следующее большее число, а затем происходит переход на Р₁₅, т.е. проверяется: $L \cdot M < Z$? Когда после проверки оператора Р₁₇ получается $L \cdot M > Z$, работа блока заканчивается оператором Я₁₉ и происходит возврат в программу "Масштабирование".

Структура операторной схемы подпрограммы "Масштабирование"

Н₀ К₁ Р₂₊₄ А₃⁵ А₄² Р₅₊₇ А₆⁸ А₇⁵ А₈⁶ А₉ К₁₀ Р₁₁₊₂ Я₁₂ ..

- Н₀ - фиктивный оператор начала.
- К₁ - засылка 0 в регистр адреса оси (для оси x).
- Р₂ - проверка: $|X_{max}| > |X_{min}|$?
- А₃ - занесение $|X_{max} - X_{HK}|$ в L .
- А₄ - занесение $|X_{min} - X_{HK}|$ в L .
- Р₅ - проверка знака .
- А₆ - занесение $|x_{\varphi} - x_0| \rightarrow Z$.
- А₇ - занесение $x_0 \rightarrow Z$.
- А₈ - блок подбора масштаба по L и Z .
- А₉ - занесение масштаба в рабочую ячейку M_x .

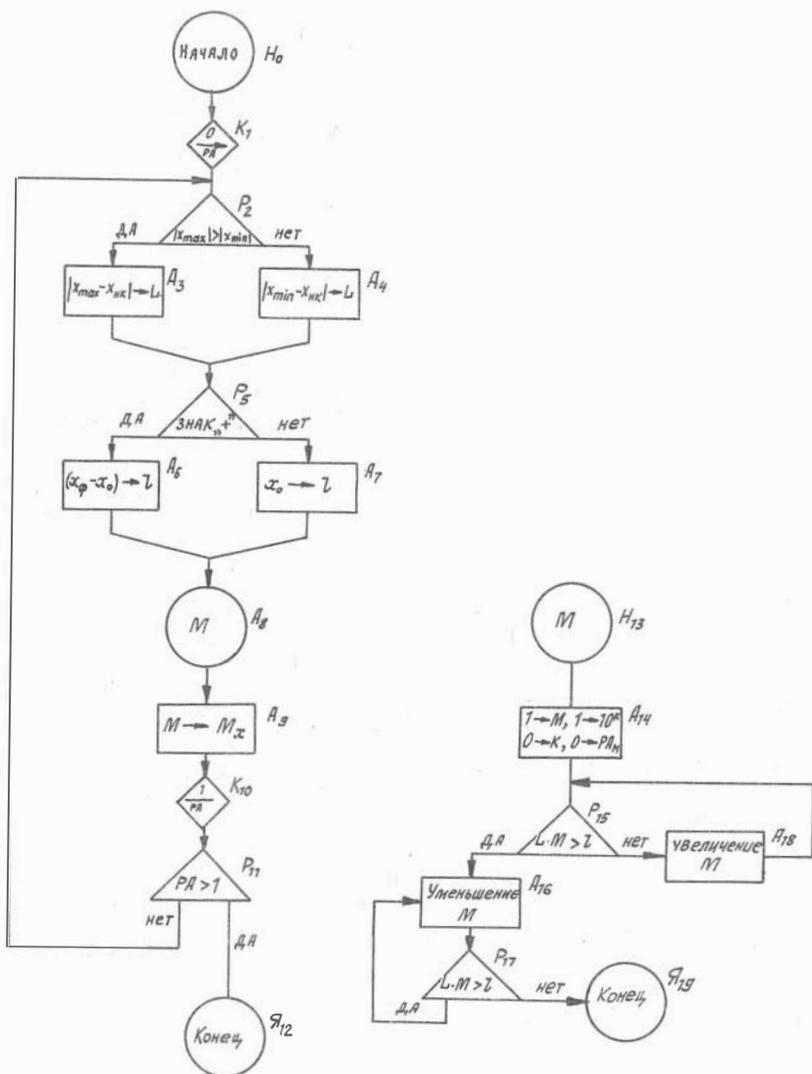


Рис. 35. Блок-схема программы «Масштабирование»

K₁₀ - прибавление 1 к РА для оси y .

P₁₁ - проверка: РА > 1?

П о д б л о к п о д б о р а м а с ш т а б а

H_{13} A_{14} P_{15+18} A_{16} P_{17+19} A_{18} A_{19} .

H₁₃ - начало.

A₁₄ - занесение исходных коэффициентов.

P₁₅ - проверка: $L \cdot M > L$?

A₁₆ - уменьшение М.

P₁₇ - проверка: $L \cdot M > L$?

A₁₈ - увеличение М.

A₁₉ - конец работн.

4. ОПИСАНИЕ БЛОКА "ВЫБОР ШАГА ОСЕЙ" ПРОГРАММЫ "ГРАФИК"

О п и с а н и е р а б о т н
о п е р а т о р н о й с х е м ы

Шаг осей выбирается либо из 4-х стандартных, тогда в A₂ третьей строки обращения стоит 0, либо по адресу, указанному в A₂.

Оператор H₀ - фиктивный оператор начала. Оператор P₁ проверяет: A₂ = 0 ? Если да, то оператор A₂ выбирает Δx , Δy по заданным из таблиц. Если нет, то оператор A₃ по указанному в обращении адресу выбирает Δx , Δy из МОЗУ.

После операторов A₂ и A₃ - конец работн, оператор A₄.

С т р у к т у р а о п е р а т о р н о й с х е м ы

H_0 P_{1+3} A_2 A_3 A_4 .

H₀ - фиктивный оператор начала.

P₁ - проверка: A₂ = 0 ?

- A_2 - выбор Δx , Δy из таблиц.
- A_3 - выбор Δx , Δy из МОЗУ.
- A_4 - конец работн.

5. ОПИСАНИЕ БЛОКА "РАМКА"

О п и с а н и е о п е р а т о р н о й с х е м н
п о д п р о г р а м м н " Р а м к а "

При обращении к подпрограмме анализируется условное число (оператор P_1), (задана ли рамка). Если рамка задана, то исполняется спуск пера и задаются последовательно координаты углов листа ($0, y_f; y_f, x_f; x_f, 0; 0, 0$) - оператор A_2 . Затем выполняется обращение к блоку УГ - оператор A_3 и осуществляется вычерчивание рамки. После оператора A_3 - конец работн - оператор A_4 .

С т р у к т у р а о п е р а т о р н о й с х е м н:

$H_0 P_1 A_2 A_3 A_4$.

- H_0 - начало.
- P_1 - проверка условия: вычерчивать ли рамку?
- A_2 - спуск пера, задание координат углов листа.
- A_3 - обращение к блоку УГ.
- A_4 - конец работн.

6. ОПИСАНИЕ БЛОКА "КООРДИНАТНАЯ ОСЬ" ПРОГРАММЫ "ГРАФИК"

О п и с а н и е р а б о т н
о п е р а т о р н о й с х е м н

Назначением данной подпрограммы является вычерчивание координатных осей с нанесением меток шагов Δx , Δy .

Начало работн - фиктивный оператор H_0 . Оператор K_1 засылает 1 в счетчик осей. Счетчик в данном случае считает

по $1/2$ оси, т.к. положительная и отрицательная полуоси вычерчиваются отдельно. Оператор A_2 заснает исходные значения $x_0 \rightarrow l, -\Delta x \rightarrow \Delta x$ рабочее, где l - счетчик длины оси, а Δx - шаг по оси x , выбранный ранее. Оператор A_3 через программу УГ осуществляет подъем пера. Оператор A_4 производит уход в $x_0 y_0$ также через программу УГ. Оператор A_5 осуществляет спуск пера, т.е. тоже обращается к программе УГ.

Дальше работает оператор K_6 , который из счетчика длины оси вычитает Δx . Оператор P_7 проверяет условие $l < 0$. Если нет, то оператор A_8 заснает $0 \rightarrow \Delta y_k, \Delta x_{\text{раб}} \rightarrow \Delta x_k$ и с этими данными обращается к программе УГ, которая осуществляет перемещение на $-\Delta x$. Оператор A_9 наносит метку. Вначале задается $\Delta y_k = y_{\text{метки}}, \Delta x_k = 0$ и выполняется верхняя часть метки, затем задается $\Delta y_k = -y_{\text{метки}}, \Delta x_k = 0$ и выполняется нижняя часть метки. Эти действия оператор производит через программу УГ.

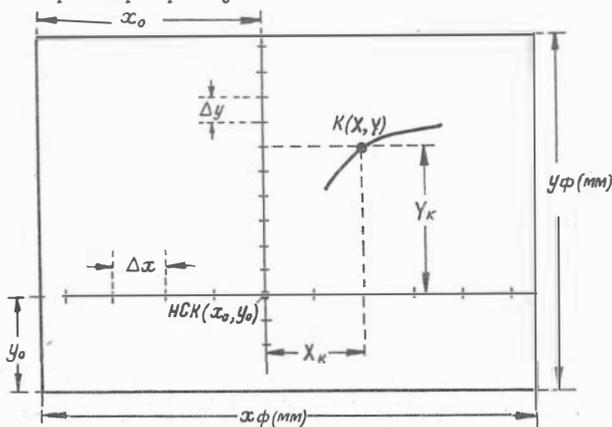


Рис. 36

После A_9 происходит уход на K_6 . Когда в счетчике длины осей появится $l < 0$, то после оператора P_7 работает оператор K_{10} , который прибавляет к счетчику осей 1 и заснает новое значение счетчика в его рабочую ячейку. Оператор P_{11} проверяет, выполнены ли обе половинки оси или только одна, т.е. проверяет условие: $K > 1$. Если нет, то выполнена только одна полуось. Оператор A_{12} подготавливает данные

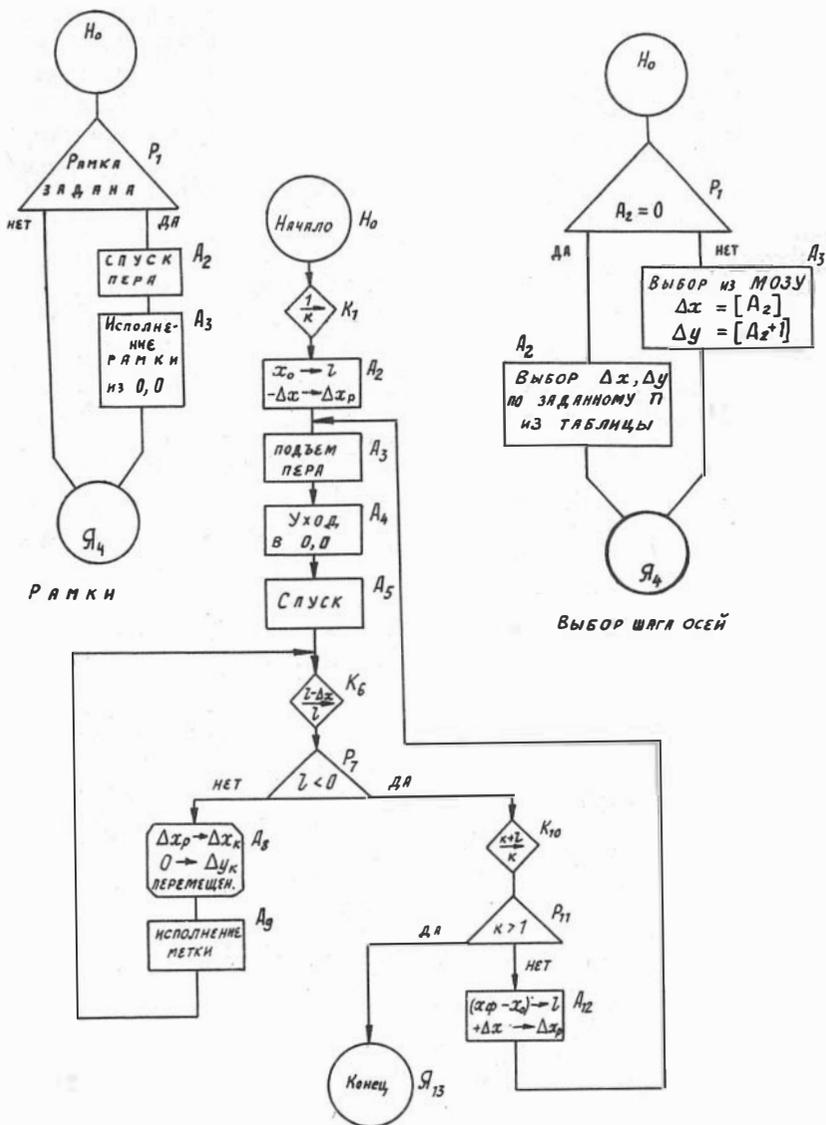


Рис. 37 Блок-схема программы „Координатная ось“

для вычерчивания второй полуоси. Для этого в ячейку z заносится $x_{\text{ф}} - x_0$, т.е. размах оси x вправо от оси y , а в ячейку $\Delta x_{\text{зад}} - \Delta x$. После этого происходит возврат на оператор A_3 , начиная с которого цикл повторяется. Когда в счетчике осей появится $K > 1$, значит, обе полуоси вычерчены, работа заканчивается. Оператор $Я_{13}$ - конец работы. Блок-схема алгоритма исполнения оси y аналогична.

Структура операторной схемы подпрограммы "Координатная ось"

- H_0 - начало работы.
- K_1 - занесение 1 в счетчик осей.
- A_2 - занесение $x_0 \rightarrow z, -\Delta x \rightarrow \Delta x_{\text{зад}}$.
- A_3 - подъем пера.
- A_4 - уход в x_0, y_0 .
- A_5 - спуск пера.
- K_6 - занесение $z - \Delta x \rightarrow z$.
- P_7 - проверка: $z < 0$?
- A_8 - перемещение на 1 рабочий шаг.
- A_9 - исполнение метки.
- K_{10} - прибавление 1 к счетчику осей.
- P_{11} - проверка: $K > 1$?
- A_{12} - занесение $x_{\text{ф}} - x_0 \rightarrow z, \Delta x \rightarrow \Delta x_{\text{зад}}$ для выполнения второй полуоси.
- $Я_{13}$ - конец работы.

$H_0 K_1 A_2^{2,12} A_3 A_4 A_5^9 K_6 P_7^{10} A_8 A_9^6 K_{10} P_{11}^{11} A_{12}^3 Я_{13}$.

7. ОПИСАНИЕ БЛОКА "КООРДИНАТНАЯ СЕТКА" ПРОГРАММЫ "ГРАФИК"

Описание работы операторной схемы

Целью настоящей подпрограммы является нанесение равномерной прямоугольной координатной сетки по заданным $\Delta x, \Delta y$.

Начало работы - фиктивный оператор H_0 . Оператор P_1 проверяет, задана ли сетка и, если она задана, оператор A_2 осуществляет спуск пера. Оператор A_3 заносит в рабочие ячейки исходные значения. В счетчик высот границ листа K заносится $y_{\text{ф}}$ (для вычерчивания горизонтальных линий), а в Δ раб. заносится „ $-x_{\text{ф}}$ “. Оператор K_4 заносит в счетчик границы величины $(K - \Delta y)$, а оператор P_5 проверяет условие $K < 0$. Если нет, то оператор A_{12} заснает $O \rightarrow \Delta x_k$, $\Delta y \rightarrow \Delta y_k$ для исполнения перемещения по оси y на величину Δy . Оператор A_{13} осуществляет это перемещение через программу УГ.

Оператор A_{14} заносит $\Delta \text{раб.} \rightarrow \Delta x_k$, $O \rightarrow \Delta y_k$ для вычерчивания горизонтальной линии, а также заносит $\Delta x_k \rightarrow \Delta \text{раб.}$ для следующего горизонтального хода. Оператор A_{15} вычерчивает горизонтальную линию. Происходит переход на оператор K_4 для исполнения следующего хода. Когда после оператора P_5 идет выполнение условия $K < 0$, то после P_5 работает оператор A_6 , который производит подъем пера. Оператор A_7 осуществляет переход в $0,0$, а оператор A_8 - спуск пера. Все эти операторы - A_6 , A_7 , A_8 работают через программу УГ. Дальше программа перестраивается для проведения вертикальных линий. Оператор A_9 заносит $x_{\text{ф}} \rightarrow k$, „ $-y_{\text{ф}}$ “ $\rightarrow \Delta \text{раб.}$ Оператор K_{10} заснает в счетчик величину $(K - \Delta x)$, а оператор P_{11} проверяет условие $K < 0$. Проведение вертикальных линий осуществляется аналогично горизонтальным. Если условие $K < 0$ не выполняется, то после оператора P_{11} работает оператор A_{16} , который заносит $\Delta x \rightarrow \Delta x_k$, $O \rightarrow \Delta y_k$ для перемещения по оси x на величину Δx . Это перемещение осуществляет оператор A_{17} через программу УГ.

Оператор A_{18} заснает $O \rightarrow \Delta x_k$, $\Delta \text{раб.} \rightarrow \Delta y_k$ для вычерчивания вертикальной линии и $\Delta y_k \rightarrow \Delta \text{раб.}$ для обратного хода по вертикали. Оператор A_{19} обращается к УГ и вычерчивает вертикальную линию. После этого происходит переход на оператор K_{10} , начиная с которого повторяется цикл вычерчивания линии.

Когда в счетчике K появится отрицательное число, значит длина листа, предоставленного для сетки, исчерпана. После оператора P_{11} при выполнении условия $K < 0$ работает оператор A_{20} , который осуществляет через программу УГ уход в $0,0$.

На этот же оператор передается управление и оператором P_1 , когда сетка не задана. После оператора A_{20} работа закончена. Оператор A_{21} - конец работы. Сетка вычерчена, перо перемещается в $0,0$.

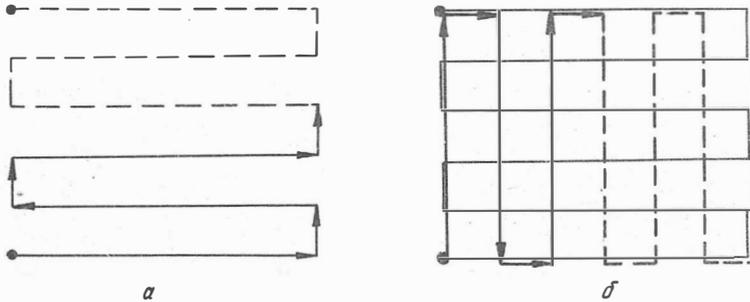


Рис. 39

Структура операторной схемы под программой "Координатная сетка"

$N_0 P_{1+20} A_2 A_3 K_4 P_{5+12} A_6 A_7 A_8 A_9 N_{10} P_{11+16}^{A_{20}} A_{12} A_{13} A_{14} A_{15} A_{16} A_{17} A_{18} A_{19}^{A_{20}} A_{21}$

- N_0 - фиктивный оператор начала.
- P_1 - проверка: задана ли сетка.
- A_2 - спуск карандаша.
- A_3 - подготовка счетчика ($K = y_{\text{ф}}$) и смещения ($\Delta_{\text{раб.}} = -x_{\text{ф}}$).
- K_4 - счетчик горизонтальных линий.
- P_5 - проверка условия $K < 0$.
- A_6 - подъем карандаша.
- A_7 - переход в $0,0$.
- A_8 - спуск карандаша.
- A_9 - подготовка счетчика ($K = x_{\text{ф}}$) и смещения ($\Delta_{\text{раб.}} = -y_{\text{ф}}$).
- K_{10} - счетчик вертикальных линий.
- P_{11} - проверка условия $K < 0$.
- A_{12} - занесение $\Delta x_k = 0$, $\Delta y_k = \Delta y$ для перемещения вдоль оси y на 1 шаг.

- A₁₃ - исполнение перемещения.
- A₁₄ - занесение $\Delta x_k = -\Delta_{рад}$, $\Delta y_k = 0$ для проведения горизонтальной линии и запоминание Δx_k в $\Delta_{рад}$. для следующего хода.
- A₁₅ - исполнение линии.
- A₁₆ - занесение $\Delta x_k = \Delta x$, $\Delta y_k = 0$ для сдвига вдоль оси x на i шаг.
- A₁₇ - исполнение перемещения.
- A₁₈ - занесение $\Delta x_k = 0$, $\Delta y_k = -\Delta_{рад}$ и запоминание Δy_k в $\Delta_{рад}$.
- A₁₉ - исполнение линии.
- A₂₀ - уход в 0,0.
- Я₂₁ - конец работн.

8. ОПИСАНИЕ БЛОКА "КОНТРОЛЬ ВЫЧЕРЧИВАНИЯ КРИВЫХ" ПРОГРАММЫ "ГРАФИК"

О п и с а н и е р а б о т н о п е р а т о р н о й с х е м ы

Задачей блока является преобразование реальных заданных величин X, Y в абсолютные перемещения в пределах заданного листа с учетом их расположения относительно координатных осей.

Начало работн - фиктивный оператор Н₀. Оператор А₁ выбирает символ по коду, указанному в обращении. Символ - условное обозначение точек данного графика. Оператор К₂ засчитывает 0 в счетчик очередных точек x, y . Оператор Р₃ проверяет, не наступил ли конец цикла, и в случае "нет" последнего работает оператор А₄, который выбирает очередную точку с координатами X_k, Y_k . Оператор К₅ прибавляет к счетчику точек 1.

Оператор А₆ подсчитывает абсолютную координату точки $x = x_0 + M_x (X_k - X_{HK})$. Абсолютная координата точки может оказаться за пределами листа. Оператор Р₇ проверяет условие $x < 0$, т.е. не вышла ли точка за лист влево. Если нет, то оператор Р₈ проверяет условие $x > x_{ф}$, т.е. не вышла

ла ли точка за лист вправо. Если точка вышла влево или вправо за пределы листа, то после операторов P_7 (при выходе влево) или P_8 (при выходе вправо) работает оператор A_{22} .

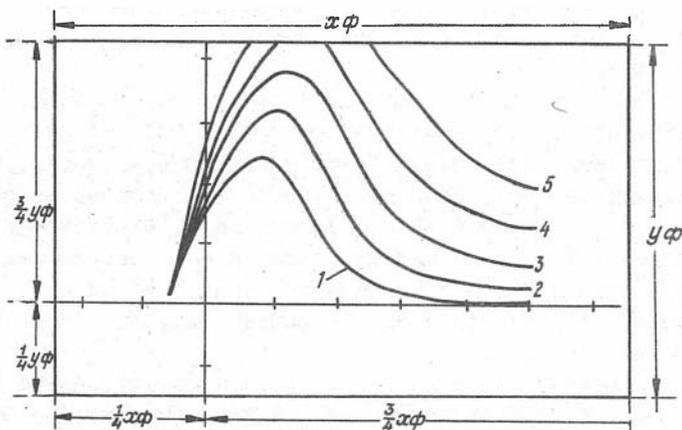


Рис. 40

Если точка не выходит за пределы листа, то после оператора P_8 работает оператор A_9 , который подсчитывает абсолютную координату точки $y = y_0 + My(Y_k - Y_{нк})$. Оператор P_{10} проверяет, не вышла ли точка за пределы листа вниз, т.е. проверяет условие $y < 0$. Если нет, то оператор P_{11} проверяет, не вышла ли точка за пределы листа вверх, т.е. проверяет условие $y > y_φ$. Если точка вышла вниз или вверх за пределы листа, то после операторов P_{10} (при выходе вниз) или P_{11} (при выходе вверх) работает оператор A_{22} . Оператор A_{22} запоминает, что было прерывание кривой, т.к. в любом случае попадания точки за пределы листа вычерчивание кривой прерывается. Карандаш стоит на месте до тех пор, пока не произойдет возвращения кривой в пределы листа в какой-то новой точке.

После оператора P_{11} , если точка не вышла вверх за пре -

делу листа, работает оператор P_{12} , который проверяет, было ли прерывание кривой. Если кривая прерывалась, то после оператора P_{12} оператор A_{13} осуществляет подъем карандаша, оператор A_{14} выполняет перевод его в точку x, y , а оператор A_{15} производит спуск карандаша. Все эти операторы A_{13}, A_{14}, A_{15} работают через программу УГ. После оператора A_{15} работает оператор A_{16} , который дает команду убрать из памяти прерывание кривой.

Если же кривая не прерывалась, то после оператора P_{12} работает оператор P_{17} , который проверяет, чертить ли кривую между символами. Если чертить, то оператор A_{18} производит спуск карандаша, если нет, то оператор A_{19} производит подъем карандаша. И спуск и подъем карандаша осуществляется через программу УГ. Через нее же оператор A_{23} , работающий после операторов A_{18} и A_{19} , производит ход до точки x, y . Оператор A_{24} обращается к блоку рисовки символа, чтобы пометить точку на графике.

После операторов A_{16}, A_{21} и A_{22} происходит возвращение к оператору P_3 , проверяющему, не кончился ли цикл. Конец цикла — это конец работы, оператор P_{23} .

Структура операторной схемы под программы "Контроль в черчении кривых"

- H_0 — фиктивный оператор начала.
- A_1 — выбор кода символа (из обращения).
- K_2 — засылка 0 в счетчик очередных точек.
- P_3 — проверка условия: конец цикла?
- A_4 — выбор очередных X_k, Y_k .
- K_5 — прибавление 1 к счетчику точек.
- A_6 — подсчет абсолютной координаты $x = x_0 + Mx(X_k - X_{HK})$.
- P_7 — проверка условия $x < 0$ (контроль крайней левой точки).
- P_8 — проверка условия $x > x_{\varphi}$ (контроль крайней правой точки).
- A_9 — подсчет абсолютной координаты $y = y_0 + My(Y_k - Y_{HK})$.

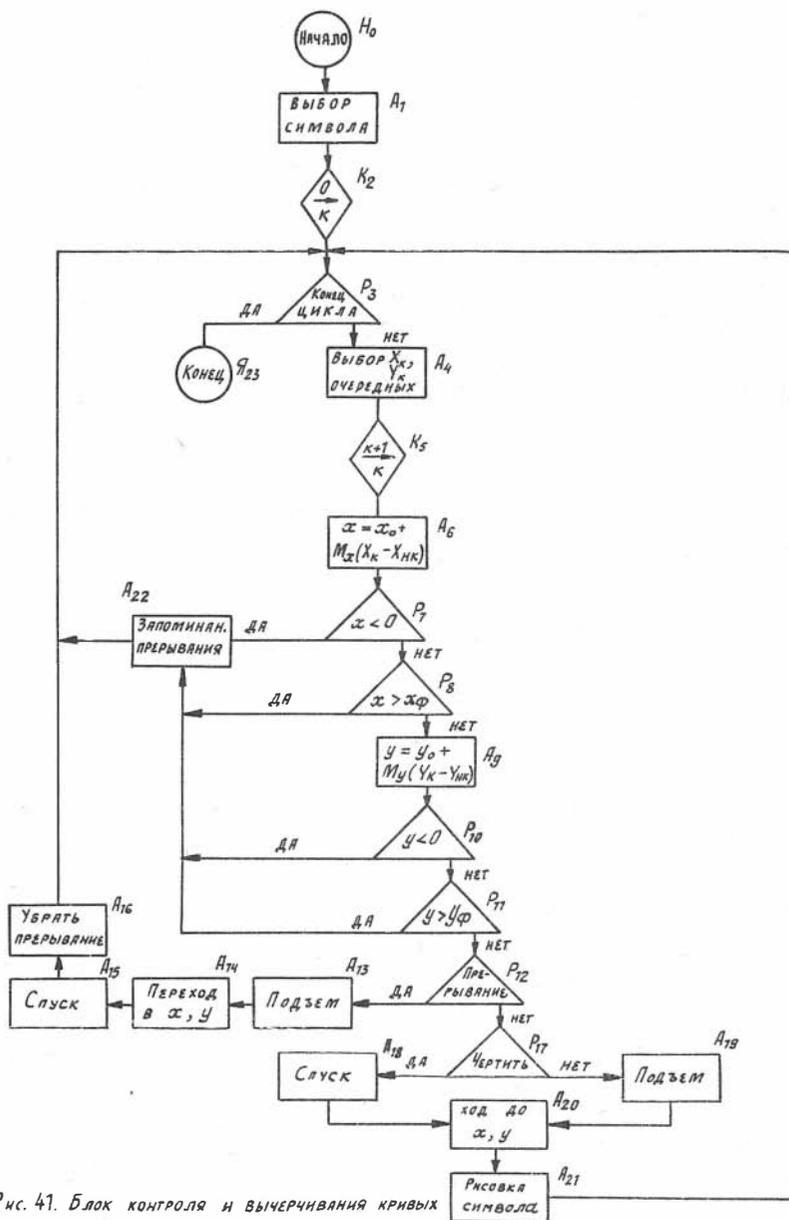


Рис. 41. Блок контроля и вычерчивания кривых

- P_{10} - проверка условия $y < 0$ (контроль нижней крайней точки).
 P_{11} - проверка условия $y > y_{\text{ф}}$ (контроль верхней крайней точки).
 P_{12} - проверка: был ли выход кривой за пределы листа.
 A_{13} - подъем карандаша.
 A_{14} - переход в точку x, y .
 A_{15} - спуск карандаша.
 A_{16} - убрать прерывание кривой.
 P_{17} - проверка: нужно ли чертить кривую между символами.
 A_{18} - спуск карандаша.
 A_{19} - подъем карандаша.
 A_{20} - ход до точки x, y .
 A_{21} - обращение к блоку рисовки символов.
 A_{22} - запоминание прерывания.
 A_{23} - конец работы.

$H_0, A_1, K_2, {}^{16, 21, 22}P_3, {}^{125}A_4, K_5, A_6, P_{7+22}, P_{8+22}, A_9, P_{10+22}, P_{11+22}, P_{12}^{17}, A_{13}, A_{14}, A_{15}, A_{16}^3$

${}^{12}P_{17}, {}^{119}A_{18}, {}^{20}A_{19}, {}^{17}A_{20}, {}^{20}A_{21}^3, 7, 8, 10, 11, A_{22}^3, {}^3A_{23}$

9. ИНСТРУКЦИЯ К ПРОГРАММЕ "ГРАФИК"

Программа была опробована на машине М 20. Обращение выбрано аналогично обращению к стандартным программам в системе ИС-2.

Адрес	Код	A ₁	A ₂	A ₃
x	016	$x+1$	2002	7610
$x+1$	000	x	Р СП	y
$x+2$	000	УЧ	$\Delta x, (\Delta y)$	z

К - ключ режимов (3 разряда слева во 2-ой строке обращения). Задействовано 3 разных режима, в зависимости от номера режима вся остальная информация во 2-ой и 3-ей строке обращения будет расшифровываться различно.

Р е ж и м К₁

К₁ - если есть "1" (в 45-ом разряде) - режим смены листа и выполнения оформления.

Смысловая часть обращения передается через УЧ - (условное число, которое занимает 12 разрядов A₁ 3-ей строки).

У с л о в н о е ч и с л о

A ₁											
36	35	34	33	32	31	30	29	28	27	26	25
номер формата			$xл$	$xл$	$ул$	$ул$	оси сетка	рамка "1"	станд. шаг оси		
			смещение осей								

3 разряда (34 ÷ 36) указывают номер выбираемого формата – восьмеричные цифры (0 ÷ 7).

4 разряда отводятся для указания смещения осей. Если геолог знает характер выводимой функции, он может задать одно из девяти расположений точки пересечения осей (в центре листа или смещенной на 1/4 вверх, вниз, влево, вправо). Для каждой оси по 2 разряда

Смещение осей

Код		Ось x	Код		Ось y
33	32		31	30	
0	0	центр	0	0	центр
1	0	влево	1	0	вниз
0	1	вправо	0	1	вверх
1	1	выбор автомат.	1	1	выбор автомат.

Предусмотрен автоматический поиск оптимального расположения осей на основе конкретной числовой информации.

29-й разряд – выбор: координатные оси ("0") или координатная сетка ("1").

28-й разряд – если требуется рамка, ограничивающая лист.

В 26-м и 27-м разрядах указывается номер шага меток на осях (восьмеричное число 0 ÷ 3).

0 – шаг меток	$\Delta x = \Delta y = 1$ см
1 – "-"	$\Delta x = \Delta y = 2$ см
2 – "-"	$\Delta x = \Delta y = 5$ см
3 – "-"	$\Delta x = \Delta y = 10$ см

Примечание: Если геолога не удовлетворяет ни один из стандартных шагов, он может вводить любой нестандартный шаг. Для этого необходимо поместить в какой-то ячейке Δx , в ячейке следующей – Δy и указать в A_2 3-й строки обращения адрес ячейки Δx . Если $A_2 = 0$, то выбирается стандартный шаг, указанный 26, 27 разрядом; если $A_2 \neq 0$, то Δx , Δy выбирается из указанного адреса.

25-й разряд ÷ резервный.

A_2 второй строки обращения не задействован.

A_1 2-ой строки обращения - адрес начала массива X -ов.
 A_2 2-ой строки обращения - адрес начала массива Y -ов.
 A_3 3-ей строки обращения - длина массива.

Массив X и Y нужен в K_1 режиме только для анализа и автоматического выбора смещения осей, а также для кадрирования в пределах каждой оси, для автоматического масштабирования. Но само вычерчивание графика будет только по третьему режиму (K_3).

Режим K_2 - н а д п и с и.

Режим K_2 - если есть "1" 44 р. 2-ой строки обращения и нет "1" 45-го разряда. K_1 и K_2 несовместимы, так как у них слишком различная трактовка задающих слов.

Здесь x, y - адреса, в которых указаны координаты начала текста в абсолютных значениях выбранного листа (т.е. с отсчетом от левого нижнего угла листа). А если $x = y = 0$, то это означает, что текст начинать с места стояния карандаша.

$\Delta x, \Delta y$ (адрес следующей ячейки) - адреса чисел, указывающих угол наклона строки (как вектор, указывающий направление из 0,0 в точку $\Delta x, \Delta y$).

Вместо $УЧ$ ставить адрес начала массива текста. Z - длина массива.

Режим K_3 - к о н т р о л ь и в ы ч е р ч и в а н и е г р а ф и к о в.

"1" 43 разряда. Если есть еще и "1" 45 разряда, то выполнится сначала режим K_1 , потом K_3 . Если "1" 43 разряда и "1" 44 разряда во 2-ой строке, то выполняется только режим K_2 , а режим K_3 выполняться не будет.

x, y во 2-ой строке для K_3 режима означают адреса начала массива X -ов и Y -ов соответственно. Z - длина массива.

10. ПРОГРАММА "ГРАФИК"

2000	252	2777	0000	7601	к блоку фиксации
1.	016	2002	7617	7625	
2.	016	2060	7573	7601	БЭИ начало
3.	000	0000	0054	0000	
4.	007	0000	0000	0000	УЧ (услов.число,сдвинуто)
5.	000	0000	0000	0000	рабочая ячейка
6.	107	6200	0000	0000	l_0 длина формата # 0
7.	107	4540	0000	0000	h_0 высота формата # 0
2010	101	4000	0000	0000	x_0 длина выбранного формата
1.	101	4000	0000	0000	y_0 высота выбранного формата
2.	002	0000	0000	0000	x_{k-1} координаты точки стояния
3.	003	0000	0000	0000	y_{k-1} карандаша (в шагах ГП)
4.	001	0000	0000	0000	Δx шаг меток координатных
5.	001	0000	0000	0000	Δy осей (в мм)
6.	001	0000	0000	0000	x_0 координаты точки пересеч.
7.	001	0000	0000	0000	y_0 осей в абсолютн. коорд. л-та
2020	001	0000	0000	0000	$x_{нк}$ числ.знач.в начале осей
1.	001	0000	0000	0000	$y_{нк}$ (точке пересечения)
2.	001	0000	0000	0000	x_k координаты очередной точки
3.	001	0000	0000	0000	y_k
4.	001	0000	0000	0000	x_{max}
5.	001	0000	0000	0000	y_{max}
6.	001	0000	0000	0000	x_{min}
7.	001	0000	0000	0000	y_{min}
					предельные числовые значения вводимого графика
2030	002	0000	0000	0000	Δx_{k-1}
1.	002	0000	0000	0000	Δy_{k-1}
2.	101	0000	0000	0000	Δx_k
3.	101	0000	0000	0000	Δy_k
4.	002	0000	0000	0000	Δx_{k+1}
5.	002	0000	0000	0000	Δy_{k+1}
6.	002	0000	0000	0000	Δx диапазон графика по
7.	002	0000	0000	0000	Δy соответствующим осям
2040	002	0000	0000	0000	счетчик осей
1.	002	0000	0000	0000	Δp рабочий
2.	101	4000	0000	0000	M_x масштабные коэффициенты
3.	101	4000	0000	0000	M_y
4.	003	0000	0000	0000	x_{k-1}
5.	003	0000	0000	0000	y_{k-1}
6.	075	6314	6314	6315	$h = 0,1$ мм величина шага граф.
7.	003	0000	0000	0000	Δ ведущий
2050	003	0000	0000	0000	Δ ведомый
1.	003	0000	0000	0000	δ ведущий, δ ведомый
2.	003	0000	0000	0000	$\beta = \Delta$ ведом./ Δ ведущ.
3.	003	0000	0000	0000	$C \pm \Delta$ (счетчик отрезка)
4.	003	0000	0000	0000	R ошибка
5.	003	0000	0000	0000	
6.	003	0000	0000	0000	
7.	003	0000	0000	0000	

2060	055	2111	7704	2111	
1.	013	2111	7604	2111	формирование блока выборки данных
2.	055	2112	7704	2112	
3.	054	0130	7607	7607	
4.	013	2112	7607	2112	
5.	016	2714	7573	7601	БЭИ 2 (3-я строка информации) ключ режима текста
6.	070	0000	0000	0000	
7.	054	0111	7604	2004	сдвиг
2070	055	2004	7714	0000	есть смена листа?
1.	036	0000	2100	2106	
2.	016	2125	2073	2106	обеспеч. уход на смену листа $A_2 = 0?$
3.	055	7616	7732	0000	
4.	036	0000	2210	0000	
5.	072	0000	7616	0000	$A_2 \rightarrow PA$
6.	400	0000	0000	2014	выборка Δx
7.	400	0001	0000	2015	выборка Δy
2100	054	0130	7607	0001	формулирование конца цикла
1.	013	2123	0001	2114	
2.	000	0000	0000	0000	
3.	000	0000	0000	0000	
4.	032	0000	7626	2105	блок запоминания
5.	052	0000	0000	0000	
6.	007	0000	0000	0000	на смену листа
7.	056	0000	2644	0000	без сменн листа
2110	052	0000	0077	0000	PA очередн.
1.	400	0000	0000	2022	Блок выборки очередных X_k, Y_k
2.	400	0000	0000	2023	
3.	652	0001	0000	2110	X_k Y_k
4.	012	3000	2122	0000	+1 PA
5.	000	0000	0000	0000	конец массива?
6.	032	0000	2117	2105	возврат
7.	016	2120	7634	7601	блок воспроизведения
2120	016	7610	7600	7601	Концовка
1.	000	0000	0000	0000	в ИС и уход в программу
2.	016	0000	7747	0000	
3.	012	0000	2122	0000	с очередным X_k, Y_k
4.	000	0000	0000	0000	
5.	000	0000	0000	0001	$O \rightarrow x_{сд}$ начало
6.	055	2005	7714	0000	был справа?
7.	036	2006	2131	2010	$Z_o \rightarrow x_{сф}$
2130	002	0000	2006	0001	$x_{сф} \rightarrow x_{сд}$
1.	055	2004	7713	0002	формат № 0
2.	036	2007	2145	2011	
3.	001	2007	2007	2011	формат № 1
4.	015	0002	7711	0000	
5.	036	0000	2145	0000	
6.	001	2006	2006	2010	формат № 2
7.	015	0002	7712	0000	

2140 036 7761 2142 0000
 1. 001 2011 2011 2011
 2. 000 0000 0000 2005
 3. 002 0000 2011 2023
 4. 056 0001 2152 2022
 5. 015 0001 0000 0000
 6. 076 0000 2142 0000
 7. 000 2006 0000 2022
 2150 000 0000 0000 2023
 1. 075 2005 7714 2005
 2. 016 2153 3173 3177
 3. 000 7761 0000 2042

формат \mathbb{R}^2
 формат \mathbb{R}^3
 убрать признак правой стороны
 $Y_k = -Y_{\varphi}$ (вниз или влево-вниз)
 $x_k = -x_{c\varphi}$ (орг.сдвига по оси X)
 можно ли вправо?
 $x_k = l_{\varphi}$
 $Y_k = 0$
 запомнить признак прав.стороны
 подъем гера
 $1 \rightarrow M_x$

4. 000 7761 0000 2043
 5. 016 2156 3224 3212
 6. 000 0000 0000 2012
 7. 000 0000 0000 2013
 2160 000 0000 0000 2044
 1. 000 0000 0000 2045
 2. 016 2203 2163 2115
 3. 052 0000 0000 2110
 4. 016 2165 2111 2122
 5. 000 2022 0000 2024
 6. 000 2022 0000 2026
 7. 000 2023 0000 2025

$1 \rightarrow M_y$
 сменить лист
 $0 \rightarrow \tilde{x}_{i-1}$
 $0 \rightarrow \tilde{y}_{i-1}$ точке стояния каран-
 $0 \rightarrow x_{i-1}$ даша присваивается
 $0 \rightarrow y_{i-1}$ абсол.координата 0,0
 засылка возврата
 $0 \rightarrow PA$ очередное
 за X_1, Y_1
 $X_1 \rightarrow X_{max}$
 $X_1 \rightarrow X_{min}$
 $Y_1 \rightarrow Y_{max}$

Анализ исходн. данных

2170 000 2023 0000 2027
 1. 016 2172 2110 2122
 2. 052 0000 0000 0000
 3. 602 2024 2022 0000
 4. 076 0000 2176 0000
 5. 500 2022 0000 2024
 6. 602 2022 2026 0000
 7. 076 0000 2201 0000
 2200 500 2022 0000 2026
 1. 112 0001 2173 0001
 2. 016 2172 2110 2122
 3. 000 0000 0000 0000

$Y_1 \rightarrow Y_{min}$
 за очередным X_k, Y_k
 $X_k > X_{max} ?$
 $X_k \rightarrow X_{max}$
 $X_k < X_{min}$
 $X_k \rightarrow X_{min}$
 цикл для Y_k
 цикл по всем X_k, Y_k

4. 052 0001 0000 2235
 5. 056 0000 2360 0000
 6. 007 0000 0000 0000
 7. 000 0000 0003 0000
 2210 054 0053 2004 0001
 1. 055 0001 2207 0001
 2. 072 0000 0001 0000
 3. 405 2300 2303 2014
 4. 405 2300 2303 2015
 5. 056 0000 2100 0000
 6. 401 2024 0000 0000
 7. 076 0000 2223 0000

на выбор позиции
 начало
 на смену листа
 $X_{max} < 0 ?$

Выбор шага осей $\Delta x, \Delta y$

Анализ X, Y экстремальных

2220 404 2016 7762 0001
 1. 501 2016 0001 2016
 2. 056 0000 2226 0000
 3. 401 2026 0000 0000
 4. 036 0000 2226 0000
 5. 504 2016 7762 2016
 6. 054 0107 2004 0001
 7. 055 0001 7714 0000
 2230 036 0000 2235 0000
 1. 604 2016 2014 0001
 2. 061 0001 7753 0001
 3. 001 0001 0000 0001

все X вправо от 0

$$x_0 = \frac{3}{4} x_{\phi}$$

$$x_{min} < 0 ?$$

все X слева $x_0 = \frac{1}{4} x_{\phi}$

сдвиг YЧ

сетка задана?

уточнение координат
 начала оси

4. 305 0001 2014 2016
 5. 052 0000 0001 0000
 6. 077 0000 0000 0000
 7. 007 0000 0000 0000
 2240 052 0000 0000 0000
 1. 702 2024 2026 2036
 2. 202 0000 2036 0000
 3. 176 0000 2271 2020
 4. 000 0000 0000 0000
 5. 000 0000 0000 0000
 6. 202 0000 2026 0000
 7. 076 0000 2256 0000

X_0 кратно шагу сетки
 возврат (на ось Y)

$$x_{max} - x_{min} = \Delta X$$

Числовое значение
 в начале координат

$$x_{min} > 0$$

(ось вся „+“)

2250 604 2036 2024 0001
 1. 003 0001 7764 0000
 2. 076 0000 2267 0000
 3. 400 2026 0000 2305
 4. 016 2255 2273 2277
 5. 104 7761 2304 2020
 6. 401 2024 0000 0000
 7. 076 0000 2267 0000
 2260 604 2036 2026 0001
 1. 003 0001 7764 0000
 2. 076 0000 2267 0000
 3. 003 2024 0000 2305

$$|\Delta X / x_{max}| < 0,5 ?$$

$$x_{min} \rightarrow L$$

подбор из табл. ближайш. значения
 $x_{нк}$ в сторону уменьш.

$$x_{max} < 0 ?$$

$$|\Delta X / x_{min}| < 0,5 ?$$

(ось вся „-“)

$$x_{max} \rightarrow L$$

4. 016 2265 2273 2277
 5. 104 7761 2304 2020
 6. 000 0000 0000 0000
 7. 007 0000 0000 0000
 2270 000 0000 0000 0000
 1. 000 0000 0000 0000
 2. 077 0000 0000 0000
 3. 000 2300 0000 2306
 4. 452 0000 0000 2276
 5. 016 2276 2322 2353
 6. 052 0000 0000 0000
 7. 070 0000 0000 0000

подбор $x_{нк}$ из таблиц

(возврат)

резерв для $\Delta X = 0$

РА

вспомогательный
 блок для поиска
 "круглых" знач.

возврат

2300	101	4000	0000	0000	1	Блок подбора масштаба
1.	102	4000	0000	0000	2	
2.	103	5000	0000	0000	5	
3.	104	5000	0000	0000	10	
4.	101	4000	0000	0000	M	
5.	101	4000	0000	0000	L (заданная величина)	
6.	101	4000	0000	0000	L _K линейный размер	
7.	101	4000	0000	0000	10 ^K	
2310	101	0000	0000	0000	счетчик K	
1.	007	0000	0000	0000	1 → M	начало
2.	000	2300	0000	2304	1 → 10 ^K	
3.	000	2300	0000	2307		
4.	000	0000	0000	2310	0 → K	
5.	052	0000	0000	2324	0 → PAM	
6.	016	2321	2317	2333	проверять после увеличения	
7.	016	2350	2320	2345	проверять после уменьшения	
2320	007	0000	0000	0000		
1.	005	2305	2304	0001		
2.	003	0001	2306	0000		уменьшать M
3.	076	0000	2336	0000		
4.	052	0000	0000	0000	PAM (масштаба)	Увеличение M
5.	405	2300	2307	2304	(1V2V5) x 10 ^K → M	
6.	652	0001	0001	2324	+1 PAM	
7.	112	0004	2333	0000	на проверку	
2330	005	2307	2303	2307	10 ^K · 10 → 10 ^K	
1.	001	2310	2300	2310	K + 1 → K	
2.	032	0000	2325	0000		
3.	000	0000	0000	0000	(возврат)	
4.	000	0000	0000	0000		
5.	000	0000	0000	0000		
6.	072	0000	2324	0000	PAM	Уменьшение M
7.	652	7777	0000	2324		
2340	112	0003	2344	0000	перебор (5V2V1)	
1.	004	2307	2303	2307	10 ^K /10 → 10 ^K	
2.	002	2310	2300	2310	K - 1 → K	
3.	032	0000	2337	0002	PA = 2	
4.	405	2300	2307	2304	(5V2V1) x 10 ^K → M	
5.	070	0000	0000	0000	(возврат)	
6.	000	0000	0000	0000		
7.	000	0000	0000	0000		
2350	005	2305	2304	0001	(L · M) > L	
1.	003	0001	2306	0000		
2.	076	0000	2336	0000		
3.	070	0000	0000	0000		
4.	000	0000	0000	0000	возврат (M найден)	
5.	000	0000	0000	0000		
6.	000	0000	0000	0000		
7.	000	0005	0000	0000		

2360	054	0102	2004	0003	} Кадрирование		
1.	052	0000	0000	0000		подготовка УЧ $x_{\varphi/2} \rightarrow x_o$ $0 \rightarrow x_{нк}$ } середина листа смещение автоматическое?	
2.	504	2010	7762	2016			
3.	055	0003	7713	0003			
4.	136	0000	2371	2020			
5.	015	0003	7713	0000			
6.	076	0000	2400	0000			
7.	016	2370	2216	2235			
2370	016	2371	2241	2267			} числовое значение
1.	054	0104	2004	0003			
2.	112	0001	2362	0001			
3.	003	0000	0000	0000			
4.	056	0000	2614	0000	}		
5.	003	0000	0000	0000			
6.	003	0000	0000	0000			
7.	003	0000	0000	0000			
2400	015	0003	7714	0000		} смещение влево?	
1.	076	0000	2404	0000			
2.	016	2371	2225	2235		}	
3.	003	0000	0000	0000			
4.	016	2371	2220	2235	} смещение вправо		
5.	003	0000	0000	0000			
6.	003	0000	0000	0000			
7.	001	0000	0000	0000			
2410	016	2411	3170	3177	} рамка задана?		
1.	055	2004	2407	0000			
2.	036	0000	2431	2427			
3.	000	0000	0000	2032		} $\Delta x_k = 0$ $\Delta y_k = y_{\varphi}$ } левая граница	
4.	000	2011	0000	2033			
5.	016	2416	3215	3212		} обращение к УГ	
6.	000	2010	0000	2032			
7.	000	0000	0000	2033	} верхняя граница		
2420	016	2421	3215	3212		} правая граница	
1.	000	0000	0000	2032			
2.	002	0000	2011	2033			
3.	016	2424	3215	3212	} нижняя граница		
4.	002	0000	2010	2032			
5.	000	0000	0000	2033			
6.	016	2427	3215	3212			
7.	000	0000	0000	0000		} (возврат для сетки)	
2430	002	0000	0000	0000			} задана сетка? $x_{\varphi} \rightarrow$ счетчик оси $-x_{\varphi} \rightarrow \Delta P$ (рабочее) Сг оси - $\Delta y < 0$?
1.	055	2004	2430	0000			
2.	036	0000	2500	0000			
3.	000	2011	0000	2040			
4.	062	0000	2010	2041			
5.	000	0000	0000	0000			
6.	003	2040	2015	2040			
7.	036	0000	2450	0000			

2440	000	0000	0000	2032	$0 \rightarrow \Delta x_k$	} шаг сетки вверх
1.	000	2015	0000	2033	$\Delta y \rightarrow \Delta y_p$	
2.	016	2443	3215	3212	обращение к УГ	} вдоль оси X
3.	062	0000	2041	2032	смена направления	
4.	000	0000	0000	2033		} обращение к УГ и снова нач.цикл.
5.	000	2032	0000	2041		
6.	016	2436	3215	3212		
7.	000	0000	0000	0000		
2450	016	2451	3173	3177	подъем пера	
1.	000	0000	0000	2022	$0 \rightarrow x_k$	} уход в 0,0
2.	000	0000	0000	2023	$0 \rightarrow y_k$	
3.	016	2454	3224	3212		
4.	000	2010	0000	2040	$y_p \rightarrow$ счетчик оси	
5.	062	0000	2011	2041	Δ рабочее	
6.	016	2457	3170	3177	спуск пера	
7.	003	2040	2014	2040	счетчик оси $-\Delta x < 0?$	
2460	036	0000	2472	0000		
1.	000	2014	0000	2032		} шаг сетки вправо
2.	000	0000	0000	2033		
3.	016	2464	3215	3212		} смена направления
4.	000	0000	0000	2032		
5.	002	0000	2041	2033		} вдоль оси Y
6.	000	2033	0000	2041		
7.	016	2457	3215	3212		
2470	000	0000	0000	0000		
1.	000	0000	0000	0000		} возврат в 0,0
2.	000	0000	0000	2022		
3.	000	0000	0000	2023		
4.	016	2475	3173	3177		
5.	016	2606	3224	3212		
6.	070	0000	0000	0000		
7.	101	4000	0000	0000		
2500	016	2501	3173	3177	подъем пера	
1.	000	2016	0000	2022	$x_o \rightarrow x_k$	} до точки пересеч. осей
2.	000	2017	0000	2023	$y_o \rightarrow y_k$	
3.	016	2504	3224	3212		
4.	016	2505	3170	3177	спуск пера	
5.	002	2010	2016	2040	$x_p - x_o \rightarrow$ счетчик оси X	} Ось "+X"
6.	004	2015	2303	2477	$0,7\Delta y \rightarrow$ метка (размер)	
7.	000	0000	0000	0000		
2510	003	2040	2014	2040	конец оси?	
1.	036	0000	2522	2033	$0 \rightarrow$	
2.	000	2014	0000	2032	$\Delta x \rightarrow \Delta x_k$	
3.	016	2514	3215	3212	исполнение (УГ)	
4.	000	0000	0000	2032	$0 \rightarrow \Delta x_k$	
5.	002	0000	2477	2033	-метка $\rightarrow \Delta y_k$	
6.	016	2517	3215	3212	исполнение метки (УГ)	
7.	000	2477	0000	2033		

2520	016	2510	3215	3212		цикл
1.	000	0000	0000	0000	$\left. \begin{array}{l} x_0 \rightarrow x_k \\ y_0 \rightarrow y_k \end{array} \right\}$	возврат в точку пересечения осей
2.	000	2016	0000	2022		
3.	000	2017	0000	2023	счетчик оси	Ось "-X"
4.	016	2525	3224	3212		
5.	000	2016	0000	2040		
6.	000	0000	0000	0000		
7.	003	2040	2014	2040		
2530	036	0000	2542	0000	конец оси?	
1.	002	0000	2014	2032	$\left. \begin{array}{l} \text{"}\Delta x\text{"} \rightarrow \Delta x_k \\ 0 \rightarrow \Delta y_k \end{array} \right\}$	шаг по оси
2.	000	0000	0000	2033		
3.	016	2534	3215	3212		
4.	000	0000	0000	2032	метка	
5.	002	0000	2477	2033		
6.	016	2537	3215	3212		
7.	000	2477	0000	2033		
2540	016	2527	3215	3212	цикл	
1.	000	0000	0000	0000	спуск	переход в точку пересечения осей
2.	000	2016	0000	2022		
3.	000	2017	0000	2023		
4.	016	2545	3224	3212		
5.	016	2546	3170	3177		
6.	002	2011	2017	2040		
7.	004	2014	2303	2477		
					начальные установки	Ось "+Y"
2550	000	0000	0000	0000		
1.	003	2040	2015	2040	конец оси?	
2.	036	0000	2563	2032		
3.	000	2015	0000	2033	шаг по оси	
4.	016	2555	3215	3212		
5.	002	0000	2477	2032		
6.	000	0000	0000	2033	метка	
7.	016	2560	3215	3212		
2560	000	2477	0000	2032		
1.	016	2551	3215	3212	цикл	
2.	000	0000	0000	0000		
3.	000	2016	0000	2022	переход в точку пересечения осей	
4.	000	2017	0000	2023		
5.	016	2566	3224	3212		
6.	000	2017	0000	2040		
7.	000	0000	0000	0000		
2570	003	2040	2015	2040	конец оси?	
1.	036	0000	2602	2032		
2.	002	0000	2015	2033		
3.	016	2574	3215	3212	шаг по оси	
4.	002	0000	2477	2032		
5.	000	0000	0000	2033		
6.	016	2577	3215	3212	метка	
7.	000	2477	0000	2032		
						Ось "-Y"

2600	016	2570	3215	3212	цикл	
1.	000	0000	0000	0000		
2.	050	0500	0000	2043	печать параметров графика на вычерч. заданной кривой	
3.	070	2010	0000	0000		
4.	056	0000	2644	0000		
5.	077	0000	0000	0000		
6.	000	0000	0000	0000		
7.	000	0000	0000	0000		
2610	000	0000	0000	0000		
1.	000	0000	0000	0000		
2.	056	0000	2644	0000		
3.	000	0000	0000	0000		
4.	052	0001	0000	2626	настройка на X Масштабирование определение большего вылета (от точки пересечения) L (для подбора M) полярность вылета? "-" ($x_0 \rightarrow l$) "+"	
5.	603	2026	2024	0000		
6.	436	2024	2620	0003		
7.	400	2026	0000	0003		
2620	001	0003	0000	2305		
1.	436	2016	2624	2306		
2.	003	0003	0000	2305		
3.	602	2010	2016	2306		
4.	200	2306	2014	0000	$x_\varphi - x_0 \rightarrow l$	
5.	016	2626	2312	2353	подбор M (масштабн. коэфф-ента)	
6.	052	0000	0001	0000	настройка на Y	
7.	000	2304	0000	2042	$M \rightarrow M_x$	
2630	016	2631	2615	2626	подбор M	
1.	000	2304	0000	2043	$M \rightarrow M_y$	
2.	000	0000	0000	0000		
3.	056	0000	2410	0000		
4.	000	0000	0000	0000		
5.	000	0000	0000	0000		
6.	000	0000	0000	0000		
7.	000	0000	0000	0000		
2640	000	0000	0000	0000		
1.	000	0000	0000	0000		
2.	000	0000	0000	0000		
3.	000	0000	0000	0000		
4.	052	0000	0000	2110	0 → PA очередн. начало засылка возврата Контроль и за X, Y вычерчивание иммитация ожидан. кривой	
5.	016	2705	2646	2115		
6.	016	2647	2110	2122		
7.	000	7747	0000	2656		
2650	002	2022	2020	2022		
1.	005	2022	2042	0001		$x_k = (X_k - X_{нк}) \cdot M_x + x_0$
2.	001	0001	2016	2022		$x_k < 0 ?$
3.	076	0000	2660	0000		
4.	000	7747	0000	2656	ждать (не чертить)	
5.	016	2650	2110	2122	за следующим X, Y	
6.	000	0000	0000	0000		
7.	000	0000	0000	0000		

2660 002 2010 2022 0000
 1. 036 0000 2654 0000
 2. 002 2023 2021 2023
 3. 005 2023 2043 0002
 4. 001 2017 0002 2023
 5. 036 0000 2654 0000
 6. 000 0000 0000 0000
 7. 002 2011 2023 0000
 2670 036 0000 2654 0000
 1. 000 0000 0000 0000
 2. 015 0000 2656 0000
 3. 036 0000 2702 0000

$x_k > x_{\phi} ?$

$$y_k = (Y_k - Y_{HK}) \cdot M_y + y_0$$

$y_k < 0 ?$

$y_k > y_{\phi} ?$

было ожидание?

4. 016 2675 3173 3177
 5. 016 2676 3224 3212
 6. 016 2677 3170 3177
 7. 000 0000 0000 2656
 2700 016 2650 2110 2122
 1. 000 0000 0000 0000
 2. 056 0000 3310 0000
 3. 000 0000 0000 0000
 4. 000 0000 0000 0000
 5. 000 0000 0000 0000
 6. 000 0000 0000 0000
 7. 056 0000 2116 0000

подъем

ход до точки (x_k, y_k)

спуск

убрать признак ожидания

за следующим x, y

блок окончания x, y
 (резерв)

на конец СП

2710 003 0000 0000 0000
 1. 003 0000 0000 0000
 2. 003 0000 0000 0000
 3. 003 0000 0000 0000
 4. 055 7604 7740 3306
 5. 015 3306 2066 0000
 6. 076 0000 2067 2762
 7. 016 2720 2075 2106
 2720 016 2721 2724 2115
 1. 000 0000 0000 0000
 2. 000 0000 0000 0000
 3. 056 0000 2116 0000

задан текст?

ТЕКСТ

выборка $\Delta x, \Delta y$ формирование
 на обработку текста

возврат после выполнения текста
 (резерв)

на конец СП

4. 052 0000 0000 2110
 5. 005 2014 2014 0002
 6. 005 2015 2015 0003
 7. 001 0002 0003 0001
 2730 044 0001 0001 0002
 1. 004 2014 0002 3102
 2. 004 2015 0002 3103
 3. 000 0000 0000 0000
 4. 000 0000 0000 0000
 5. 056 0000 2754 0000
 6. 000 0000 0000 0000
 7. 000 0000 0000 0000

$$\Delta x^2$$

$$\Delta y^2$$

$$\sin \alpha = \frac{\Delta y}{\sqrt{\Delta x^2 + \Delta y^2}}$$

$$\cos \alpha = \frac{\Delta x}{\sqrt{\Delta x^2 + \Delta y^2}}$$

за первым словом

2740	000	0000	0020	0000		
1.	000	0000	0040	0000		
2.	000	0000	0050	0000		
3.	000	0000	0060	0000		
4.	054	0042	3051	0001	} новое слово выбор заданного \mathbb{R} шрифта	
5.	072	0000	0001	0000		
6.	400	3064	0000	3050		
7.	000	3050	0000	3001		
2750	032	0000	2752	0000		
1.	052	0000	0000	0000		очередной символ
2.	652	0001	0000	2751		
3.	112	0007	2757	0000	конец слова?	
4.	052	0001	0000	2751	пересчет	
5.	016	2756	2110	2122	за следующим словом	
6.	056	2022	2744	3051	на анализ слова	
7.	055	3051	7740	3052	обработка символа	
2760	054	0106	3051	3051		
1.	067	3052	0000	3053		
2.	000	0000	0000	0000		
3.	015	3052	7740	0000	признак служебного?	
4.	076	0000	3000	0000		
5.	016	2766	2751	2762	выборка служебного символа	
6.	033	3053	2740	0000	анализ служебного	
7.	076	0000	2772	0000	$S < 20?$	
2770	072	0000	3053	0000		
1.	456	3111	2777	3110	} выбор масштаба (по таблице)	
2.	033	3053	2741	0000		
3.	076	0000	2777	0000		
4.	072	0000	3053	0000		
5.	000	0000	0000	0000		
6.	000	0000	0000	0000		
7.	056	0000	2751	2762		за очередным СИМВОЛОМ
3000	072	0000	3053	3004	} <table border="1" data-bbox="702 945 882 994"><tr><td>Расшифровка символа</td></tr></table>	Расшифровка символа
Расшифровка символа						
1.	652	0000	0000	0000	выбор строки элементов	
2.	400	0000	0000	3070	счетчик элементов	
3.	000	3072	0000	3073		
4.	000	0000	0000	0000		
5.	003	3073	7761	3073	последний элемент	
6.	036	0000	3130	0000		
7.	055	3070	3074	3071	„00“?	
3010	076	0000	3013	0000		
1.	000	0000	0000	0000		
2.	056	0000	3046	0000		
3.	015	3071	3075	0000	"40"?	
4.	076	0000	3017	0000		
5.	016	3046	3173	3177	подъем пера	
6.	000	0000	0000	0000		
7.	015	3071	3076	0000	"04"?	

3020	076	0000	3022	0000
1.	016	3046	3170	3177
2.	000	0000	0000	0000
3.	000	0000	0000	0000
4.	052	0000	0000	0000
5.	155	3071	7714	0000
6.	136	3106	3030	3100
7.	100	3107	0000	3100
3030	055	3071	7713	0001
1.	054	0070	0001	0001
2.	513	3100	0001	3100
3.	054	0103	3071	3071

спуск пера

Формирование $\Delta x, \Delta y$

4.	112	0001	3025	0001
5.	005	3100	3103	0001
6.	005	3101	3102	0002
7.	002	0001	0002	3104
3040	005	3100	3102	0001
1.	005	3101	3103	0002
2.	001	0001	0002	3105
3.	005	3104	3110	2032
4.	005	3105	3110	2033
5.	016	3046	3215	3212
6.	054	0106	3070	3070
7.	056	0000	3004	0000

$\Delta x = \Delta x \cdot \cos \alpha - \Delta y \cdot \sin \alpha$

$\Delta y = \Delta x \cdot \sin \alpha + \Delta y \cdot \cos \alpha$

$\Delta x = M_{\text{дуга}} \cdot \Delta x$

$\Delta y = M_{\text{дуга}} \cdot \Delta y$

исполнение элемента (УГ)

подготовка следующего

на анализ следующего элемента

3050	052	0000	0000	0000
1.	052	0000	0000	0000
2.	077	0000	0000	0000
3.	000	0000	0000	0000
4.	652	0000	3350	0000
5.	652	0000	3450	0000
6.	652	0000	3550	0000
7.	652	0000	3650	0000
3060	652	0000	3350	0000
1.	652	0000	0000	0000
2.	652	0000	0000	0000
3.	652	0000	0000	0000

№ шрифта (рабочего)

очередное слово

очередной символ

начало шрифта № 0

"-" - приложения № 1

"-" - приложения № 2

4.	652	0000	3350	0000
5.	652	0000	0000	0000
6.	652	0000	0000	0000
7.	652	0000	0000	0000
3070	007	0000	0000	0000
1.	770	0000	0000	0000
2.	103	6000	0000	0000
3.	000	0000	0000	0000
4.	770	0000	0000	0000
5.	400	0000	0000	0000
6.	040	0000	0000	0000
7.	440	0000	0000	0000

начало шрифта № 0

резерв для других шрифтов

строка элементов

счетчик элементов

"подъем"

"спуск"

3100	001	0000	0000	0000	Δx	
1.	001	0000	0000	0000	Δy	
2.	001	0000	0000	0000	$\sin \alpha$	
3.	001	0000	0000	0000	$\cos \alpha$	
4.	001	0000	0000	0000		
5.	001	0000	0000	0000		
6.	102	0000	0000	0000		
7.	302	0000	0000	0000		
3110	101	4000	0000	0000	M букв	(масштабный коэффициент)
1.	101	4000	0000	0000	0	
2.	102	4000	0000	0000	1	
3.	100	4000	0000	0000	2	
4.	101	6000	0000	0000	3	Таблица М
5.	102	6000	0000	0000	4	
6.	100	6000	0000	0000	5	
7.	077	4000	0000	0000	6	
3120	076	5000	0000	0000	7	
1.	103	4000	0000	0000	10	
2.	104	4000	0000	0000	11	
3.	105	4000	0000	0000	12	
4.	075	4000	0000	0000	13	
5.	074	5000	0000	0000	14	
6.	103	6000	0000	0000	15	
7.	102	5000	0000	0000	16	
3130	054	0106	3070	0001	анализ последнего элемента	
1.	076	0000	3134	3004	есть продолжение?	
2.	016	2751	3007	3004	обработка последнего элемента	
3.	000	0000	0000	0000		
4.	054	0042	0001	0002		
5.	072	0000	0002	0000	выделение Р приложения	
6.	400	3054	0000	3141		
7.	054	0045	3070	0001	выделение строки	
3140	072	0000	0001	0000		
1.	652	0000	0000	0000		
2.	456	0000	3003	3070	на новую строку	
3.	000	0000	0000	0000		
4.	000	4000	0000	0000		
5.	000	2000	0000	0000		
6.	000	1000	0000	0000		
7.	000	0400	0000	0000		
3150	000	0200	0000	0000		
1.	000	0100	0000	0000		
2.	000	0040	0000	0000		
3.	000	0020	0000	0000		
4.	000	0010	0000	0000		
5.	000	0004	0000	0000		
6.	000	0002	0000	0000		
7.	000	0001	0000	0000		

3160	000	0010	0000	0000	+x
1.	000	0004	0000	0000	-x
2.	000	0000	0000	0001	+y
3.	000	0000	0000	0002	-y
4.	000	0020	0000	0000	подъем
5.	000	0040	0000	0000	спуск
6.	000	0100	0000	0000	
7.	000	0200	0000	0000	
3170	067	3165	2000	0000	} спуск пера
1.	000	3201	0000	0001	
2.	056	0000	3175	0000	
3.	067	3164	2000	0000	} подъем пера
4.	000	3201	0000	0001	
5.	003	0001	7761	0001	
6.	076	0000	3175	0000	
7.	000	0000	0000	0000	(возврат)
3200	104	5000	0000	0000	C1
1.	105	5000	0000	0000	C2
2.	110	4000	0000	0000	C3
3.	100	4000	0000	0000	C4
4.	003	0000	0000	0000	
5.	067	0003	2000	0000	исполнение шага
6.	000	0000	0000	0000	
7.	056	0000	3175	0000	расчет следующего
3210	000	0000	0010	0000	
1.	000	0000	0004	0000	
2.	000	0000	0002	0000	(возврат)
3.	005	2032	2042	2032	вход с $\Delta x, \Delta y$
4.	005	2033	2043	2033	
5.	001	2032	2044	2022	
6.	001	2033	2045	2023	
7.	056	0000	3224	0000	
3220	000	0000	0001	0000	
1.	000	0000	0002	0000	
2.	005	2022	2042	2022	вход с x_k, y_k
3.	005	2023	2043	2023	
4.	004	2022	2046	0002	
5.	041	0002	7752	0002	
6.	002	0002	7752	0002	\tilde{x}_k (в шагах)
7.	004	2023	2046	0003	
3230	041	0003	7752	0003	
1.	002	0003	7752	0003	\tilde{y}_k (в шагах)
2.	002	0002	2012	2047	Δx (ведущ.)
3.	036	3161	3235	2051	
4.	000	3160	0000	2051	
5.	000	0002	0000	2012	
6.	000	2022	0000	2044	
7.	000	2023	0000	2045	

3240	002	0003	2013	2050	Δy (ведом.)
1.	036	3163	3244	0001	
2.	000	0000	0040	0000	
3.	000	3162	0000	0001	
4.	075	2051	0001	2051	
5.	000	0003	0000	2013	
6.	003	2050	2047	0000	Δ ведущ. > Δ ведом.?
7.	036	2047	3253	0001	
3250	000	2050	0000	2047	реверс Δ ведущ. и сигналов
1.	000	0001	0000	2050	
2.	067	2051	0000	2051	
3.	003	0000	2047	0000	
4.	076	3201	3212	2054	
5.	004	2050	2047	2052	$\beta = \frac{\Delta_{ведом.}}{\Delta_{ведущ.}}$
6.	000	0000	0000	0002	
7.	003	2047	0000	2053	$\Delta_{ведущ.} \rightarrow C \Delta$
3260	003	2052	0000	2052	следующий шаг
1.	003	2053	7761	2053	конец ?
2.	036	2051	3212	0003	(засылка сигнала ведущего)
3.	001	0002	2052	0002	$R = R + \beta$
4.	002	0002	7764	0000	$R < 0,5 ?$
5.	036	2054	3271	0001	
6.	002	0002	7761	0002	$R = R - 1$
7.	067	2051	0000	0003	дополнение ведущим сигналом
3270	075	0003	2051	0003	
1.	003	2053	7760	0000	блок временной задержки
2.	076	3201	3277	0001	
3.	003	2054	3200	0000	
4.	036	3200	3277	0001	
5.	003	2054	3203	2054	
6.	003	2054	0000	0001	
7.	000	0000	0000	0000	
3300	016	3261	3204	3177	на исполнение шага
1.	000	0000	0000	0000	
2.	000	0000	0000	0000	
3.	000	0000	0000	0000	
4.	000	0000	0017	0000	
5.	740	0000	0000	0000	
6.	070	0000	0000	0000	код символа
7.	070	0000	0000	0000	строка символа
3310	055	7714	3306	0001	рисовать?
1.	076	0000	3314	0000	Символ графика
2.	016	3314	3170	3177	
3.	016	3314	3173	3177	
4.	016	3315	3224	3212	
5.	054	0050	3306	0001	
6.	055	0001	3304	0002	
7.	072	0000	0002	0000	

3320	436	3530	3344	3307
1.	016	3322	3170	3177
2.	055	3307	7714	0000
3.	036	3346	3325	2022
4.	000	3347	0000	2022
5.	055	3307	7712	0000
6.	036	0000	3331	0000
7.	013	2022	3345	2022
3330	005	2022	3110	2022
1.	054	0102	3307	3307
2.	055	3307	7714	0000
3.	036	3346	3335	2023

элемент
знак Δx

Δx

4.	000	3347	0000	2023
5.	055	3307	7712	0000
6.	036	0000	3341	0000
7.	013	2023	3345	2023
3340	005	2023	3110	2023
1.	016	3342	3215	3212
2.	054	0102	3307	3307
3.	076	0000	3322	0000
4.	056	0000	2110	0000
5.	000	4000	0000	0000
6.	102	0000	0000	0000
7.	302	0000	0000	0000

Δy

исполнение элемента
конец символа?

за следующей точкой

3350	010	4020	0000	0001
1.	120	2041	2000	0021
2.	030	2041	1201	5031
3.	030	2041	1201	5051
4.	131	3045	6563	0061
5.	232	3046	0600	6071
6.	010	4020	2112	0101
7.	030	2040	1202	0121
3360	010	4011	1206	0131
1.	320	4115	5605	1001
2.	210	4020	2066	0141
3.	030	4202	0402	7000

0
1
2
3
4
5
6
7
8
9
+
-

Строки символов
шрифт № 0

4.	042	3230	0000	0121
5.	200	4000	0000	0151
6.	040	1100	5504	0300
7.	403	0300	0000	0000
3370	100	4030	7401	0161
1.	131	3041	6601	2021
2.	045	2021	2401	7170
3.	041	2025	2401	7270
4.	010	4222	2666	2171
5.	222	2046	0604	0201
6.	131	3045	0051	0211
7.	045	0030	3105	0021

/
:
пробел
10
↑
{
x
=
:
[

3400	041	0030	3501	0021	Ј
1.	010	4222	2660	2221	*
2.	130	3041	2560	0241	,
3.	130	4120	1500	0241	≠
4.	010	4222	2551	0251	<
5.	130	4337	7374	0200	>
6.	043	3733	7401	7000	:
7.	200	4100	1500	5261	:
3410	041	3367	2570	0271	А
1.	043	0110	2517	0301	Б
2.	043	0110	1517	0321	В
3.	232	3045	0000	0311	Г
4.	040	5013	0300	5331	Д
5.	042	0206	0600	3341	Е
6.	042	3632	7100	3351	Ж
7.	030	2041	1201	5051	З
3420	040	3034	0110	0371	И
1.	131	3041	0614	0371	И
2.	040	3030	7051	1401	К
3.	041	3132	0000	0021	Л
4.	040	3033	6320	0021	М
5.	040	3030	7301	0411	Н
6.	010	4020	0000	0001	О
7.	040	3033	0100	0021	П
3430	040	2301	1025	1311	Р
1.	232	2045	1605	5421	С
2.	030	3042	0200	0431	Т
3.	043	0110	2030	7441	У
4.	330	4036	0550	6451	Ф
5.	042	3632	7000	0361	Х
6.	040	3030	7071	0461	Ц
7.	030	0140	3030	7441	Ч
3440	040	3030	7073	0501	Ш
1.	040	3030	7073	0521	Ш
2.	332	3040	7074	0531	Ъ
3.	302	0000	0000	0531	Ъ
4.	130	4300	2516	0551	Э
5.	040	3030	7100	6561	Я
6.	042	2200	2027	0571	Я
7.	077	0000	0000	0000	
3450	000	2112	0150	7011	0
1.	055	5605	1403	5300	1
2.	070	7402	0000	0000	2
3.	056	6650	5000	0041	3
4.	202	0402	0000	0000	4
5.	055	5150	0000	0011	5
6.	105	0020	2000	0021	6
7.	301	5050	0000	0011	7

Приложение № 1

строки продолжений

3460	155	1605	5060	6111	10
1.	021	1201	5050	0011	11
2.	676	7403	0300	0000	12
3.	510	1112	0150	0051	13
4.	202	0402	6000	0000	14
5.	015	0051	0564	0320	15
6.	040	3200	7604	0300	16
7.	262	6402	5000	0000	17
3470	060	4202	0402	6000	20
1.	014	0070	6040	5151	21
2.	060	6026	0202	0231	22
3.	606	2262	6403	5000	23
4.	051	0015	0402	7170	24
5.	606	0400	0000	0201	25
6.	400	3030	4000	0241	26
7.	131	3200	7074	0200	27
3500	022	0205	0000	0311	30
1.	506	0070	7403	0300	31
2.	301	1015	1000	0311	32
3.	017	0600	3133	0021	33
4.	307	0032	0205	0311	34
5.	070	7031	0000	0361	35
6.	236	7274	0200	0000	36
7.	550	4070	7232	3021	37
3510	337	7374	0200	0000	40
1.	030	7070	0402	0000	41
2.	060	6152	0114	0250	42
3.	006	0070	7402	0200	43
4.	057	0510	3403	7370	44
5.	152	0201	1025	1431	45
6.	300	3030	0000	0471	46
7.	070	7005	5403	1000	47
3520	030	3070	7000	0511	50
1.	300	3030	7402	7000	51
2.	030	3070	7200	0461	52
3.	736	3040	7073	0541	53
4.	110	2517	0403	6360	54
5.	551	1201	5070	0011	55
6.	020	2112	0150	7011	56
7.	550	6153	0064	0200	57
3530	650	4740	0000	0000	▽
1.	047	4353	4000	0000	∇
2.	650	4746	1000	0000	∇
3.	050	3600	0000	0000	□
4.	247	4051	7000	0000	■
5.	232	4172	3000	0000	■
6.	076	7272	3000	0000	◇
7.	047	4211	4142	0000	+

КА	000	0001	0000	0000	
1.	050	0413	0000	7767	} ТЕСТОВЫЙ ПРИМЕР вззов ИС ввод программн "ГРАФИК"
2.	070	7500	0001	0000	
3.	010	2000	0004	0000	
4.	050	0500	0000	0600	
5.	070	0000	0000	0000	
6.	056	0000	0031	0000	
7.	000	0000	0000	0000	
0010	000	7777	7777	0000	} печать данной программн на тест № 1
1.	052	0000	0000	0000	
2.	652	0001	0000	0011	
3.	100	0001	0000	4000	накопление x, y $x \rightarrow$
4.	100	0002	0000	5000	} $y \rightarrow$ конец цикла?
5.	012	0777	0030	0000	
6.	000	0000	0000	0000	
7.	000	0000	0000	0000	
0020	000	0000	0000	0000	
1.	000	0000	0000	0000	
2.	000	0000	0000	0000	
3.	000	0000	0000	0000	
4.	016	0025	2002	7610	} обращение к СП "ГРАФИК"
5.	000	4000	0000	5000	
6.	000	5774	0000	0777	
7.	052	0000	0000	0011	
0030	000	0000	0000	0000	} (возврат) блок управления тестом с ДЗУ
1.	020	0004	0000	0001	
2.	055	0001	0037	0000	
3.	036	0000	0061	0000	
4.	020	0001	0000	0041	
5.	020	0002	0000	0042	
6.	020	0003	0000	0043	
7.	077	0000	0000	0000	
0040	056	0000	0061	0000	} а } параметрн тестового в } примера (функции) с }
1.	105	5000	0000	0000	
2.	105	5000	0000	0000	
3.	105	5000	0000	0000	
4.	075	4000	0000	0000	} δ - шаг функции
5.	000	0000	0000	0000	
6.	000	0000	0000	0000	
7.	000	0000	0000	0000	
0050	100	4000	0000	0000	
1.	101	4000	0000	0000	
2.	104	5000	0000	0000	
3.	301	4000	0000	0000	
4.	304	5000	0000	0000	
5.	102	6000	0000	0000	
6.	111	4000	0000	0000	
7.	105	5000	0000	0000	

0060 103 7000 0000 0000
 1. 000 0054 0000 0045
 2. 001 0045 0044 0045
 3. 016 0064 7501 7610 }
 4. 075 0045 0005 0046 }
 5. 000 0045 0000 0001
 6. 005 0046 0042 0002
 7. 016 0070 0011 0030
 0070 000 0000 0000 0000
 1. 056 0000 0062 0000
 2. 000 0000 0000 0000
 3. 000 0000 0000 0000

$x + \delta \rightarrow x$
 к СП ($y = \sin x$)

$\sin x$

x
 $y = \sigma \cdot \sin x$
 отправить X, Y в рабочее поле

цикл расчета точек
 (выход из цикла по заполн. РП)

4. 056 0044 0076 0043
 5. 020 0003 0000 0047
 6. 001 0047 0043 0047
 7. 000 0000 0000 0000
 0100 000 0000 0000 0000
 1. 016 0102 7501 7610 }
 2. 075 0047 0005 0073 }
 3. 002 0051 0073 0001
 4. 005 0041 0001 0001
 5. 005 0042 0073 0002
 6. 016 0107 0011 0030
 7. 003 0047 0060 0000

ДЗУ3 $\rightarrow \theta$
 $\theta = \theta + \delta$ следующий шаг

Эллипс

$\sin \theta$
 $1 - \sin \theta$
 $x = a \cdot (1 - \sin \theta)$
 $y = \sigma \cdot \sin \theta$
 отправить X, Y в РП

0110 036 0000 0076 0000
 1. 056 0000 0031 0047
 2. 012 0777 0030 0000
 3. 034 0064 0011 0003
 4. 020 0004 0000 0004
 5. 055 0004 0010 0004
 6. 036 0000 0023 0000
 7. 000 0004 0000 0026
 0120 013 0026 0003 0026
 1. 016 0025 2002 7610
 2. 000 4000 0000 5000
 3. 000 5754 0000 0777

в блок управления ДЗУ

4. 052 0000 0000 0011
 5. 000 0000 0000 0000
 6. 012 0777 0030 0000
 7. 000 0000 0000 0000
 0130 000 0000 0000 0000
 1. 000 0000 0000 0000
 2. 000 0000 0000 0000
 3. 000 0000 0000 0000
 4. 000 0000 0000 0000
 5. 016 0025 2002 7610
 6. 000 4000 0000 5000
 7. 000 5754 0000 0777

0140	052	0000	0000	0011
1.	000	0000	0000	0000
2.	012	0777	0030	0000
3.	000	0000	0000	0000
4.	000	0000	0000	0000
5.	000	0000	0000	0000
6.	000	0000	0000	0000
7.	000	0000	0000	0000
0150	000	0000	0000	0000
1.	016	0025	2002	7610
2.	000	4000	0000	5000
3.	000	6761	0000	0777

4.	052	0000	0000	0011
5.	000	0000	0000	0000
6.	000	0000	0000	0000
7.	000	0000	0000	0000
0160	000	0000	0000	0000
1.	000	0000	0000	0000
2.	000	0000	0000	0000
3.	000	0000	0000	0000
4.	000	0000	0000	0000
5.	000	0000	0000	0000
6.	000	0000	0000	0000
7.	000	0000	0000	0000

§2. АВТОМАТИЧЕСКАЯ РИСОВКА

СИМВОЛОВ

ДЛЯ ГЕОЛОГИЧЕСКИХ ПОСТРОЕНИЙ

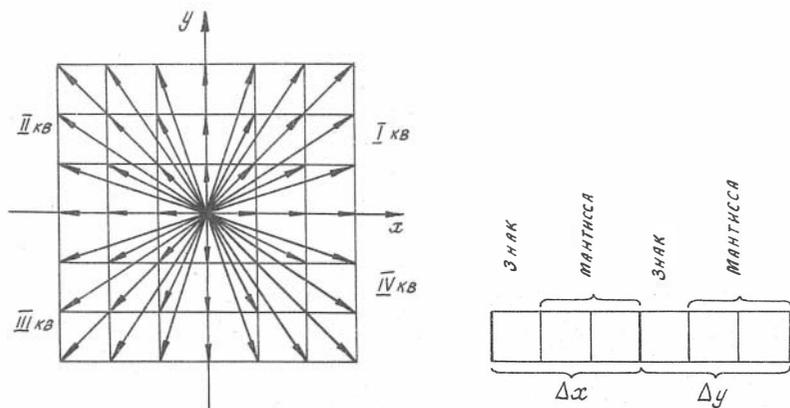
Графическое изображение цифр, букв, различных условных знаков, надписей, текста возможно с применением ЦВМ и графопостроителей [9]. Ниже приводится метод преобразования любого текста, символов в последовательности перемещений карандаша графопостроителя.

1. П р и н я т н ы е о п р е д е л е н и я

Условная единица — наименьший отрезок на оси координат, принятый за единицу, конкретная величина которого выбирается исходя из желаемой подробности отражения строения символа (плавность, конфигурация и т.д.).

Элемент — отрезок прямой, определенный составляющими по осям x и y соответственно $\pm\Delta x, \pm\Delta y$, выраженный в условных единицах. Под элементом понимается перемещение карандаша из точки стояния в точку с координатами $\Delta x, \Delta y$ в относительной системе координат, центр которой находится в точке стояния карандаша. Направление движения карандаша определяется знаками соответствующих составляющих. Один из вариантов: на каждый элемент отводится 6 разрядов, по 3 разряда на Δx и Δy соответственно. При этом один разряд — знак и два разряда — значение составляющей элемента в условных единицах (мантисса). Принятое определение элемента позволяет в данном варианте из любой точки стояния карандаша задать перемещение в любую из 48 окружающих (4 квадрата, в каждом квадрате по 12 комбинаций из условных

единиц), что дает большую гибкость в изображении символов.



Виды элементов при
3х разрядах на составляющую

Структура элемента
из 6 разрядов

Рис. 42

С и м в о л — буква любого алфавита, цифра, условный знак и любое другое графическое изображение, часто встречающееся при выводе графических данных, введенные в систему так, чтобы можно было удобно ими пользоваться. Символ конструируется из целого числа элементов, количество которых может быть в широких пределах от нуля до нескольких сотен, что позволяет иметь символы довольно большой сложности.

Ш р и ф т — упорядоченная система символов, где каждому символу соответствует определенное место. В шрифте объединяются однотипные символы для удобства их применения (русские буквы, цифры, математические знаки, греческий алфавит, топографические условные знаки и так далее). Размер шрифта ограничен выбранной системой кодировки шифра символов. (В данном варианте выбрана система кодировки по 6 двоичных разрядов на каждый символ, поз-

тому в одном шрифте может быть $2^6 = 64$ символа). Количество шрифтов определено принятой системой кодировки "слов" текста (в нашем варианте может быть 1, 2, ..., 8 шрифтов).

С л о в о - ячейка памяти ЦВМ, в которой закодированы символы текста по принятой в данном варианте системе. В слово входит номер шрифта и целое одинаковое число символов (равное 7 для принятой системы кодировки символов). Порядок символов определяется текстом.

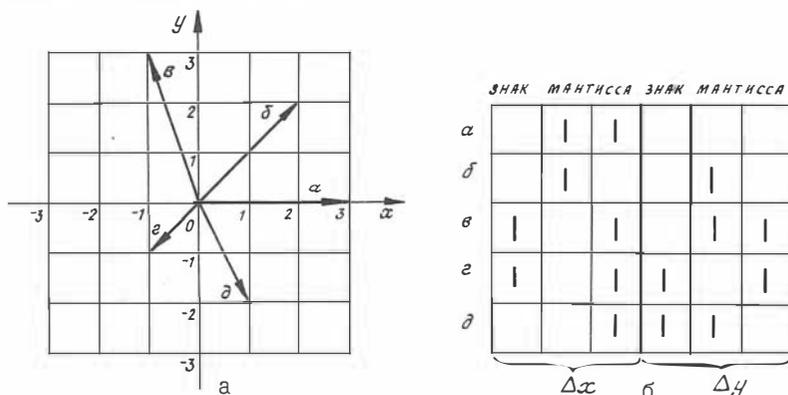
Т е к с т - любая последовательность символов, которые нужно графически отобразить. Текст состоит из слов. Размер текста задается, исходя из наличия памяти (указывается начало и конец текста), то-есть текст **может** состоять из одного или многих слов. Начало текста указывается в обращении координатами x, y . Рассчитывается размещение графического изображения символов на рабочем поле. Компоновка текста: количество символов в строке, масштаб символов, угол наклона, переход на следующую строку, расстояние между строками и т.д. заложена в служебных символах.

2. К о д и р о в к а

К о д и р о в к а составляющих элементов. При выборе способа кодирования элемента преследовалась цель создать элемент с достаточно большой гибкостью в смысле угла наклона и длины элемента, чтобы была возможность строить разнообразно по конфигурации символы. С другой стороны, преследовалась достаточно плотная упаковка элементов в ячейке машины. Если взять один разряд на мантиссу составляющей, то получится всего 8 разных элементов (8 разных направлений), но единственная длина. На элемент в целом уйдет 4 разряда. В ячейке разместится 11 элементов. Если взять два разряда, то получим 48 направлений по 3 разным длинам составляющих при упаковке по 7 элементов в ячейке. Если взять три разряда, то могут быть 196 направлений отрезков, 7 разных длин составляющих, но в ячейке будет только 5 элементов. Этот вариант тоже может быть приемлем. Если увеличить дальше разрядность мантиссы, то при возрастании гибкости элемента снижается возможность упаковки многих элементов в ячейки. Общее количество

ство ячеек на символ повышается, так как символ обычно состоит из многих элементов. Возрастающая гибкость элементов не найдет применения. Число недействующих разрядов также возрастает. Например, при мантиссе 3 или 4 теряется 2 разряда, при мантиссе 5 - 6 разрядов.

Кодировка при мантиссе составляющей 3 и 4 вполне приемлема для построения сложных символов, и тому же имеем достаточно плотную упаковку элементов в ячейке. Кодировка при мантиссе составляющей в 2 единицы была проведена на алфавите, цифрах, знаках.



Пример задания 5-ти элементов из одной точки в принятой кодировке

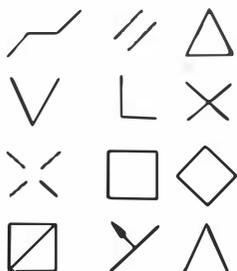
Кодировка элементов „a“, „f“, „e“, „d“ в двоичной системе

Рис. 43

В кодировке элементов заложена возможность выполнения подъема и спуска карандаша при вычерчивании символа. Для этого использованы комбинации типа - 0,0; 0,-0. Элемент 00 не выполняет никакого действия.

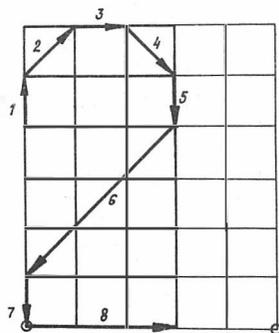
К о д и р о в к а с и м в о л а. Предусмотрена возможность конструирования символов из большого числа элементов. Каждый символ расшифровывается как порядковый номер той ячейки, где он закодирован. Если число элементов символа равно или меньше того, что умещается в ячейке, ос-

тающиеся разряды будут оставлены нулевыми и символ закончится. Если же символ не уместится в одной ячейке, он мо-



а

Примеры символов



б

Пример символа
из элементов

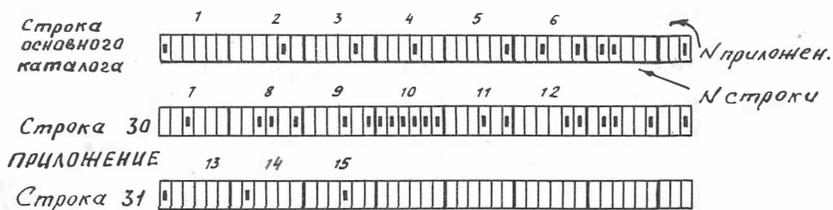
Рис. 44

Таблица значений составляющих элементов символа "2"

N	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Δx	0	0	0	0	+1	+1	+1	0	-3	0	0	0	+2	0	0
Δy	+2	+2	+1	+1	+1	0	-1	-1	-3	-1	+3	0	0	0	0

жет быть продолжен. Для этого в последних трех разрядах (не задействованных вообще под элементы, так как они являются остатком от 45 разрядов $45 - 6 \cdot 7 = 3$ []), указывается номер массива, в котором есть продолжение данного символа. В таком случае 6 разрядов последнего элемента расшифровываются как номер этого продолжения. По этим данным программа находит ячейку (адрес начала приложения плюс номер строки продолжения) и расшифровывает закодированные в ней элементы. Если и тут не конец, то таким же образом указывается следующее продолжение и так далее, пока символ не будет весь распечатан. На один символ можно задействовать очень большое чис-

ло продолжений (в данном варианте $7 \times 64 = 448$). Это качество позволит в одном алфавите собирать самые разнообразные по сложности символы и все они будут закодированы экономно.



Кодировка символа "2" (см. рис.44а)
в строке основного каталога и приложения

Рис. 45

Поскольку каждый символ закодирован, то закодирован весь шрифт. Шрифт состоит из основного массива-каталога, где каждому символу соответствует строка, и одного или нескольких (до 7) массивов-приложений, в которых закодированы продолжения больших символов в произвольном порядке (в отличие от основного каталога, в котором строки жестко соответствуют коду символа). Соответственно на весь шрифт занято требуемое количество ячеек.

В кодировке символов, кроме того, заложен служебный символ-ключ (77), который говорит о том, что следующие 6 разрядов, обычно занимаемые символом, есть не символ, а условное число для изменения какого-либо параметра программы. Например, условные числа от 0 до 20 служат для изменения масштаба, условные числа от 21 до 40 служат для изменения угла поворота символов относительно первоначальной оси координат, остальные - еще на какие-либо манипуляции (подъем строки для написания степени, спуск строки, возврат на шаг, возврат в начало строки, переход к следующей строке и т.д. Эта гибкость кодировки позволяет кодировать любой многоэтапный сложный текст с изменением мас-

штаба хоть через каждую букву, с изменением направления надписи и так далее.

К о д и р о в а н и е с л о в а и т е к с т а .
 Выбрана кодировка символов по 6 разрядов на символ из со-
 образений удобства записи в восьмеричной системе. Всего
 символов может быть очень много, а система 6 разрядов до-
 пускает только 64 в десятичной системе (64_{10}) или 77 в
 восьмеричной (77_8). Для расширения набора символов введе-
 но понятие шрифта. Тем более, что в 45 разрядной ячейке
 остаются незадействованны 3 разряда, которые в кодировке
 слова использованы для указания, из какого шрифта символ
 в этом слове. Таким образом, имеется возможность в одном
 тексте использовать символы 8 различных шрифтов, т.е. об-
 щее число символов может быть до $8 \times 64 = 512$ в одной си-
 стеме.

Кодировка слова имеет вид:

0	40	41	42	43	44	45	46
Зр	6р						
№ шрифта	А	Б	В	Г	Д	Е	Ж

Кодировка текста имеет вид:

№ шрифта	Шифры символов						
	6р	6р	6р	6р	6р	6р	6р
0	53	41	40	17	12	11	09
2	01	03	13	17	50	03	05
1	10	12	10	77	22	40	15
0	14	12	9	17	20	22	30
0	13	77	10	15	17	19	21

3. О п и с а н и е р а б о т н ы х о п е р а т о р н ы х с х е м н ы х п р о г р а м м н ы х "С и м в о л"

В начале схемн стоит оператор H_0 . Это пустой оператор, служит заголовком, адресом программн, по которому к ней следует обращаться. Оператор A_1 выбирает из каталога символов строку, в которой закодирован нужный нам символ, т.е. символ, который должен быть изображен в результате работы всей программн "Символ". Строка выбирается по коду самого символа, т.е. если код символа - 15, то берется 15-ая строка, если 10 - 10-ая строка. Номер шрифта учтен в программе "Текст". (Адрес строки = адрес начала шрифта + код символа).

Оператор K_2 устанавливает счетчик элементов. В строке при выбранной системе кодировки может быть максимально 7 элементов, но место последнего, седьмого элемента, может занять указание на то, что имеется продолжение символа. В связи с этим оператор K_3 отсчитывает только 6 элементов, а седьмой элемент проверяется в дальнейшем отдельно. Оператор K_3 вычитает из счетчика элементов единицу, т.е. фиксирует количество оставшихся, нераскодированных элементов в строке после конца работы всего цикла. Оператор P_4 проверяет, не получилось ли в результате действия оператора K_3 число меньше 0. Пока идут элементы с 1-го по 6-ой, в счетчике K_3 будут числа ≥ 0 и после оператора P_4 будет работать оператор A_9 , который выбирает из строки очередной элемент, а идущий за ним оператор A_{10} подготавливает следующий элемент. После работы этих двух операторов элемент, который был записан первым в строке, теперь будет выделен, записан отдельно, а на место его 6-ти разрядов будут подвинуты 6 разрядов следующего за ним элемента. С оператора P_{11} начинается анализ того элемента, который был выделен оператором A_9 . Оператор P_{11} проверяет, не является ли этот элемент нулевым элементом. Если элемент нулевой, то после P_{11} работает оператор K_3 , который вычитает единицу из счетчика элементов и пока не получится $K < 0$, будут снова работать операторы P_4 , A_9 , A_{10} , P_{11} . Если элемент не нулевой, то после оператора P_{11} работает оператор P_{12} , который проверяет, не является ли очередной элемент кодом "40". Если да, то оператор A_{13} формиру-

ет обращение к программе УГ для осуществления подъема пера, а после выполнения его происходит переход к оператору K_3 . Если же кода "40" нет, то оператор A_{14} проверяет, не является ли он кодом "04". Если очередной элемент - код "04", то оператор A_{15} формирует обращение к программе УГ для осуществления спуска пера, после чего снова работает оператор K_3 . В случае, когда очередной элемент ни код "40", ни код "04", значит он - код перемещения - его координат и направления. Оператор A_{16} раскодирует Δx , Δy , т.е. представляет их в виде цифр в двоичной системе со своим знаком. Оператор A_{17} формирует обращение к программе УГ для выполнения заданного перемещения.

Так раскодируются и выполняются все элементы до 7-го. Когда очередным элементом станет 7-ой, то оператор K_3 получит после вычитания единицы из счетчика элементов отрицательное число. Это обнаруживает оператор P_4 и после него в таком случае работает оператор P_5 . Он проверяет, есть ли продолжение данного символа. Продолжение устанавливается по последним 3-м разрядам строки. Если они нулевые, то продолжения нет. В этом случае нужно еще исполнить 7-ой элемент. После оператора P_5 работает оператор A_8 , который заснает после исполнения 7-го элемента уход на конец. Исполнение 7-го элемента осуществляется по тому же пути, как и 1-6 элементов, т.е. после оператора A_8 работает A_9 и дальше до A_{17} . После A_{17} идет оператор P_{18} , который проверяет, засна ли уход на конец. В случае, когда засна уход, после P_{18} идет оператор A_{19} , означающий конец работы.

Если после оператора A_{17} нет ухода на конец, то после него работает оператор K_3 .

Но может быть и так, что 7 элементов строки не дают еще полное изображение символа и есть его продолжение в каком-то приложении. В этом случае 6 разрядов 7-го элемента и 3 старшие разряда строки дают полный адрес продолжения символа. После оператора P_5 , когда он установит, что есть продолжение символа, работает оператор A_6 . Он выделяет приложение, в котором содержится продолжение символа, а оператор A_7 выделяет в этом приложении строку, относящуюся к этому символу. С этими данными снова схема начинает работу

с оператора H_0 .

Конец работы наступит тогда, когда символ полностью выполнен, нет никаких продолжений, и после A_{17} заслан уход на конец.

4. Структура операторной схемы программы "Символ"

- H_0 - оператор начала, заголовок.
- A_1 - выбор строки символа из каталога.
- K_2 - установка счетчика элементов.
- K_3 - вычитание единицы из счетчика элементов.
- P_4 - проверка: $K < 0$?
- P_5 - проверка: есть продолжение?
- A_6 - выделение \mathbb{P} приложения с продолжением.
- A_7 - выделение \mathbb{P} строки с продолжением.
- A_8 - засылка ухода на конец после исполнения перемещения.
- A_9 - выделение очередного элемента.
- A_{10} - подготовка следующего элемента.
- P_{11} - проверка: "00"? (есть ли отсутствие действия).
- P_{12} - проверка: "40"? (есть ли подъем).
- A_{13} - исполнение подъема.
- P_{14} - проверка: "04"? (есть ли спуск).
- A_{15} - исполнение спуска.
- A_{16} - раскодировка $\Delta x, \Delta y$.
- A_{17} - исполнение перемещения.
- P_{18} - проверка: уход на конец?
- A_{19} - конец.

$H_0 A_1 K_2^{11,13,15,18} K_3 P_{419} P_{548} A_6 A_7 A_8^{48} A_9 A_{10} P_{11}^{13}$

$P_{2414} A_{13}^3 P_{1416}^{12} A_{15}^3 A_{16}^{14} A_{17} P_{1843} A_{19}$.

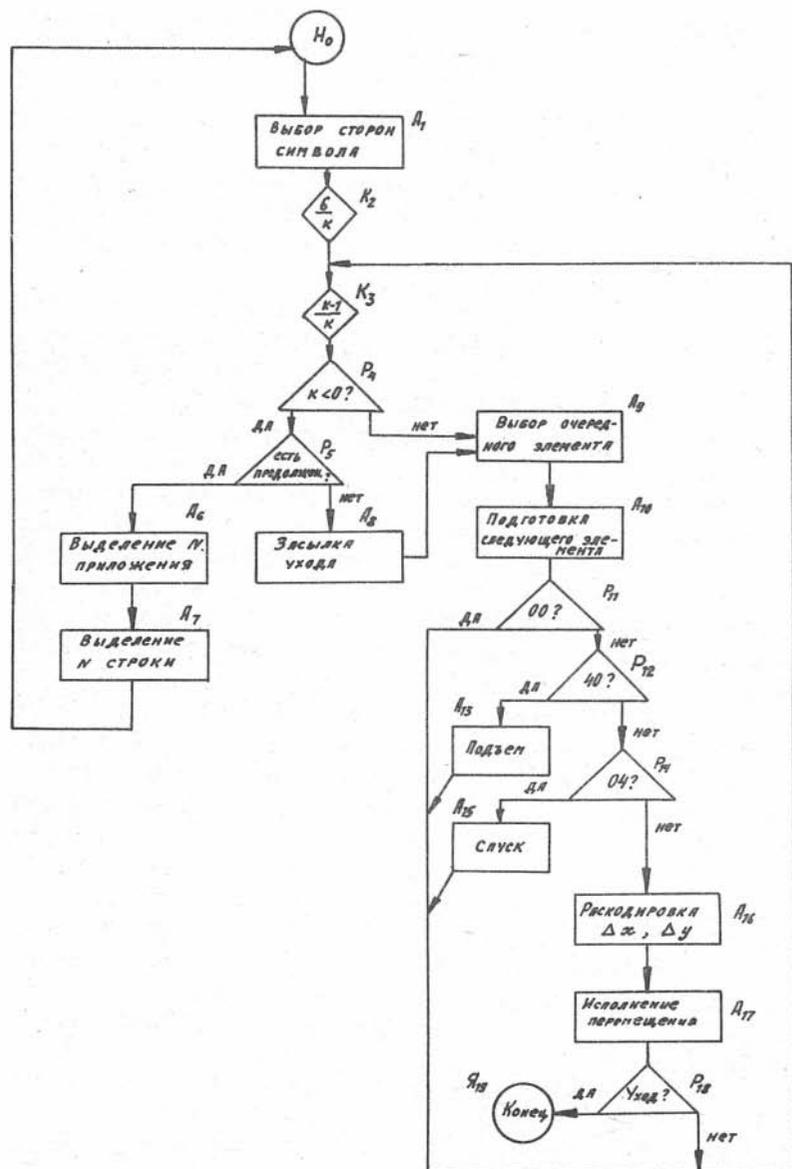


Рис. 46. Блок-схема программы «Символ»

5. О п и с а н и е р а б о т н ы х о п е р а т о р н ы х с х е м н ы х п р о г р а м м н ы х "Т е к с т"

Оператор H_0 , стоящий в начале программы, является оператором начала. Это пустой оператор, своего рода заголовок программы. Оператор K_1 устанавливает в счетчике слов K начальное значение Z_0 . Текст в данном случае состоит из набора ячеек, каждая из которых - слово. "Длина" текста Z задается в обращении к подпрограмме.

Оператор A_2 выбирает Δx по адресу A_2 2-ой строки обращения к программе, адрес Δy находится в следующей за Δx ячейке. По выбранным Δx и Δy оператор A_3 изменяет угол поворота осей через программу УГ.

Далее оператор P_4 проверяет, не кончился ли массив слов. Пока массив не кончился, после оператора P_4 работает оператор A_5 , который выбирает очередное слово из текста, а за ним оператор A_6 выделяет номер шрифта, который должен использоваться для написания данного слова. Номер шрифта указан в 3-х старших разрядах слова. Затем работает оператор K_7 , который устанавливает начальное значение счетчика символов в слове. Максимально в слове при выбранной системе кодировки имеется 7 символов, поэтому начальное $i = 7$. Оператор P_8 проверяет, не кончилось ли слово, т.е. проверяет условие $i = 0$. Если слово не кончилось, то после оператора P_8 работает оператор K_9 , который вычитает единицу из счетчика символов i . Затем оператор A_{10} выделяет очередной символ в слове и одновременно подготавливает следующий. Если очередной символ имеет служебное значение, то идет на оператор P_{15} , если же это обычный символ - то на A_{13} . Это распределение осуществляет оператор P_{11} . Обычно это пустая ячейка и если идет простой символ, то он после оператора A_{10} попадает на оператор P_{12} . Если же символ служебный, то в оператор P_{11} будет послана команда, отсылающая символ на оператор A_{15} .

Пусть очередной символ - обычный. После оператора P_{11} его проверяет оператор P_{12} , а именно он выясняет, не равен ли символ 77. Символ $/S/$ 77 является признаком того, что

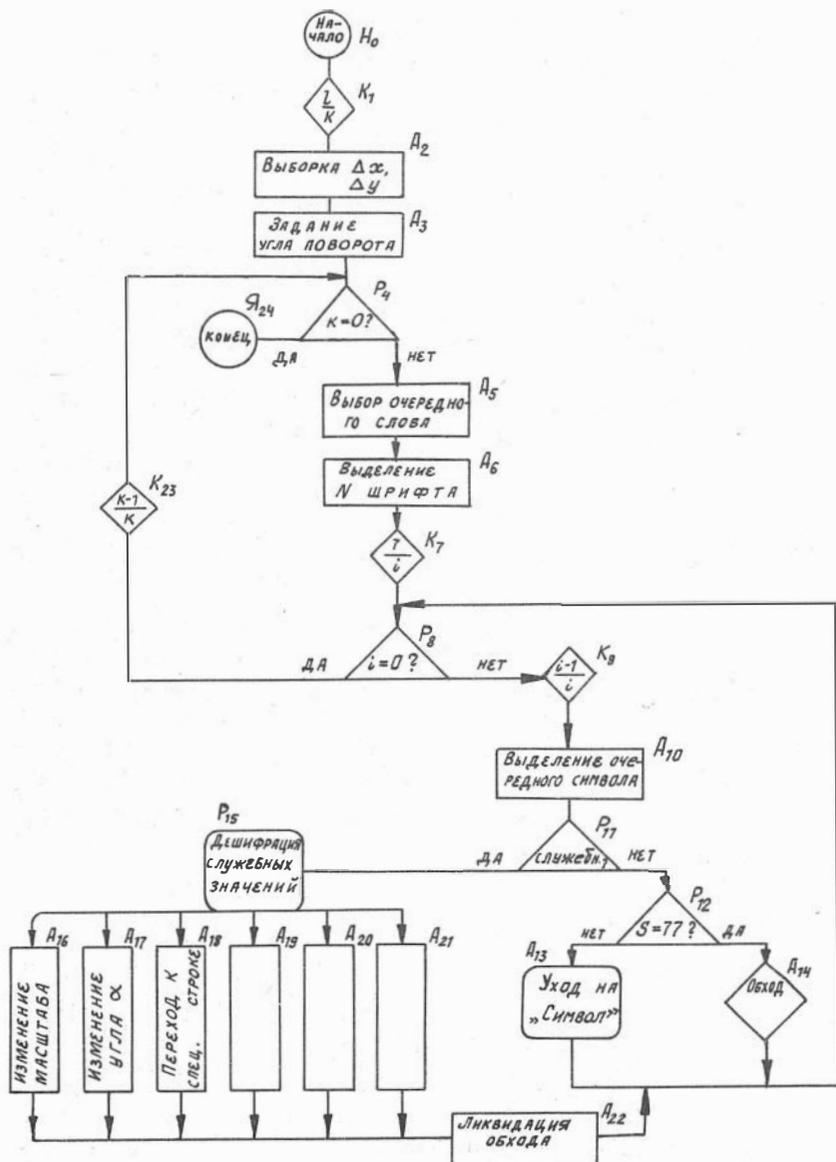


Рис. 47. Блок-схема программы „Текст“

следующий за ним символ - служебный, поэтому, если оператор P_{12} установит, что $S = 77$, то следом будет работать оператор A_{14} , который и заснлает в P_{11} обход для служебных символов. Если же символ не равен 77, то после оператора P_{12} через оператор A_{13} происходит уход в блок рисовки символа по программе "Символ". После операторов A_{13} и A_{14} происходит переход на оператор P_8 , т.е. начинается работа с новым очередным символом того же слова. В случае служебного символа после оператора P_{11} работает оператор A_{15} , задачей которого является дешифрация служебных значений. В зависимости от конкретного содержания служебного символа работает один из операторов A_{16} , A_{17} , A_{18} , A_{19} , A_{20} , A_{21} . Оператор A_{16} изменяет масштаб, оператор A_{17} изменяет угол α , оператор A_{18} осуществляет переход к новой строке. Операторы A_{19} , A_{20} , A_{21} пока зарезервированы с целью расширения возможностей программы. После дешифрации служебного символа нужно убрать команду на обход, занланную в оператор P_{11} , что и делает оператор A_{22} , который работает после любого из операторов A_{16} - A_{21} . От оператора A_{22} происходит переход на оператор P_8 , т.е. начинается работа со следующим символом.

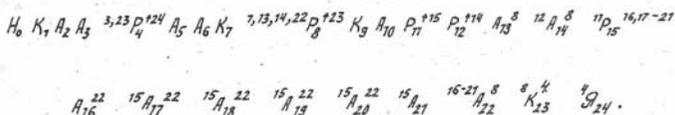
Когда оператор P_8 установит, что слово кончилось, т.е. все 7 символов данного слова расшифрованы, то необходимо переходить к следующему слову. Оператор K_{23} вычитает из счетчика слов единицу, а за ним работает оператор P_4 , проверяющий конец массива слов, конец текста. Если оператор P_4 установит, что $K = 0$, это означает конец работы - оператор A_{24} .

6. Структура операторной схемы программы "Текст"

- H_0 - оператор начала, заголовок.
- K_1 - начальная установка счетчика слов.
- A_2 - выбор Δx , Δy из адресов, указанных в 3-ей строке обращения.
- A_3 - задание угла поворота текста α .
- P_4 - проверка конца цикла слов ($K = 0?$).

- A₅ - выбор очередного слова.
- A₆ - выделение номера шрифта.
- K₇ - установка счетчика символов в слове.
- P₈ - проверка конца слова ($i = 0?$).
- K₉ - вычитание единицы из счетчика символов.
- A₁₀ - выделение очередного символа и подготовка следующего.
- P₁₁ - проверка: есть ли уход на служебное слово.
- P₁₂ - проверка признака служебного слова.
- A₁₃ - расшифровка символа.
- A₁₄ - запоминание обхода для служебного слова.
- P₁₅ - дешифрация служебных символов.
- A₁₆ - изменение масштаба.
- A₁₇ - изменение α .
- A₁₈ - переход к следующей строке.
- A₁₉, A₂₀, A₂₁ - служебные действия.
- A₂₂ - ликвидация обхода для служебного символа.
- K₂₃ - вычитание единицы из счетчика слов.
- A₂₄ - конец работы.

Операторная схема имеет вид:



7. ИНСТРУКЦИЯ К ПРОГРАММАМ "СИМВОЛ" , "ТЕКСТ"

Программы предназначены для вычерчивания всевозможных символов с помощью ЦВМ и графопостроителей.

Обращение к программе "Символ" имеет вид:

<i>Адрес</i>	<i>КОП</i>	<i>A₁</i>	<i>A₂</i>	<i>A₃</i>
<i>K</i>	<i>16</i>	<i>K+1</i>	<i>6632</i>	<i>6630</i>
<i>K+1</i>	<i>Nш, S</i>	<i>M</i>	<i>Δx / Δy</i>	<i>—</i>

Во второй строке обращения в КОП из 9 разрядов 3 разряда предназначены для номера шрифта, а 6 разрядов – для кода символа. M – адрес масштаба.

Программа располагается в МОЗУ на фиксированном месте.

Длина (начало 6100, конец 7077) ~ 1000₈

Рабочие ячейки I-3

Поле закодированных символов шрифта № 0 5000–5177

Время выполнения обращения к подпрограмме зависит от количества и размеров выводимых на графопостроитель изображений. В среднем следует исходить из скорости работы графопостроителя ~ 9 см/сек.

Обращение к программе "Текст" имеет вид:

<i>Адрес</i>	<i>КОП</i>	<i>A₁</i>	<i>A₂</i>	<i>A₃</i>
<i>K</i>	<i>16</i>	<i>K+1</i>	<i>6631</i>	<i>6630</i>
<i>K+1</i>	<i>00</i>	<i>α</i>	<i>Δx / Δy</i>	<i>ω</i>

где α – адрес начала текста,

ω – адрес конца текста,

Δx/Δy – адрес ячейки с числом Δx ; в следующей ячейке-

ке - Δy . Δx и Δy служат для определения угла наклона текста по отношению к осям координат.

8. Примеры кодировки текста

а. Без служебных символов

Пример № 1. Требуется закодировать следующий текст: "СИБИРСКОЕ ОТДЕЛЕНИЕ АКАДЕМИИ НАУК".

Отведем место в памяти для нашего текста, например, с ячейки 0100. Последовательно символ за символом выбирает из таблицы соответствующие коды и составляем "слова" нашего текста.

Адрес	КОП	A_1	A_2	A_3	
0100	0 61	50 41	50 60	61 52	СИБИРСК
0101	0 56	45 17	56 62	44 45	ОЕЛ ОТДЕ
0102	0 53	45 55	50 45	17 40	ЛЕНИНА
0103	0 52	40 44	45 54	50 50	КАДЕМИИ
0104	0 17	55 40	63 52	17 17	НАУК

Каждая ячейка - "слово" по терминологии, принятой в программе, состоит всегда из 7 символов. Начинается с восьмичисленной цифры в КОПе, указывающей, из какого шрифта символ данного слова. В нашем примере все символы из шрифта $\mathbb{R}O$.

Чтобы выполнить текст, необходимо задать две команды обращения, например, так:

ячейки	КОП	A_1	A_2	A_3
0010	16	0011	6631	6630
0011	00	0100	0000	0104

В случае, когда текст, который нужно закодировать, заканчивается внутри слова, оставшиеся места под коды символов должны быть заполнены кодами пробелов.

б. С использованием служебных символов.

Пример № 2. Требуется написать такой текст "С и б и р с к о е О т д е л е н и е", т.е. отличается от примера выше заглавными буквами (необходимо изменение

масштаба).

Адрес	КОП	A ₁	A ₂	A ₃	
0105	0 77	01 61	77 00	50 41	"М1" С "МО" и б
0106	0 50	60 61	52 56	45 17	ирское
0107	0 77	01 56	77 00	62 44	"М1" О "МО" т д
0110	0 45	53 45	55 50	45 17	еление

Обращение к программе имеет вид:

Адрес	КОП	A ₁	A ₂	A ₃
0011	16	0012	6631	6630
0012	00	0105	0000	0110

Пример № 3. Написать с несколькими поворотами:

" 0 1 2 3 4 5 6 7 8 9 "

Адрес	КОП	A ₁	A ₂	A ₃	
0111	0 77	21 00	01 02	77 22	"поворот 21" 012 "пов.22"
0112	0 03	04 05	77 20	06 07	345 "поворот 20" 67

Обращение к программе имеет вид:

Адрес	КОП	A ₁	A ₂	A ₃
0013	16	0014	6631	6630
0014	00	0111	0000	0112

Пример № 4. Требуется повернуть на угол

$\arctg \frac{\Delta y}{\Delta x}$. Текст примера № 1.

В памяти ЦВМ поместить в 2-х ячейках числа:

Ячейка	Число	Расшифровка
0 200	101 400 000	$1_{10} = \Delta x$
0 201	104 500 000	$10_{10} = \Delta y$

Берем пример № 1. Обращение к программе будет иметь вид:

Адрес	КОП	A ₁	A ₂	A ₃
0010	16	0011	6631	6630
0011	00	0100	0200	0104

9. ПРОГРАММЫ "СИМВОЛ" И "ТЕКСТ"

6630	070	0000	0000	0000	команда возврата	
1.	056	0000	6674	6703	начало "текста"	
2.	016	6713	6674	6703	начало "символа"	
3.	077	0000	0000	0000		
4.	000	0000	0000	0000	строка информации	
5.	077	7070	7000	0000	очередное слово	
6.	052	0000	0000	0000	№ шрифта	
7.	101	4000	0000	0000	масштаб	
6640	077	0000	0000	0000	очередной символ	
1.	000	0000	0077	0000	очередной символ в A2	
2.	770	7777	7777	7777	строка элементов символа	
3.	770	0000	0000	0000	рабочая ячейка	
4.	103	6000	0000	0000		
5.	301	4000	0000	0000	счетчик элементов	
6.	770	0000	0000	0000		
7.	400	0000	0000	0000	подъем	
6650	040	0000	0000	0000	спуск	
1.	440	0000	0000	0000		
2.	102	0000	0000	0000	} для формирования $\Delta x, \Delta y$	
3.	302	0000	0000	0000		
4.	101	4000	0000	0000		
5.	300	7700	7700	0000		
6.	000	0000	0000	0000		
7.	000	0000	0000	0000		
6660	052	0000	0000	0000	} выбор очередного слова	
1.	652	0001	0000	6660		
2.	400	0000	0000	6635		
3.	012	0000	6724	0000		
4.	000	0001	0000	0000		
5.	052	0000	0000	0000		
6.	013	6630	6725	6630		
7.	056	0000	6630	0000	конец массива слов	
6670	000	0000	0000	0000	восстановление PA	
1.	012	0001	6724	0000	увеличение на 1 адреса возвр.	
2.	000	7777	0000	0000	уход в основную программу	
3.	000	0000	7777	0000		
4.	472	0000	6630	6665	} настройка блока выборки слов	
5.	400	0000	0000	6634		
6.	072	0000	6634	6730		
7.	055	6634	6673	0000		
6700	436	0000	6703	6122		} выборка строки информации
1.	400	0001	0000	6123		
2.	016	6703	6324	6354		
3.	000	0000	0000	0000		
4.	054	0064	6634	0001		
5.	072	0000	0001	0000		
6.	452	0000	0000	6660		
7.	054	0130	6634	0001	формирование блока выборки слов	

6710	013	6671	0001	6663	
1.	056	0000	6660	0000	за 1-м словом
2.	000	0000	0000	0000	
3.	055	6634	6672	0001	← одиночный символ
4.	036	6634	6720	6635	
5.	013	6722	0001	6717	
6.	000	0000	0000	0000	
7.	000	0000	0000	6637	засылка масштаба
6720	016	6721	6724	6730	выбор № шрифта
1.	016	6664	6740	6732	исполнение символа с последующим уходом в основную програм.
2.	000	0000	0000	6637	
3.	000	0000	0000	0000	
4.	054	0042	6635	0001	новое слово
5.	072	0000	0001	0000	выделение № шрифта
6.	400	7050	0000	6636	
7.	000	6636	0000	6771	
6730	000	0000	0000	0000	
1.	032	0000	6733	0000	
2.	052	0000	0000	0000	Очередной символ
3.	652	0001	0000	6732	счетчик символов в слове
4.	112	0007	6740	0000	
5.	052	0001	0000	6732	конец слова
6.	056	0000	6660	0000	за следующим символом
7.	000	0000	0000	0000	
6740	055	6635	6633	6640	Анализ символа
1.	054	0106	6635	6635	выделение символа очередного
2.	067	6640	0000	6641	подготовка следующего символа
3.	000	0000	0000	0000	(обход в случае служебн.симв.)
4.	015	6640	6633	0000	
5.	076	0000	6770	0000	
6.	016	6747	6732	6743	очередной символ - служебный
7.	072	0000	6641	0000	
6750	132	0020	6755	0000	
1.	455	7060	6753	6637	$S < 20$ масштаб
2.	056	0000	6767	0000	
3.	777	7700	0000	0000	
4.	000	0000	0000	0000	
5.	132	0040	6763	7760	
6.	254	0117	7060	6643	$S < 40$ поворот
7.	016	6760	7012	7024	
6760	016	6761	6107	6111	
1.	007	6656	0000	6657	
2.	056	0000	6767	7024	
3.	000	0000	0000	0000	
4.	000	0000	0000	0000	
5.	000	0000	0000	0000	
6.	000	0000	0000	0000	
7.	056	0000	6732	6743	за следующим символом

					Расшифровка символа	
6770	072	0000	6641	6774	выбор строки элементов	
1.	652	0000	5000	0000		
2.	400	0000	0000	6642		
3.	000	6644	0000	6645		
4.	000	0000	0000	0000		
5.	003	6645	6654	6645		
6.	036	0000	7034	0000		
7.	055	6642	6646	6643	установка счетчика элементов обход после последн.элемента конец элементов? анализ элемента выделение элемента анализ 00?	
7000	036	0000	7027	6354		
1.	015	6643	6647	0000	подъем пера	
2.	076	0000	7004	0000		
3.	016	7026	6103	6111		
4.	015	6643	6650	0000	спуск пера	
5.	076	0000	7007	0000		
6.	016	7026	6104	6111		
7.	000	0000	0000	0000		
7010	000	0000	0000	0000		
1.	000	0000	0000	0000		
2.	052	0000	0000	0000		
3.	155	6643	6647	0000	формирование составляющих элемента Δx , Δy	
4.	136	6652	7016	6656		
5.	100	6653	0000	6656		
6.	055	6643	6655	0001		
7.	054	0070	0001	0001		
7020	513	6656	0001	6656		обращение к УГ (рисовка элемента) подготовка следующего за следующим элементом
1.	054	0103	6643	6643		
2.	505	6656	6637	6656		
3.	112	0001	7013	0001		
4.	052	0000	0000	0000		
5.	016	7026	6101	6111		
6.	000	6656	0000	6657		
7.	054	0106	6642	6642		
7030	056	0000	6774	0000		
1.	000	0000	0000	0000		
2.	100	0000	0000	0000		
3.	000	0000	0100	0000		
4.	054	0106	6642	0001	анализ последн.элемента строки уход на продолжение уход за следующим символом	
5.	076	0000	7037	6774		
6.	016	6732	6777	6774		
7.	033	6636	7033	7045		
7040	013	7045	7033	7045		
1.	073	0001	7032	0001		
2.	076	0001	7040	0000		
3.	054	0045	6642	0001		
4.	072	0000	0001	0000	выбор строки элементов на исполнение	
5.	652	0000	5000	0000		
6.	056	0000	6772	6774		
7.	000	0000	0000	0000		

7050	652	0000	5000	0000	строка	начала	каталога	шрифта	№	0
1.	652	0000	0000	0000	-	"	-	-	1	1
2.	652	0000	0000	0000	-	"	-	-	2	2
3.	652	0000	0000	0000	-	"	-	-	3	3
4.	652	0000	0000	0000	-	"	-	-	4	4
5.	652	0000	0000	0000	-	"	-	-	5	5
6.	652	0000	0000	0000	-	"	-	-	6	6
7.	652	0000	0000	0000	-	"	-	-	7	7
7060	101	4030	0000	0000	M0	=	1,0	} константы стандартных масштабов и поворотов		
1.	101	6035	0000	0000	M1	=	1,5			
2.	102	4037	0000	0000	M2	=	2,0			
3.	102	5017	0000	0000	M3	=	2,5			
4.	102	6007	0000	0000	M4	=	3,0			
5.	103	4057	0000	0000	M5	=	4,0			
6.	103	5077	0000	0000	M6	=	5,0			
7.	104	5075	0000	0000	M7	=	10,0			
7070	100	7070	0000	0000	M10	=	7/8			
1.	100	6071	0000	0000	M11	=	3/4			
2.	100	4073	0000	0000	M12	=	0,5			
3.	077	6053	0000	0000	M13	=	3/8			
4.	077	4003	0000	0000	M14	=	1/4			
5.	076	6013	0000	0000	M15	=	3/16			
6.	076	4033	0000	0000	M16	=	1/8			
7.	075	6031	0000	0000	M17	=	3/32			
5000	010	4020	0000	0001	СИМВОЛ		КОД		} Основной каталог шрифт № 0	
1.	120	2041	2000	0021	0		00			
2.	030	2041	1201	5031	1		01			
3.	030	2041	1201	5051	2		02			
4.	131	3045	6563	0061	3		03			
5.	232	3046	0600	6071	4		04			
6.	010	4020	2112	0101	5		05			
7.	030	2040	1202	0121	6		06			
5010	010	4011	1206	0131	7		07			
1.	320	4115	5605	1001	8		10			
2.	210	4020	2066	0141	9		11			
3.	030	4202	0402	7000	+		12			
4.	042	3230	0000	0121	-		13			
5.	200	4000	0000	0151	/		14			
6.	040	1100	5504	0300	:		15			
7.	403	0300	0000	0000	.		16			
5020	100	4030	7401	0161	†		17			
1.	131	3041	6601	2021	†		20			
2.	045	2021	2401	7170	(21			
3.	041	2025	2401	7270)		22			
4.	010	4222	2666	2171	x		23			
5.	222	2046	0604	0201	=		24			
6.	131	3045	0051	0211	:		25			
7.	045	0030	3105	0021	⌈		26			
					⌋		27			

					СИМВОЛ	КОД	
5030	041	0030	3501	0021]	30	Основной каталог шрифт № 0
1.	010	4222	2660	2221	*	31	
2.	130	3041	2560	0241	,	32	
3.	130	4120	1500	0241	,	33	
4.	010	4222	2551	0251	≠	34	
5.	130	4337	7374	0200	<	35	
6.	043	3733	7401	7000	>	36	
7.	200	4100	1500	5261	:.	37	
5040	041	3367	2570	0271	А	40	
1.	043	0110	2517	0301	Б	41	
2.	043	0110	1517	0321	В	42	
3.	232	3045	0000	0311	Г	43	
	4.	040	5013	0300	Д	44	
	5.	042	0206	0600	Е	45	
	6.	042	3632	7100	Ж	46	
	7.	030	2041	1201	З	47	
5050	040	3034	0110	0371	И	50	
1.	131	3041	0614	0371	И	51	
2.	040	3030	7051	1401	К	52	
3.	041	3132	0000	0021	Л	53	
4.	040	3033	6320	0021	М	54	
5.	040	3030	7301	0411	Н	55	
6.	010	4020	0000	0001	О	56	
7.	040	3033	0100	0021	П	57	
5060	040	2301	1025	1311	Р	60	
1.	232	2045	1605	5421	С	61	
2.	030	3042	0200	0431	Т	62	
3.	043	0110	2030	7441	У	63	
4.	330	4036	0550	6451	Ф	64	
5.	042	3632	7000	0361	Х	65	
6.	040	3030	7071	0461	Ц	66	
7.	030	0140	3030	7441	Ч	67	
5070	040	3030	7073	0501	Ш	70	
1.	040	3030	7073	0521	Щ	71	
2.	332	3040	7074	0531	Ъ	72	
3.	302	0000	0000	0531	Ь	73	
	4.	130	4300	2516	Э	74	
	5.	040	3030	7100	Ю	75	
	6.	042	2200	2027	Я	76	
	7.	077	0000	0000			
5100	000	2112	0150	7011		00	
1.	055	5605	1403	5300		01	
2.	070	7402	0000	0000		02	
3.	056	6650	5000	0041		03	
4.	202	0402	0000	0000		04	
5.	055	5150	0000	0011		05	
6.	105	0020	2000	0021		06	
7.	301	5050	0000	0011		07	

приложение № 1
шрифта № 0

					код	
5110	155	1605	5060	6111	10	
1.	021	1201	5050	0011	11	
2.	676	7403	0300	0000	12	
3.	510	1112	0150	0051	13	
4.	202	0402	6000	0000	14	
5.	015	0051	0564	0320	15	
6.	040	3200	7604	0300	16	
7.	262	6402	5000	0000	17	
5120	060	4202	0402	6000	20	
1.	014	0070	6040	5151	21	
2.	060	6026	0202	0231	22	
3.	606	2262	6403	5000	23	
	4.	051	0015	0402	7170	24
	5.	606	0400	0000	0201	25
	6.	400	3030	4000	0241	26
	7.	131	3200	7074	0200	27
5130	022	0205	0000	0311	30	
1.	506	0070	7403	0300	31	
2.	301	1015	1000	0311	32	
3.	017	0600	3133	0021	33	
4.	307	0032	0205	0311	34	
5.	070	7031	0000	0361	35	
6.	236	7274	0200	0000	36	
7.	550	4070	7232	3021	37	
5140	337	7374	0200	0000	40	
1.	030	7070	0402	0000	41	
2.	060	6152	0114	0250	42	
3.	006	0070	7402	0200	43	
4.	057	0510	3403	7370	44	
5.	152	0201	1025	1431	45	
6.	300	3030	0000	0471	46	
7.	070	7005	5403	1000	47	
5150	030	3070	7000	0511	50	
1.	300	3030	7402	7000	51	
2.	030	3070	7000	0461	52	
3.	736	3040	7073	0541	53	
	4.	110	2517	0403	6360	54
	5.	551	1201	5070	0011	55
	6.	020	2112	0150	7011	56
	7.	550	6153	0064	0200	57
5160	000	0000	0000	0000	60	
1.	000	0000	0000	0000		
2.	000	0000	0000	0000		
3.	000	0000	0000	0000		
4.	000	0000	0000	0000		
5.	000	0000	0000	0000		
6.	000	0000	0000	0000		
7.	000	0000	0000	0000		

приложение № 1
шрифта № 0

ЛИТЕРАТУРА

1. Будняк А.А., Петренко А.И., Сигорский В.П. - Автоматизация ввода и вывода геолого-геофизической информации для ЭЦВМ. Труды Укр. геол.-разв.ин-та, 1968, вып.22.

2. Герасимова О.И. и др. - Автоматический вывод геолого-геофизической графической информации из ЭЦВМ. ИГиГ СО АН СССР, СНИИГГиМС, Новосибирск, 1968.

3. Головкин В.А. и др. - Моделирование на ЭВМ обработки геолого-геофизических полей. Программа "Изолиния-2", НГУ, СНИИГГиМС, Новосибирск, 1970.

4. Дядюра В.А., Будняк А.А., Петренко А.И. - Автоматическое построение геофизических карт и графиков. В сб. "Геофизические исследования на Украине". Киев, 1969.

5. Крамак В.С., Перфильев Л.Г. - Программное обеспечение устройства "Атлас". Алма-Ата, 1968.

6. Литвиненко О.К., Русьянов Ю.Г. - Построение гравиметрических карт с помощью электронных цифровых вычислительных машин, информ.сооб. ОНТИ ВИЭМС, Москва, 1967, №17.

7. Ляшенко В.Ф. - Программирование для цифровых вычислительных машин М-20, БЭСМ-3М, БЭСМ-4, М-220. Сов. радио, М, 1967.

8. Романенко Ю.А., Шемякин М.Л. - Построение плавных графиков геолого-геофизических функций с помощью алгебраических полиномов третьего порядка. Проблемы региональной геологии и петрографии и методы геохимических и геофизических исследований. Изд. "Наука", Новосибирск, 1969.

9. Романенко Ю.А., Шубина Л.Е. - Генерация символов и знаков для графических устройств ЭЦВМ. "Материалы по геологии, геофизике и полезным ископаемым Сибири (к научно-технической конференции молодых ученых и специалистов, посвященной 100-летию со дня рождения В.И.Ленина)", 1970.

СО Д Е Р Ж А Н И Е

ЧАСТЬ 1. МЕТОДЫ ПОСТРОЕНИЯ КОНТУРНЫХ ГЕОЛОГО-ГЕОФИЗИЧЕСКИХ ЛИНИЙ	5
§ 1. Метод построения контурной линии с помощью полиномов третьей степени	6
1. Описание операторной схемы программы "Контур-1"	11
2. Инструкция к программе "Контур-1"	16
3. Контрольный пример к программе "Контур-1"	19
4. Программа "Контур-1" (на Альфа-языке)	23
§ 2. Метод построения контурной линии дугами окружностей	27
1. Описание операторной схемы программы "Контур-2"	33
2. Инструкция для работы с программой "Контур-2"	37
3. Программа "Контур-2" (на Альфа-языке)	43
§ 3. Метод построения контурной линии с применением прямых и окружностей	47
1. Описание блок-схемы программы "Контур-3"	58
2. Инструкция к программе "Контур-3"	77
3. Программа "Контур-3" (на Альфа-языке)	86
§ 4. Метод построения контурной ломаной линии	95
1. Описание функциональной блок-схемы	101
2. Описание работы операторной схемы алгоритма	103
3. Структура операторной схемы алгоритма	105
4. Описание работы операторной схемы программы УГ (управление графопостроителем)	107
5. Инструкция к программе "Управление графопостроителем"	115
6. Программа "Управление графопостроителем"	117

ЧАСТЬ 2. ЭЛЕМЕНТЫ АВТОМАТИЧЕСКОГО ОФОРМЛЕНИЯ ГРАФИКОВ	
	В ГЕОЛОГИИ И ГЕОФИЗИКЕ 121
§ 1.	Описание блок-схем программ "График" 122
1.	Описание блока "Выбор формата и смена листа" . . . 125
2.	Описание блока "Кадрирование" 130
3.	Описание блока "Масштабирование" 134
4.	Описание блока "Выбор шага осей" 138
5.	Описание блока "Рамка" 139
6.	Описание блока "Координатная ось" 139
7.	Описание блока "Координатная сетка" 142
8.	Описание блока "Контроль вычерчивания кривых" . . 146
9.	Инструкция к программе "График" 151
10.	Программа "График" 154
§ 2.	Автоматическая рисовка символов для геологических построений 175
1.	Принятие определения 175
2.	Кодировка 177
3.	Описание работы операторной схем программы "Символ" 182
4.	Структура операторной схем программы "Символ" . . 184
5.	Описание работы операторной схем программы "Текст" 186
6.	Структура операторной схем программы "Текст" . . . 188
7.	Инструкция к программам "Символ", "Текст" 190
8.	Примеры кодировки текста 191
9.	Программы "Символ" и "Текст" 193
	Л и т е р а т у р а 199

Технический редактор *Л. А. Панина*

Подписано к печати 2. VII. 71 г. МН 03586
Бумага 60×84/16. Печ. л. 12,75. Уч.-изд. л. 11,50.
Тираж 500. Заказ 270. Цена 40 коп.

Институт геологии и геофизики СО АН СССР
Новосибирск, 90. Ротапринт.

О П Е Ч А Т К И

страница	строка	напечатано	должно быть
89	69	$x_1[N] := x[1] - x[x - 1]$	$x_1[N] := x[1] - x[N - 1]$
44	34	$x[l] := e ;$	$x[l] := e ;$
44	26	$M_2 : e := \underline{\text{истина}}$	$M_2 : L_2 := \underline{\text{истина}}$
26	64	, <u>если l то</u>	, <u>если l' то</u>
II5	4снизу	6II0	6III
II6	10сверху	6II0	6III
96	5снизу	$y_i = E\left(\frac{ Y_i }{h} + 0,5\right) \cdot \frac{Y_i}{ Y_i }$	$y_i = E\left(\frac{ Y_i }{h} + 0,5\right) \cdot \frac{Y_i}{ Y_i } \cdot h$
96	6снизу	$x_i = E\left(\frac{ X_i }{h} + 0,5\right) \cdot \frac{X_i}{ X_i }$	$x_i = E\left(\frac{ X_i }{h} + 0,5\right) \cdot \frac{X_i}{ X_i } \cdot h$
99	3снизу	OA	O'A
99	12снизу	Рис.20	Рис.2I
100	9сверху	OA'	O'A