

05-8-2442
06-2-850

Воронежский государственный
архитектурно-строительный университет

224

Кафедра математического моделирования
и вычислительной техники

**ОСНОВЫ АЛГОРИТМИЗАЦИИ
ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ**

*Методические указания по курсу
"Информатика" для студентов 1-го курса
всех специальностей*

Воронеж 2005

Библиотека ВГАСУ

Составители В.П. Авдеев, Г.Т. Венгерова, В.И. Гильмутдинов,
А.Д. Кононов, А.А. Кононов

УДК 69.003:658.5.012.22
ББК 32.973

Основы алгоритмизации вычислительных процессов [Текст]: метод. указания по курсу «Информатика» для студ. 1-го курса всех спец. / Воронеж. гос. арх. - строит. ун-т; сост.: В.П. Авдеев, Г.Т. Венгерова, В.И. Гильмутдинов, А.Д. Кононов, А.А. Кононов. – Воронеж, 2005. - 41 с.

Рассматриваются практические примеры, используемые при построении алгоритмов различной структуры: линейных, разветвляющихся, циклических. Приводятся примеры, после подробного разбора которых предлагаются контрольные вопросы и упражнения для проверки усвоения студентами изучаемого материала. Методические указания предназначены для использования при изучении дисциплины «Информатика» студентами первого курса всех специальностей.

Ил. 36. Библиогр.: 5 назв.

Печатается по решению редакционно-издательского совета Воронежского государственного архитектурно-строительного университета

Рецензент – Ю.С. Радченко, к. ф. - м. н., доц. Воронежского государственного университета

Введение

Методические указания посвящены изучению основ алгоритмизации. Наверняка можно утверждать, что каждый, читающий эти строки, знаком с термином "алгоритм". Его применяют весьма широко, и не только в области вычислительной техники и программирования. Понятие алгоритма, относящееся к фундаментальным концепциям информатики, возникло задолго до появления ЭВМ и стало одним из основных понятий математики.

Разработка алгоритма - необходимый этап в процессе подготовки и решения задачи на ЭВМ, и в связи с этим алгоритмы представляют самостоятельную ценность как интеллектуальные ресурсы общества.

Цель методических указаний - помочь студентам в овладении умением конструировать алгоритмы, которые для наглядности представляются в виде блок-схем, повышение эффективности самостоятельной работы студентов по построению типовых структур алгоритмов.

Рассматриваются практические примеры, используемые при построении алгоритмов различной структуры: линейных, разветвляющихся, циклических. Приводятся примеры, после подробного разбора которых предлагаются контрольные вопросы и упражнения для проверки усвоения студентами изучаемого материала.

Предполагается при дальнейшем обучении использовать предложенные задания для решения на ЭВМ на конкретном алгоритмическом языке.

I. Основы алгоритмизации

Одним из важнейших этапов подготовки и реализации вычислительного процесса на ЭВМ является алгоритмизация решаемой задачи. Для пояснения термина "алгоритм" можно сказать, что это понятное и точное предписание (указание) исполнителю совершить определенную последовательность действий для достижения поставленной цели. Вычислительный алгоритм можно определить как последовательность действий по переработке исходных данных в результаты. Другими словами, алгоритм - точное предписание, которое задает алгоритмический процесс, начинающийся с произвольных исходных данных (из некоторой совокупности возможных для данного алгоритма исходных данных) и направленный на получение полностью определенного этими исходными данными результата.

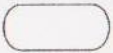
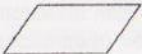
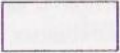


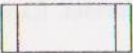

Алгоритмический процесс - процесс последовательного преобразования конструктивных объектов (слов, букв, чисел, предложений, графов, логических выражений и т. п.), происходящий дискретными шагами. Каждый шаг состоит в смене одного конструктивного объекта другим.

Если алгоритм представляет собой последовательность инструкций, которые могут быть выполнены на ЭВМ (непосредственно или после автоматической обработки - трансляции, состоящей в приведении алгоритма к исполнимому в ЭВМ виду), то такой алгоритм называется программой. Уст-

ройство ЭВМ, которое выполняет инструкции программы, называется центральным процессором (ЦП); с ним непосредственно связано одно из устройств ЭВМ, именуемое оперативной памятью (ОП). ЦП и ОП образуют вычислительный комплекс ЭВМ - вычислитель. ОП состоит из элементов - ячеек памяти, каждая из которых используется для запоминания числа или цифрового кода. Ячейка памяти имеет следующие свойства:

- а) в ней значение может сохраняться до выключения ЭВМ;
- б) если в ячейку не заносилось значение, она имеет неопределенное состояние, воспринимаемое как некоторое случайное значение;
- в) занесение в ячейку нового значения приводит к автоматическому стиранию прежнего;
- г) хранимое в ячейке значение может многократно использоваться в вычислениях.

В теории алгоритмов изучаются алгоритмы, заданные в строгом формализованном виде. Алгоритм можно представить различными способами: с помощью графического или словесного описания, в виде таблицы, последовательностью формул, записью на алгоритмическом языке (языке программирования). На практике в программировании чаще всего используется наглядное задание алгоритмов в виде блок-схем. Алгоритм в этом случае представляется графически в виде последовательности блоков, выполняющих определенные функции. Блоки соединяются стрелками, показывающими связи между ними. Внутри блоков указывается информация, характеризующая выполняемые ими функции, которые записываются словесно или с помощью формул.

-  - блок пуск-останов. Определяет начало-конец прерывание процесса вычисления;
-  - блок ввода-вывода информации;
-  - блок вычислений;
-  - блок проверки выполнения условия (логический блок);
-  - начало-конец цикла (модификация);
-  - вычисление по подпрограмме, стандартной программе;
-  - печать результатов на бумаге;



- линии потока, изображают последовательность связей между блоками;
- соединители, указывают связи между прерванными линиями потока, связывающими блоки;
- пояснения, содержание подпрограмм, формулы.

Совокупность вычислительных процессов, используемых при решении математических, экономических, научно-технических и др. задач на ЭВМ, в общем виде может быть разделена на три группы: линейные, разветвляющиеся и циклические. Структура алгоритма находится в прямой зависимости от типа отображаемого вычислительного процесса.

Контрольные вопросы и упражнения

1. Дать определение алгоритма.
2. Дать определение алгоритмического процесса.
3. Дать определение программы.
4. Дать определение центрального процессора, оперативной памяти, вычислителя.
5. Назвать свойства ячейки памяти.
6. Перечислить изобразительные средства алгоритмов.
7. Дать определение блок-схемы.
8. Перечислить известные блоки и указать их назначение.
9. Перечислить типы вычислительных процессов.

II. Линейный вычислительный процесс

Линейным называется вычислительный процесс, в котором самостоятельные этапы вычислений выполняются последовательно один за другим (в естественном порядке).

На блок-схеме линейный вычислительный процесс представляется последовательностью блоков, которые размещаются сверху вниз в порядке их выполнения.

Пример 1. Составить блок-схему вычисления выражения

$$y = \frac{(AX + B)^2}{\sqrt{A^2 + B^2}}$$

Необходимо наряду со значением y вывести значения числителя и знаменателя. На рис.2.1 приведена блок-схема вычисления требуемых значений данного выражения.

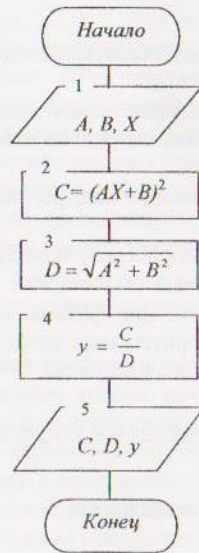


Рис 2.1

Пример 2. Найти сумму членов бесконечной геометрической прогрессии

$$\frac{1}{2} - \frac{1}{4} + \frac{1}{8} - \frac{1}{16} + \dots$$

Решение. Первый член геометрической прогрессии $x_1 = \frac{1}{2}$, а значит знаменатель

$$g = -\frac{1}{2}.$$

Сумма членов бесконечной убывающей геометрической прогрессии $S = \frac{x_1}{1-g}$. Эта формула используется для решения задачи (рис.2.2).

Задание.

Самостоятельно обобщить алгоритм на случай геометрической прогрессии с произвольными значениями x_1 и g .

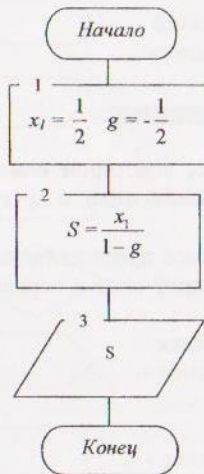


Рис.2.2

Пример 3. Составить блок-схему алгоритма для нахождения координат центра масс трех точек, массы которых m_1, m_2, m_3 , координаты их $(x_1, y_1), (x_2, y_2), (x_3, y_3)$.

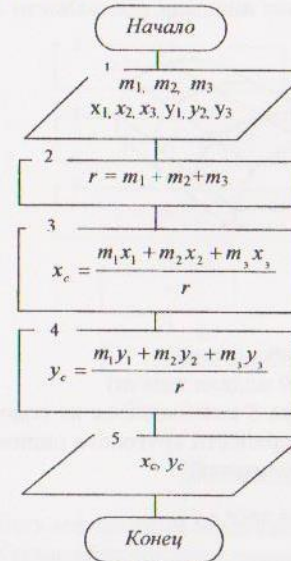


Рис. 2.3

Решение. Формулы для определения искоемых координат

$$x_c = \frac{m_1 x_1 + m_2 x_2 + m_3 x_3}{m_1 + m_2 + m_3},$$

$$y_c = \frac{m_1 y_1 + m_2 y_2 + m_3 y_3}{m_1 + m_2 + m_3}$$

и соответствующая блок-схема изображена на рис.2.3.

Пример 4. Тело брошено вертикально вверх с начальной скоростью v . Определить, через какое время тело достигает наивысшей точки и на каком расстоянии от земли.

Решение. Путь, пройденный телом, можно определить из уравнения

$$S = vt - \frac{gt^2}{2}, \quad (1)$$

где v - скорость, g - ускорение свободного падения, t - время движения тела.

Дифференцируя путь по t , получаем скорость $\frac{ds}{dt} = v - gt$.

В момент, когда тело достигает наивысшей точки, скорость его равна 0. Следовательно, $v - gt = 0$, откуда $t = \frac{v}{g}$. Зная t ,

можно определить расстояние, пройденное телом за это время. Таким образом, получены две формулы, необходимые для решения данной задачи.

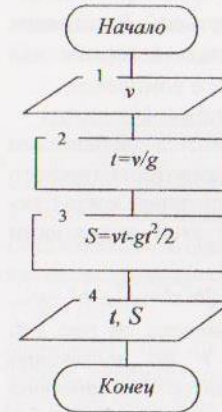


Рис.2.4

Эти формулы определяют численный метод решения, алгоритм которого приведен на рис.2.4.

Пример 5. Ангар длиной l имеет поперечное сечение в форме полукруга радиуса r (рис.2.5). Составить алгоритм вычисления площади поверхности S и объема V ангара.

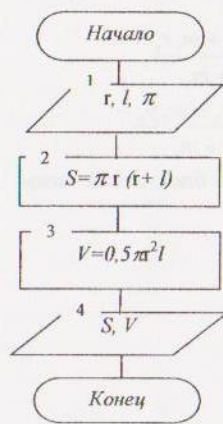


Рис. 2.6

Математическая формулировка решения
(построение математической модели задачи)
Площадь поверхности ангара S складывается из половины площади боковой поверхности кругового цилиндра и удвоенной площади полукруга:

$$S = \frac{1}{2}(2\pi r l) + 2\left(\frac{1}{2}\pi r^2\right) = \pi r(r + l),$$

а объем V равен половине объема кругового цилиндра:

$$V = \frac{1}{2}(\pi r^2 l).$$

Исходными данными для вычислений являются числовые значения переменных r, l (рис.2.6).

Пример 6. Строящийся животноводческий комплекс K будет обслуживать жителей трех деревень X, Y, Z . Заботясь об удобствах жителей, решено построить его на одинаковом удалении от центров этих деревень. Расстояния между деревнями соответственно a, b, c (рис.2.7). Составить алгоритм для вычисления расстояния r от деревень до животноводческого комплекса.

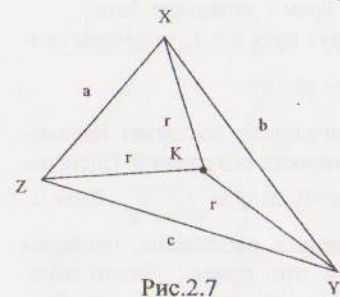


Рис.2.7

Математическая формулировка решения
Центры деревень являются вершинами треугольника, а центр животноводческого комплекса - центром описанной вокруг него окружности. Радиус этой окружности выражается формулой

$$r = abc / \sqrt{2a^2b^2 + 2a^2c^2 + 2b^2c^2 - a^4 - b^4 - c^4}.$$

Схема алгоритма приведена на рис. 2.8. Вычисление значения r по достаточно громоздкой формуле в алгоритме заменено тремя последовательно выполняемыми более простыми операциями с использовани-

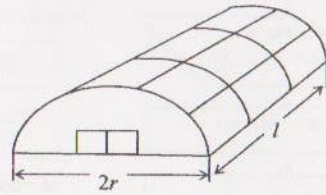


Рис. 2.5

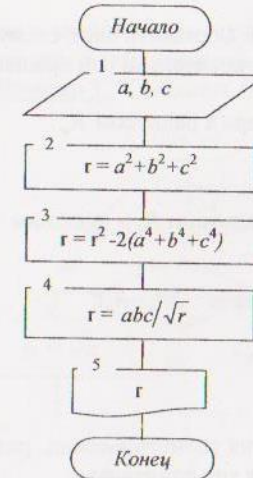


Рис. 2.8

ем промежуточных переменных, которые могут быть обозначены любыми буквами. Однако при переходе от алгоритма к программе объем занимаемой памяти будет тем больше, чем больше различных переменных используется в алгоритме. Поэтому алгоритм и программу целесообразно составлять, используя последовательные присвоения изменяющихся значений одним и тем же переменным. В алгоритме на рис.2.8 такой подход позволяет обойтись в процессе вычислений одной промежуточной переменной r .

Контрольные вопросы и упражнения

1. Дать определение линейного вычислительного процесса.
2. Каким образом представляется на блок-схеме линейный вычислительный процесс?
3. Составить блок-схему вычисления следующих выражений:

а)
$$P = \frac{NK^2 - M}{(K + L)^2},$$

вывести значения P , числителя и знаменателя;

б)
$$Y = \sqrt{AX^2 + B} - AB,$$

вывести значения Y и подкоренного выражения.

4. Составить блок-схему алгоритма для вычисления по формуле

$$Y = \frac{\sin(ax^2 + bx + c) + 2\cos(ax^2 + bx + c)}{\sqrt{ax^2 + bx + c}}, \text{ где } ax^2 + bx + c \neq 0.$$

5. Составить блок-схему алгоритма для вычисления длины окружности L , если задана площадь соответствующего круга S .
6. Составить блок-схему для чтения температуры в градусах Цельсия и печати ее в градусах Кельвина, Фаренгейта, Реомюра ($T^{\circ}K = T^{\circ}C + 273,15$; $T^{\circ}R = 0,8T^{\circ}C$; $T^{\circ}F = 1,8T^{\circ}C + 32^{\circ}$).
7. Даны величины A и B . Найти их сумму, произведение, среднее арифметическое, среднее геометрическое. Составить блок-схему.
8. Даны две числовые величины A и B . Поменять местами содержимое ячеек, в которых они находятся:
 - а) с использованием вспомогательной ячейки;

в) без использования вспомогательной ячейки. Составить блок-схему.
 9. Моменты инерции J твердых тел с массой m относительно оси вращения выражаются следующими зависимостями:

а) для сплошного однородного цилиндра с радиусом R_1 :

$$J = \frac{1}{2} m R_1^2;$$

б) для полого цилиндра с внутренним радиусом R_1 и внешним радиусом R_2 :

$$J = \frac{1}{2} m (R_1^2 + R_2^2) \quad (\text{считать в задаче } 2R_1 = R_2);$$

в) для однородного шара с радиусом R_m :

$$J = \frac{2}{5} m R_m^2.$$

Составить блок-схему алгоритма для определения геометрических размеров указанных тел, если J и m известны и для всех тел одинаковы.

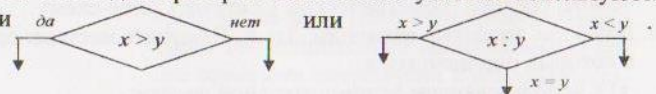
10. Для приготовления известково-цементного состава необходимо $A\%$ известкового теста, $B\%$ белого цемента, $C\%$ поваренной соли, $D\%$ пигмента, остальное составляет вода. Составить алгоритм для расчета необходимого количества воды $F[kg]$ при приготовлении $Q[kg]$ состава. Составить блок-схему.

11. Цех централизованного приготовления малярной продукции за одну смену производит: синтетической шпаклевки $A[t]$, шпатлевки масляной $B[t]$, грунтовок глиноземной $C[t]$, пасты меловой $D[t]$, замазки меловой $E[t]$, масляных окрашивающих составов $G[t]$, водно-масляной эмульсии $H[t]$. Составить блок-схему алгоритма для расчета количества выпускаемой продукции за месяц, если в первую декаду цех работал в одну смену, вторую - в полторы смены, третью - в две смены. Результаты вывести на печать.

III. Разветвляющийся вычислительный процесс

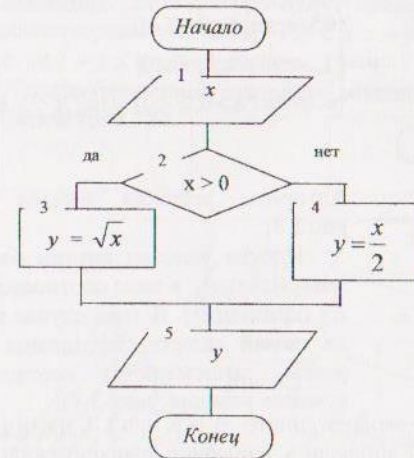
Вычислительный процесс называется разветвляющимся, если в зависимости от исходных условий или промежуточных результатов он выполняется по одному из нескольких возможных направлений, которые называются ветвями вычисления.

Выбор ветви вычислений определяется проверкой выполнения логического условия, отражающего свойства исходных данных или промежуточных результатов. После проверки логического условия в каждом конкретном случае процесс реализуется только по одной ветви, а выполнение остальных исключается. В блок-схеме для проверки логического условия используются логические блоки



Ветви в свою очередь также могут быть разветвленными, то есть могут также содержать блоки проверки выполнения условий (блоки ЕСЛИ).

Пример 1. Составить блок-схему вычисления переменной y



$$y = \begin{cases} \sqrt{x}, & \text{если } x > 0; \\ \frac{x}{2}, & \text{если } x \leq 0. \end{cases}$$

На рис 3.1 приведена блок-схема вычисления значений y . Ветвление происходит по двум взаимоисключающим ветвям.

Рис.3.1

Пример 2. Постоянная времени электрической цепи равна $T=RC$, где

R и C - соответственно сопротивление и емкость цепи. Составить алгоритм для определения R и C , если при значениях $T < T_0 = R_0 C_0$ выбирается только R , а C постоянна и равна C_0 . В противном случае выбирается C , а R постоянна и равна R_0 . Решение задачи приведено на рис 3.2.

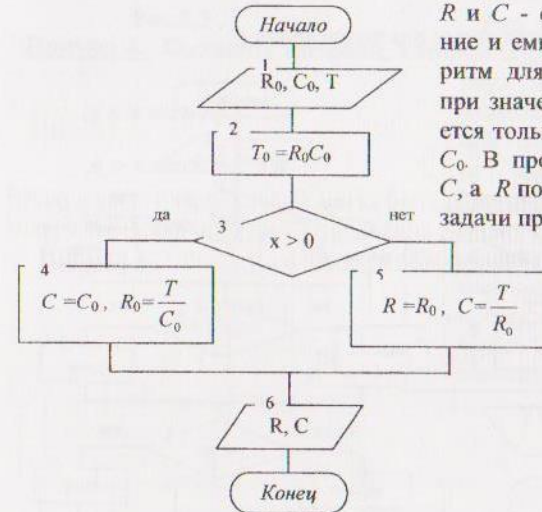


Рис.3.2

Пример 3. Если A и B отрицательные, присвоить y значения 0,5; если A и B положительные, присвоить y значение 1; если A и B имеют противоположные знаки, присвоить значение 0. Составить алгоритм.

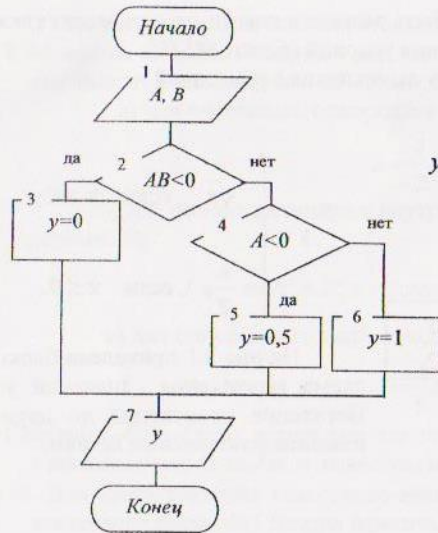


Рис.3.3

Решение.

Построим математическую модель задачи

$$y = \begin{cases} 0,5, & \text{если } A < 0, B < 0, \\ 1, & \text{если } A \geq 0, B \geq 0, \\ 0, & \text{если } (A > 0, B < 0) \text{ или } (A < 0, B > 0). \end{cases}$$

Алгоритм решения показан на рис 3.3.

Иногда условия внутри блока записывается в виде соотношения $x:y$ (сравнения). В этом случае возле линий потока, исходящих из ромба, записываются соответствующие условия (рис. 3.4).

Пример 4. Составить блок схему для вычисления F

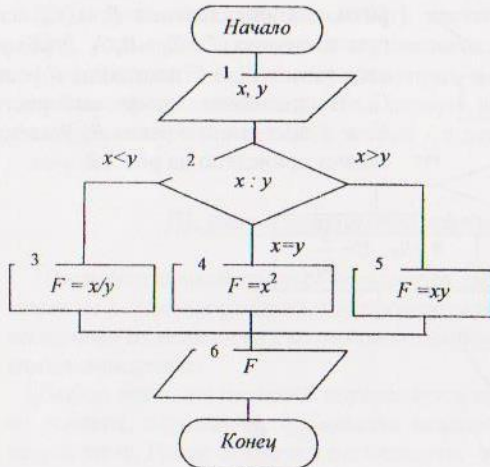


Рис. 3.4

$$F = \begin{cases} xy, & \text{если } x > y, \\ \frac{x}{y}, & \text{если } x < y, \\ x^2, & \text{если } x = y. \end{cases}$$

Пример 5. Точка A задана координатами x, y . Определить, принадлежит ли эта точка фигуре на плоскости (рис. 3.5).

Пояснение. Этой фигуре будут принадлежать точки, координаты которых удовлетворяют условиям $y \geq 0$ и $x + y \leq 1, -x + y \leq 1$, т.е. $|x| + y \leq 1$ одновременно.

Теперь блок-схема алгоритма решения задачи будет выглядеть следующим образом (рис.3.6).

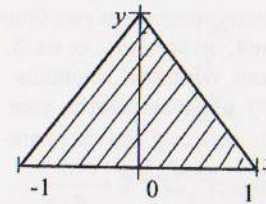


Рис.3.5

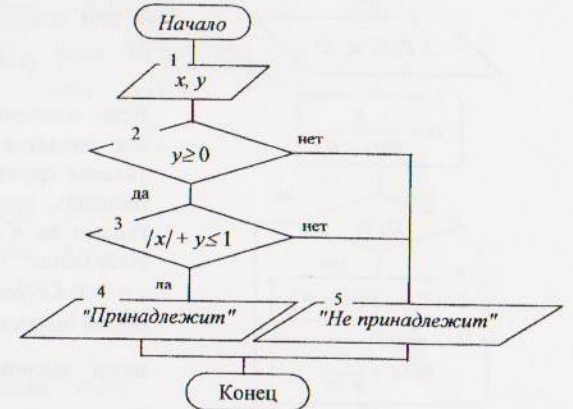


Рис. 3.6

Пример 6. Составить алгоритм, который определяет длину общей части двух отрезков числовой оси, заданных координатами своих концов соответственно a, b и c, d ($a < b; c < d$).

Решение. Если отрезки имеют общую часть, то левая координата общей части отрезков m равна максимальному из чисел a и c , а правая n - минимальному из чисел b и d . Отсюда вытекает алгоритм, показанный на рис.3.7.

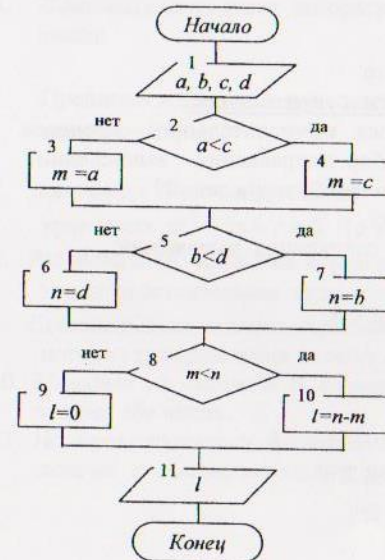


Рис.3.7

Пример 7. Бригаде строителей численностью N человек необходимо оштукатурить стены общей площадью $S \text{ м}^2$. Ей в помощь на K дней придана вторая бригада численностью M человек. Ежедневная производительность труда одного штукатура $P \text{ м}^2$. Составить алгоритм для определения количества дней для выполнения всей работы.

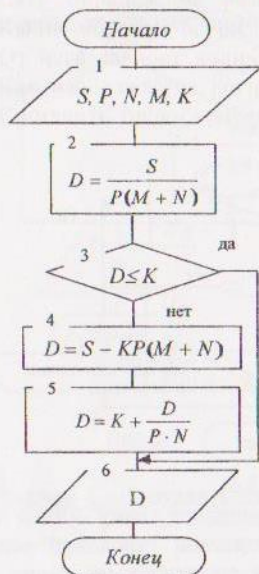


Рис. 3.8

Контрольные вопросы и упражнения.

1. Дать определение разветвляющегося вычислительного процесса.
2. Каким образом осуществляется выбор направлений вычислений?
3. От чего зависит количество направлений вычислений?
4. Как изображается логический блок?
5. Составить блок-схему вычисления следующих выражений:

$$а) y = \begin{cases} x^A, & \text{если } 0 < A < 3, \\ \sqrt{x+A}, & \text{если } A \geq 3; \end{cases}$$

$$б) y = \begin{cases} \sqrt{x}, & \text{если } x \leq 1, \\ 2-x, & \text{если } 1 < x \leq 2, \\ \sin(x-2), & \text{если } x > 2; \end{cases}$$

Математическая формулировка решения

Сначала вычисляется количество дней D при совместной работе двух бригад:

$$D = \frac{S}{P(M+N)}.$$

Если значение D не превышает K , то оно является решением задачи. В противном случае необходимо определить площадь, оштукатуренную двумя бригадами за K дней, и вычесть ее из S . Найденная таким образом площадь $\sigma = S - KP(M+N)$ обрабатывается уже одной бригадой, поэтому теперь решением задачи будет $D = K + \frac{\sigma}{P \cdot N}$.

Блок-схема алгоритма приведена на рис 3.8

$$в) y = \begin{cases} ax^2 + bx + c, & \text{если } K = 1, \\ dx^2 + ex + f, & \text{если } K = 2, \\ qx^2 + hx + i, & \text{если } K = 3; \end{cases}$$

$$г) F = \begin{cases} Ax^2, & \text{если } -2 < x < 10, \\ \sqrt{A^2 + x^2}, & \text{если } 10 \leq x < 100, \\ A, & \text{если } x = 100; \end{cases}$$

$$е) y = \sqrt{2z^2 + 3z + 1}$$

$$z = \begin{cases} Ax^2, & \text{если } x < 0, \\ 1, & \text{если } x = 0, \\ \sqrt{x}, & \text{если } x > 0; \end{cases}$$

$$ж) F = Ax^3$$

$$x = \begin{cases} B+C, & \text{если } B > C, \\ B^2, & \text{если } B = C. \end{cases}$$

6. Составить блок-схему алгоритма нахождения корней квадратного уравнения

$$ax^2 + bx + c = 0.$$

Предполагается, что для заданных значений a, b, c допустимы комплексные корни.

7. Вычислить неотрицательные действительные корни квадратного уравнения $ax^2 + bx + c = 0$ ($a \neq 0$).
8. Вычислить комплексные корни квадратного уравнения $ax^2 + bx + c = 0$, у которых действительная часть $\alpha = -b/2a$ положительна.
9. Составить блок-схему алгоритма нахождения максимального (минимального) из трех заданных чисел.
10. Меньшее из чисел X и Y заменить нулем, а в случае $X=Y$, заменить нулями оба числа.
11. По представленным блок-схемам и исходным данным определить численные значения выходных переменных.

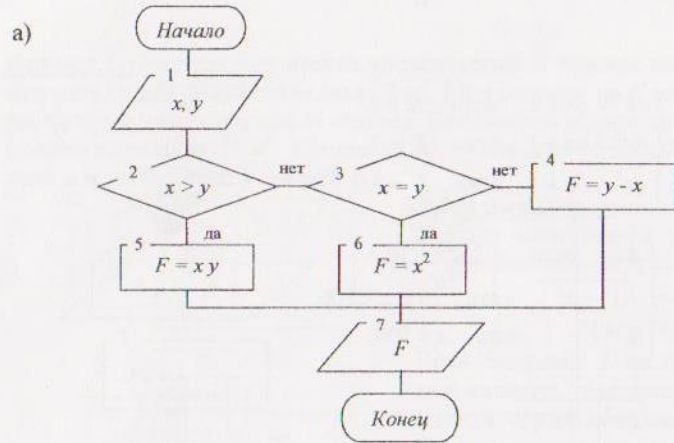


Рис.3.9

- 1) $\begin{cases} x = 10, \\ y = 15; \end{cases}$ 2) $\begin{cases} x = 1, \\ y = 1; \end{cases}$ 3) $\begin{cases} x = 25,5, \\ y = 13. \end{cases}$

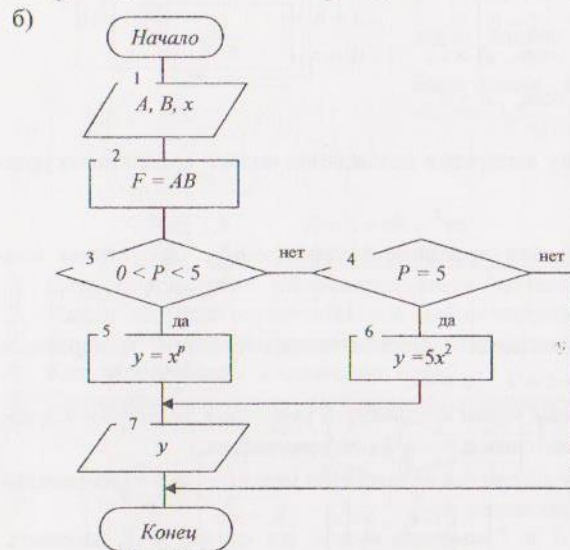


Рис.3.10

- 1) $\begin{cases} A = 2, \\ B = 1, \\ x = 3; \end{cases}$ 2) $\begin{cases} A = 5, \\ B = 6, \\ x = 1; \end{cases}$ 3) $\begin{cases} A = 2,5, \\ B = 2, \\ x = 2. \end{cases}$

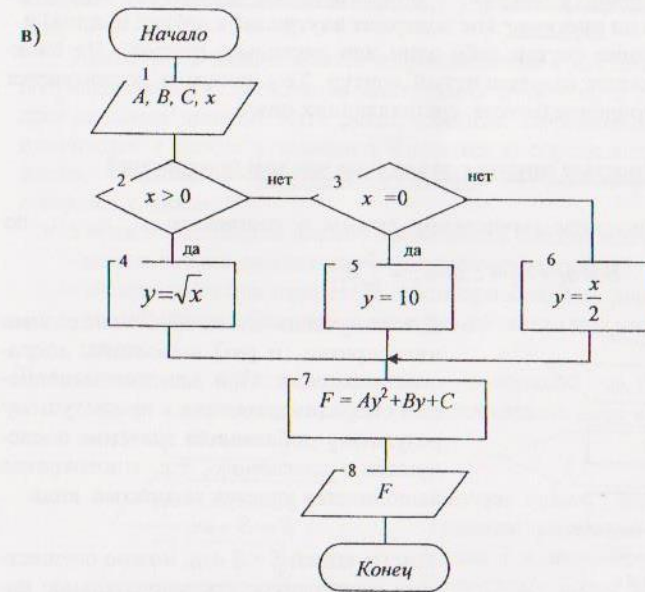


Рис. 3.11

- 1) $\begin{cases} x = 4, \\ A = B = C = 1; \end{cases}$ 2) $\begin{cases} x = 0, \\ A = B = 1, \\ C = 0; \end{cases}$ 3) $\begin{cases} x = -6, \\ C = A = 0, \\ B = 1. \end{cases}$

12. Составить алгоритм, в результате выполнения которого все числа x, y, z удваиваются, если $x \leq y \leq z$, и заменяются на их абсолютные величины в противном случае.

13. Вывести номер четверти координатной плоскости, которой принадлежит заданная точка.

14. Самолет летит из пункта A в пункт B со средней скоростью V . Составить алгоритм для нахождения времени полета t_1 , если есть встречный ветер, скорость которого V_1 , и времени t_2 , если ветра нет. Расстояние между пунктами A и B считать известным и равным S .

15. По условию предыдущей задачи составить алгоритм для нахождения t_3 , если возможен и попутный ветер, скорость которого V_2 .

IV. Циклический вычислительный процесс

Циклы - многократно повторяемые этапы вычислений. Вычислительные процессы, содержащие циклы, называются циклическими. В зависимости от количества повторений цикла различаются два основных типа: циклы с из-

вестным числом повторений и итерационные циклы. По структуре циклы разбиваются на простые (не содержат внутри себя других циклов) и сложные (содержащие внутри себя один или несколько циклов). На блок-схеме цикл изображают как замкнутый контур. Тем самым и показывается неоднократное выполнение блоков, составляющих цикл.

4.1 Простые циклы с заданным числом повторений

Рассмотрим алгоритм вычисления суммы n слагаемых a_1, a_2, \dots, a_n по

формуле $S = a_1 + a_2 + \dots + a_n = \sum_{i=1}^n a_i$.

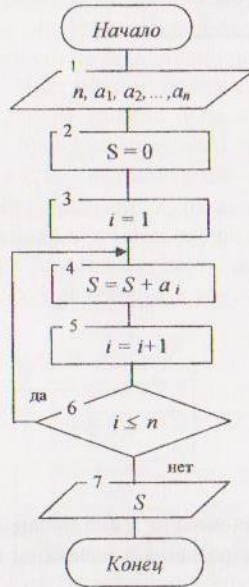


Рис.4.1

Для того, чтобы при каждом очередном выполнении инструкции $S=S+a_i$ слагаемое a_i было новым, в цикле наряду с этой инструкцией должна выполняться инструкция перехода к следующему слагаемому $i=i+1$; до цикла должно быть выполнено действие $i=1$.

Инструкции цикла не должны выполняться бесконечно: последним слагаемым должно быть слагаемое a_n . Поэтому в контур цикла включается блок проверки текущего значения i . Если это значение еще не превышает n , то вновь выполняются инструкции $\{S=S+a_i \text{ и } i=i+1\}$. Когда значение i превысит n , их выполнение должно прекратиться. Поэтому указанный блок в общем случае называется блоком проверки условия и изображается с двумя выходами (рис.4.1, блок б): линия потока, изображающая один из выходов, входит в контур цикла, другая, соответствующая случаю $i > n$,

Для получения значения S необходимо многократно (n раз) выполнить операцию сложения. При каждом выполнении операции сложения к предыдущему результату добавляется значение последующего слагаемого, т.е. многократно выполняется участок алгоритма вида

$$S = S + a_i.$$

Инструкцией $S = S + a_i$ можно осуществить постепенное последовательное накопление суммы, если предварительно выполнено действие $S=0$.

означает завершение цикла и передачу управления на блок вывода результатов.

Аналогично накапливается и произведение с той лишь разницей, что для его накопления используется инструкция $S=S * a_i$, а начальное значение произведения должно быть равно единице. Переменная, значение которой изменяется в цикле в заданных пределах и определяет момент окончания цикла, называется параметром цикла (в рассматриваемом примере i -параметр цикла).

Таким образом, для циклов с известным числом повторений задаются:

- начальное и конечное значения параметра цикла;
- закон изменения параметра цикла при каждом его повторении;
- количество необходимых повторений цикла или условие окончания цикла.

Для циклов с известным числом повторений в блок-схемах можно использовать блок вида

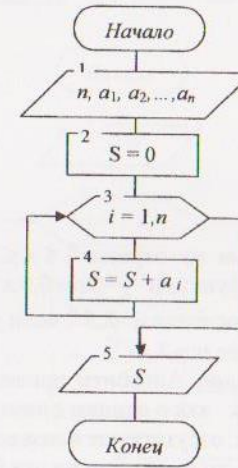
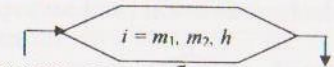


Рис. 4.2

использовать блок вида



Внутри этого блока записываются границы изменения m_1 и m_2 параметра цикла i и шаг. Это позволяет сделать блок-схему более компактной. Одна из линий потока входит в контур цикла, другая линия потока, соответствующая окончанию цикла, связывает данный блок с тем блоком, который должен выполняться по окончании цикла. Например, блок-схема на рис.4.1 с использованием данного блока будет иметь вид, приведенный на рис.4.2. Если параметр цикла с каждым шагом увеличивается на единицу, то шаг в блоке цикла, как правило, не указывается.

Пример 1. Составить алгоритм для вычисления функции

$$y = \frac{x^2}{x^2 + cx + d} \text{ на отрезке } a \leq x \leq b \text{ с шагом } \Delta x.$$

Решение. Переменная y вычисляется сначала при $x=a$, затем при $x=a+\Delta x$, затем при $x=a+2\Delta x$ и так до тех пор, пока очередное приращение Δx не приведет к выполнению условия $x > b$. В этом случае вычисления должны быть прекращены. Алгоритм для этой задачи приведен на рис.4.3, на котором блок 1 означает ввод значений постоянных c и d , необходимых для вычисления функции y и границ отрезка a, b . Затем в соответствии с заданием переменной x присваивается значение $x=a$ (блок 2), и вычисляется для этого случая значение y (блок 3). Результат вычисления выводится на печать

(блок 4). После этого переменной x присваивается новое значение $x=a+\Delta x$ (блок 5) и проверяется условие $x \leq b$ (блок 6). Если условие выполняется, то

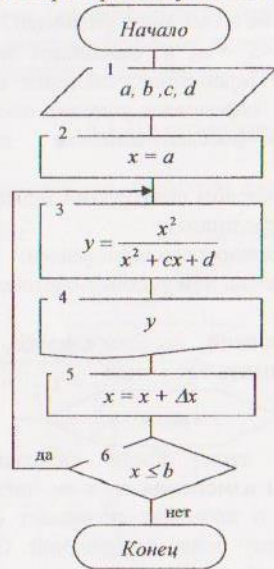


Рис.4.3

Пример 2. Составить алгоритм для вычисления на отрезке $0 \leq x \leq 2\pi$ с шагом $\pi/6$ функции $y=2\sin 0,9x$ и затем для вычисления $u=2,5^y$, если $y \leq 0$, а если $y > 0$, то $u = 1,5^{y+1}$.

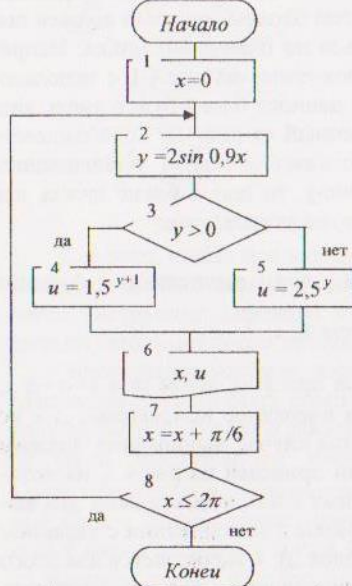


Рис.4.4

в блоке 3 происходит вычисление u для этого значения x , т.е. предыдущий этап вычислений повторяется. При $x > b$ решение должно закончиться.

Возможны случаи, когда циклические вычисления надо производить в зависимости от дополнительных условий. Тогда алгоритм с подобными вычислениями будет иметь циклическую разветвляющуюся структуру (разветвление в цикле).

Решение. Алгоритм приведен на рис.4.4. Так как в данном примере в выражениях отсутствуют неизвестные параметры, то в алгоритме нет блока ввода данных, а после блока начала алгоритма сразу следует блок 1, означающий, что переменной x присваивается начальное значение 0. После этого должно быть сделано вычисление $y = 2 \sin 0,9x$ (блок 2) и в соответствии с заданием проверено условие $y > 0$ (блок 3). Если это условие выполняется, то происходит вычисление $u = 1,5^{y+1}$, в противном случае $u = 2,5^y$ (блоки 4 и 5). После вывода на печать результата вычислений u переменной x присваивается новое значение $x = x + \pi/6$

(блок 7) и проверяется условие $x \leq 2\pi$ (блок 8). Если это условие выполняется, то предыдущий цикл вычислений u и вывода результата на печать повторяется, причем до тех пор, пока x станет больше 2π . В этом случае вычисления должны быть закончены.

Часто возникает необходимость в составлении алгоритма для обработки множества данных, которые рассматриваются как массивы. В этом случае совокупность данных обозначается одной буквой (именем массива, например, a), а элементами массива являются переменные с индексами (например, a_i), где индекс означает порядковый номер элемента этого массива. Например, если имеется ряд чисел: 2; 3.1; 8; 1.4; 0.5 и надо представить этот ряд как массив a , то его элементами являются

$$a_1=2; a_2=3.1; a_3=8; a_4=1.4; a_5=0.5.$$

Пример 3. Составить алгоритм для вычисления массива C , содержащего произведения соответствующих элементов двух массивов A и B , каждый из которых содержит 8 элементов.

Решение. При составлении алгоритма (рис.4.5) необходимо вначале ввести данные (блок 1) - элементы массивов A и B , т.е. $a_1, a_2, \dots, a_8, b_1, b_2, \dots, b_8$. Блок 2 - блок модификации, определяет начало и конец повторных вычислений $C_i = A_i \cdot B_i$ (блок 3) при изменении параметра цикла i от 1 до 8 и вывода для каждого значения i результатов на печать (блок 4). При $i > 8$ вычисления заканчиваются.

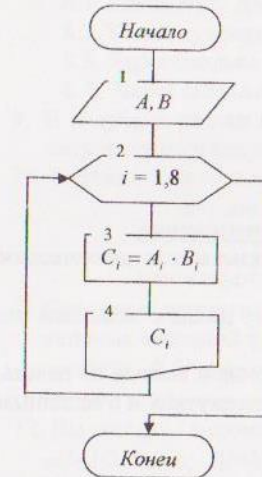


Рис.4.5

Еще один распространенный тип вычислений - проведение некоторых операций с заранее определенными элементами массива.

Пример 4. Составить алгоритм для выбора из массива a_1, a_2, \dots, a_{100} наибольшего числа.

Решение. В алгоритме на рис.4.6 вначале необходимо обеспечить ввод данных (т.е. массива из 100 чисел - блок 1). Затем величине A_{max} присваивается значение первого элемента массива (блок 2). После этого должна рассматриваться следующая величина в массиве, индекс которой на единицу больше, т.е. A_2 (блок 3), и эта величина сравнивается с A_{max} (блок 4). Если оказывается, что $A_2 \leq A_{max}$, то дальше должна рассматриваться величина A_3 , индекс которой увеличился на единицу в блоке 3, и которая сравнивается с A_{max} в блоке 4.

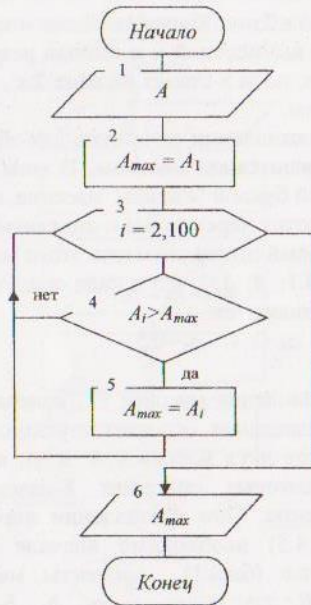


Рис.4.6

Контрольные вопросы и упражнения

1. Дать определение цикла. Какие процессы называются циклическими?
2. Какие типы циклов вы знаете?
3. Какие величины задаются в случае простого цикла с заданным числом повторений?
4. Составить блок-схемы алгоритмов вычисления и вывода на печать значений указанных функций на заданных промежутках и с заданным шагом h .

$$a) f = \frac{\sin x + \cos x}{1 + x^2}; \quad x \in [0, \pi]; \quad h = \frac{\pi}{8}.$$

$$b) f = \begin{cases} x^2 \sqrt{x-a} + e^{-2x^2}, & \text{при } x > a, \\ x \sin ax + ctgx, & \text{при } x = a, \\ e^{-ax} \cos ax + \ln \left| \frac{x+0,1}{x^2} \right|, & \text{при } x < a, \end{cases}$$

$$a = 2,5; \quad x \in [1; 5]; \quad h = 0,3.$$

Если же A_{max} при сравнении с A_2 меньше этой величины, то значение A_2 присваивается переменной A_{max} (блок 5), после чего в блоке 3 вновь происходит увеличение индекса на единицу, в блоке 4 вновь происходит сравнение A_i с A_{max} и цикл вычислений повторяется до $i=100$, после чего в качестве результата выводится значение A_{max} (блок 6) и вычисления заканчиваются.

$$в) y = \begin{cases} (1 - \frac{1}{x})e^{-x}, & \text{при } x \leq 0,5, \\ \frac{xe^x}{(x+1)^2}, & \text{при } x > 0,5. \end{cases}$$

$$x = \sqrt{z} \cdot \arctg \sqrt{z}; \quad z \in [2; 5]; \quad h = 0,2.$$

5. В студенческой группе из 25 человек найти число учащихся, рост которых не ниже 170 см.
6. Найти количество отрицательных чисел из 50 введенных.
7. Из 20 чисел найти среднее арифметическое положительных.
8. Начав тренировки, спортсмен в первый день пробежал 10 км. Каждый следующий день он увеличивал дневную норму на 10 % от нормы предыдущего дня:
 - 8.1. Какой путь пробежит спортсмен в седьмой день?
 - 8.2. Какой суммарный путь пробежит спортсмен за 7 дней?
 - 8.3. Через сколько дней он будет пробегать больше 20 км в день?
 - 8.4. Через сколько дней суммарный путь станет больше 100 км?
9. В сбербанк сделан вклад C_0 рублей из расчета P % годовых. Определить динамику возрастания вклада в течение первых 10 лет хранения.
10. Составить блок-схему печати значений функции

$$y = \frac{\ln^2 | \sin x - 0,3 |}{e}, \quad \text{больших заданного } t,$$

$$\text{где } x \in [0, 4; 2], \quad \Delta x = 0,1.$$

11. Составить алгоритм для нахождения произведения модулей отрицательных значений функции $y = \sin 2x - 0,5$, где $x \in [0, 10]$, $\Delta x = 0,1$, и суммы ее положительных значений.
12. Вычислить значение $y = m!$ (факториал натурального числа m , т. е. $m! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot m$; $0! = 1$).
13. Вычислить сумму четных чисел от 2 до 1000.
14. Разделить натуральное число m на натуральное число n , не используя операцию деления.
15. Найти произведение натуральных чисел m и n , не используя операцию умножения.
16. С клавиатуры последовательно вводятся числа до тех пор, пока не будет введен ноль.

Подсчитать:

- сумму введенных чисел;
- сколько было введено отрицательных и положительных чисел;
- найти среднее арифметическое введенных чисел; затем отдельно положительных и отрицательных чисел;

- каких чисел было введено больше - положительных или отрицательных и насколько;
- найти максимум среди вводимых чисел;
- определить порядковый номер наибольшего числа.

17. Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить сколько клеток образуется через 3, 6, 9, ... 24 часа.

18. На склад привозят однородный груз на машинах различной грузоподъемности. На компьютер, управляющий работой склада, поступает информация о весе груза очередной машины. Составить алгоритм подсчета количества машин, прибывших на склад до его заполнения, если вместимость склада не более 100 тонн.

4.2. Итерационные циклы

Рассмотрим итерационные методы. Эти методы, вообще говоря, не дают возможности найти точное решение за конечное число шагов. С их помощью строится последовательность $\{x^{(k)}\}$ такая, что

$$\lim_{k \rightarrow \infty} x^{(k)} = x,$$

где x - точное решение.

Итерационным называется вычислительный процесс, в котором для определения последующего значения переменной используется ее предыдущее значение (рекуррентная формула). Переход к рекуррентной формуле не только сокращает объем вычислений, но и часто является единственным возможным способом получить решение задачи.

Например:
$$e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!}.$$

Вычисление каждого отдельного члена ряда по прямой формуле $\frac{x^i}{i!}$ может превысить возможности ЭВМ по представлению чисел: $i!$ растет очень быстро, x^i при $x > 1$ также растет очень быстро, поэтому вполне возможно переполнение разрядной сетки. Даже если этого не произойдет, то следует помнить об ограниченности разрядной сетки ЭВМ: при записи в память более чем 20-значных чисел неизбежна потеря цифр. В циклах итерационного типа число повторений вычислений по одним и тем же формулам не задается.

При построении итерационных вычислительных процессов применяется метод последовательных приближений (метод итераций).

Пример 1. Составить блок-схему для вычисления приближенного значения корня уравнения $x = \sqrt{x+1}$ методом последовательных приближений с заданной точностью ε .

Решение. Идея методов последовательных приближений заключается в том, что задается начальное приближение x_0 , которое подставляется в правую часть заданного уравнения; результатом вычислений будет новое значение x_1 ,

которое вновь подставляется в правую часть уравнения, и в результате чего получается очередное значение x_2 и т.д. Действия повторяются до тех пор, пока абсолютная величина разности двух последовательных приближений не окажется меньше заданного числа ε (число ε определяет точность вычислений):

$$|x_{n+1} - x_n| < \varepsilon, \text{ где}$$

n - номер итерации (шага);

x_n - приближенное значение корня уравнения на n -ой итерации (предыдущее приближение);

x_{n+1} - приближенное значение корня уравнения на $(n+1)$ -ой итерации (последующее приближение).

В итерационных циклах количество повторений до завершения вычислительного процесса неизвестно. Здесь циклом управляет заданная погрешность вычислений ε . Если на очередной итерации погрешность больше или равна ε , то цикл продолжается для вычисления очередного приближенного значения результата, иначе происходит выход из цикла.

На рис.4.7 приведена блок-схема для нахождения приближенного значения корня исходного уравнения. Начальное значение приближения корня положено равным нулю (блок 2). Переход к очередному приближению пред-

ставлен инструкцией $x = \sqrt{x+1}$ (тело цикла - блок 3), в которой x в правой части представляет текущее приближение, а x слева - следующее приближение. В качестве условия окончания цикла проверяется неравенство $|\sqrt{x+1} - x| < \varepsilon$ (блок 4). При значении погрешности $|\sqrt{x+1} - x| \geq \varepsilon$

цикл продолжается: в блоке 3 при подстановке в правую часть последнего вычисленного значения x получается очередное, более точное приближение. При выполнении условия $|\sqrt{x+1} - x| < \varepsilon$

значение x может рассматриваться как результат, представляющий корень уравнения, найденный с заданной точностью; при этом осуществляется выход из цикла путем передачи управления от блока 4 к блоку 5. Число выполнений тела цикла зависит не только от начального значения и закона изменения переменной x , но и от требуемой точности результата ε .

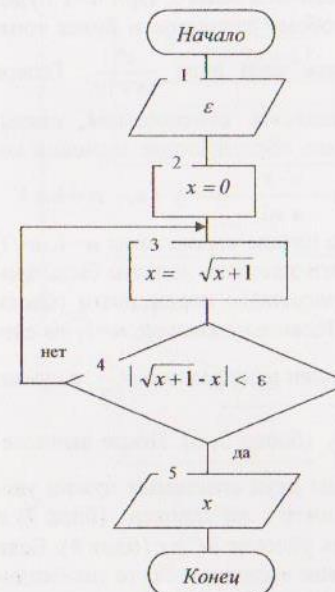


Рис.4.7

При составлении алгоритмов с итерационными циклами нельзя использовать блок модификации, т. к. неизвестно число повторений вычислений.

Типичными задачами, использующими циклы итерационного типа, являются задачи вычисления суммы членов бесконечного ряда, в которых заранее неизвестно, при каком члене ряда будет достигнута требуемая точность и произойдет выход из цикла. Для вычисления суммы членов ряда используется прием накопления суммы. В целях уменьшения затрат времени на вычисление значения каждого очередного члена ряда целесообразно использовать рекуррентную формулу.

Пример 2. Составить алгоритм для вычисления суммы членов бесконечного ряда

$$y = 1 + \frac{x}{2!} + \frac{x^2}{3!} + \frac{x^3}{4!} + \dots$$

при конкретном заданном значении x , причем вычисления должны производиться до тех пор, пока очередной член ряда станет меньше заданного значения погрешности ϵ .

Решение. Блок-схема алгоритма приведена на рис 4.8. В начале алгоритма должны быть введены величины x и ϵ (блок 1). На рис 4.8 приняты обозначения: u - очередной член ряда, n - номер этого слагаемого. При $n=1$ будет $u=1$, а любому значению n будет соответствовать член ряда $\frac{x^n}{(n+1)!}$. Теперь

можно записать соотношения, связывающие все последующие значения величин $u = \frac{u \cdot x}{n+1}$, $y = y + u$, $n = n + 1$.

Так как в начале вычисления $n=1$, $u=1$, $y=1$, то эти значения должны быть присвоены указанным переменным (блоки 2, 3, 4). Поскольку вначале $n=1$, то следующий член ряда равен $u = \frac{x}{2}$, а сумма

$$y = 1 + \frac{x}{2}$$

(блоки 5, 6). После вычисления суммы двух слагаемых нужно увеличить номер n на единицу (блок 7) и проверить условие $|u| \geq \epsilon$ (блок 8). Если это условие выполняется, то необходимо вычислить следующий член ряда и добавить к ранее полученной сумме, т. е. повторить операции в блоках 5,6,7.

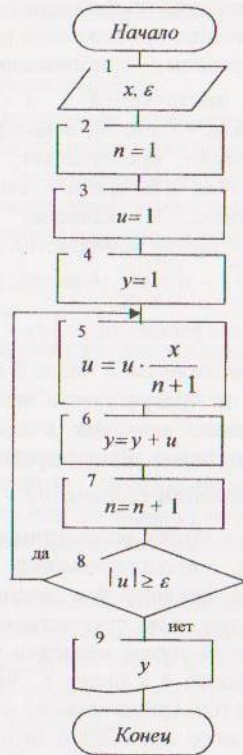


Рис.4.8

Этот цикл вычислений должен повторяться до тех пор, пока u не станет меньше ϵ . После этого значение y должно быть выведено на печать (блок 9), и вычисления завершаются.

Пример 3. Составить блок-схему вычисления $\sin x$ для задаваемых x и ϵ , используя разложение $\sin x$ в ряд Тейлора:

$$\sin x = y = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + (-1)^{n+1} \frac{x^{2n-1}}{(2n-1)!} + \dots$$

с точностью очередного члена ряда $\left| \frac{x^{2n-1}}{(2n-1)!} \right| < \epsilon$, где $n=1, 2, 3, \dots$

Решение. Для сокращения объема вычислений при алгоритмизации подобных вычислительных процессов используется следующая общая методика, которую приведем в применении к рассматриваемой задаче. Для получения рекуррентной формулы запишем выражения для n -ого и $(n+1)$ -ого членов записанного ряда:

$$u_n = (-1)^{n+1} \frac{x^{2n-1}}{(2n-1)!};$$

$$u_{n+1} = (-1)^{n+2} \frac{x^{2n+1}}{(2n+1)!}.$$

Отношение $\frac{u_{n+1}}{u_n} = -\frac{x^2}{2n(2n+1)}$ позволяет вычислить каждый последующий член ряда Тейлора, если известно значение предыдущего члена (рекуррентная формула):

$$u_{n+1} = u_n \cdot \frac{-x^2}{2n(2n+1)}.$$

Алгоритм вычисления $\sin x$ представлен блок-схемой на рис. 4.9. В блоке 2 задаются: начальное значение члена ряда $u=x$, начальная сумма ряда $y=x$ и начальный индекс $n=1$.

В блоке 3 вычисляется очередной член ряда в соответствии с рекуррентной формулой.

В блоке 4 происходит увеличение суммы ряда y на величину следующего члена.

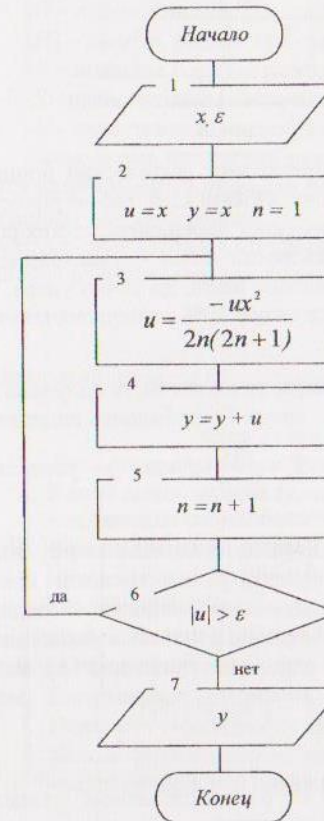


Рис.4.9

В блоке 5 производится увеличение индекса n на 1 для перехода к следующему члену ряда.

В блоке 6 проверяется окончание цикла: если очередной член ряда достигнет заданной точности ε , то осуществляется выход из цикла в блок 7, в противном случае вычисление очередного члена ряда по итерационной формуле продолжается.

Пример 4. Шар катится к краю горизонтальной полки со скоростью V , затем падает с высоты H на пол с твердым покрытием и, многократно ударяясь об пол и отскакивая, продолжает изменять свое положение в горизонтальном направлении (рис.4.10)

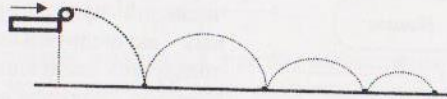


Рис.4.10

Построить блок-схему алгоритма, имитирующего этот физический процесс. Высоты подъема шара изменяются по закону убывающей геометрической прогрессии с заданным знаменателем Q . Требуется определить, пренебрегая изменением горизонтальной составляющей скорости шара V , расстояние по горизонтали от точки, в которой началось падение шара, до точки удара, после которого высота подъема шара составляет менее 10% от первоначальной, и общее число ударов шара к этому моменту.

Решение. Если известна высота H подъема шара, то может быть получено и время T свободного падения:

$$T = \sqrt{\frac{2H}{g}}, \text{ где } g - \text{ ускорение}$$

свободного падения.

Время подъема шара после удара равно времени последующего падения. Следовательно, расстояние по горизонтали, которое шар проходит между ударами, равно

$$2V \cdot \sqrt{\frac{2H}{g}}$$

В рассматриваемом процессе повторяется один и тот же набор событий: удар шара об пол, подъем, падение. В опыте изменяются некоторые физические характеристики: положение шара, его энергия. Суть моделирования процесса

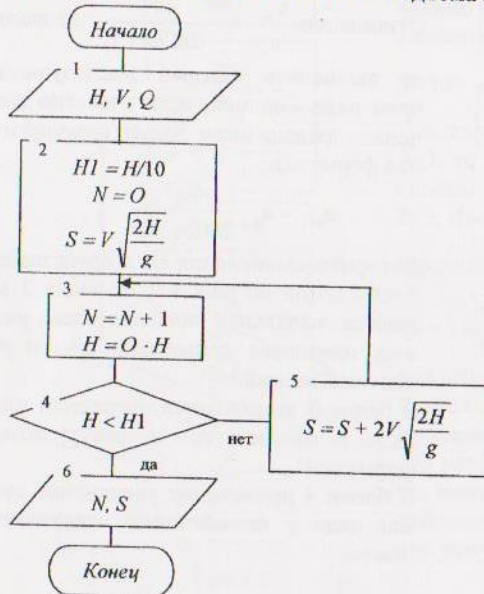


Рис.4.1.1

с помощью алгоритма состоит в том, чтобы представить переменными те характеристики, которые существенны для получения искомым результатов, и поставить в соответствие физическому процессу процесс изменения значений переменных. Повторяющийся набор событий учитывается в алгоритме с помощью цикла, где изменяются значения переменных, которыми представлены изменяющиеся при этих событиях характеристики.

Составим перечень необходимых переменных:

Q - знаменатель геометрической прогрессии, по которой изменяется высота подъема шара. Q зависит от упругих свойств материалов поверхности и шара;

H - изменяющаяся высота подъема шара (амплитудные значения);

$H1$ - высота, равная 10% от первоначальной высоты;

V - заданная горизонтальная составляющая скорости шара;

S - нарастающее расстояние по горизонтали от точки начала падения;

N - нарастающее число ударов шара.

Блок-схема алгоритма представлена на рис.4.11. В алгоритме используется итерационный цикл для накопления значений N и S . В нем многократно используется инструкция $H=Q \cdot H$ (блок 3), в результате чего последовательные значения H образуют убывающую геометрическую прогрессию. Первое падение шара учитывается до цикла в блоке 2

$$S = V \sqrt{\frac{2H}{g}}$$

Цикл оканчивается по выполнению условия $H < H1$.

Контрольные вопросы и упражнения

1. Дать определение итерационного вычислительного процесса.
2. Какой метод используется при алгоритмизации итерационных процессов?
3. Какова идея метода итерации?
4. Какова величина управляет итерационным циклом?
5. Почему при составлении алгоритмов с итерационными циклами нельзя использовать блок модификации?
6. Изложить суть методики построения алгоритма для вычисления сумм бесконечных рядов.
7. Поясните смысл рекуррентной формулы.
8. Каковы преимущества алгоритмов, использующих рекуррентные соотношения, перед алгоритмами прямого накопления суммы бесконечных рядов?
9. Составить блок-схему алгоритма вычисления корня k -й степени уравнения $y = \sqrt[k]{x}$ по итерационной формуле

$$y_{n+1} = \frac{1}{K} \left[(K-1)y_n + \frac{x}{y_n^{K-1}} \right]$$

Погрешность вычислений $|y_{n+1} - y_n| < \varepsilon$; начальное приближение $y_n = y_0$.

10. Используя формулу предыдущего задания, составить блок-схему алгоритма вычисления $y = \sqrt[3]{72}$ с точностью 10^{-4} при $y_0=1$.

11. Составить блок-схему алгоритма нахождения приближенного значения корня уравнения $x = \frac{1}{2 + \sin x}$ по итерационной формуле

$$x_{i+1} = \frac{1}{2 + \sin x_i} \text{ при } x_0 = 0 \text{ с точностью } 10^{-2}.$$

12. Составить блок-схему алгоритма определения наименьшего целого $K > 0$, при котором вычисленное значение функции $y = \frac{x^k}{K}$ становится меньше a . Определить условия, при которых задача имеет решение.

13. Составить блок-схему определения для заданного x значения $\cos x$ по формуле $\cos x = y = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$ с точностью очередного члена ряда $\frac{x^{2n}}{(2n)!} < \varepsilon$.

14. Исходя из представления $\arctg x$ знакопеременным степенным рядом $\arctg x = y = \frac{\pi}{2} - \frac{1}{x} + \frac{1}{3x^3} - \frac{1}{5x^5} + \frac{1}{7x^7} - \dots$ ($x > 1$), построить алгоритм, приближенно вычисляющий значения $\arctg x$ с точностью ε .

15. Вычислить сумму членов ряда

$$z = \frac{1}{m!} + \frac{mx}{(m+1)!} + \frac{m(m-1)}{(m+2)!}x^2 + \frac{m(m-1)(m-2)}{(m+3)!}x^3 + \dots$$

с точностью ε . Для определения текущего значения члена ряда использовать рекуррентную формулу

$$U_{n+1} = U_n \frac{x(m-n+1)}{m+n}, \quad n - \text{номер члена ряда.}$$

16. Получить рекуррентную формулу и вычислить сумму членов ряда

$$y = \frac{1}{2 \cdot 3} + \frac{2}{3 \cdot 4} + \dots + \frac{n}{(n+1)(n+2)} + \dots \text{ с точностью } \varepsilon.$$

17. Начертить блок-схему алгоритма вычисления $f(x)$ с заданной точностью ε :

а) $f(x) = x - \frac{x^3}{3! \cdot 3} + \frac{x^5}{5! \cdot 5} - \frac{x^7}{7! \cdot 7} + \dots;$

б) $f(x) = \sum_{i=1}^{\infty} (-1)^{i-1} \frac{x^i}{i};$

в) $f(x) = \sum_{i=0}^{\infty} (i+1)(-x)^i;$

г) $f(x) = \sum_{i=1}^{\infty} (-1)^i \frac{|\cos ix|}{i};$

д) $f(x) = \cos x + \frac{\cos 2x}{4} + \frac{\cos 3x}{9} + \dots + \frac{\cos nx}{n^2} + \dots;$

е) $f(x) = \frac{\cos 2x}{1 \cdot 3} + \frac{\cos 4x}{3 \cdot 5} + \frac{\cos 6x}{5 \cdot 7} + \dots;$

ж) $f(x) = 2 \left[\frac{x-1}{x+1} + \frac{(x-1)^3}{3(x+1)^3} + \frac{(x-1)^5}{5(x+1)^5} + \dots \right];$

з) $f(x) = x - \frac{x^3}{2!3^2} + \frac{x^5}{4!5^2} - \frac{x^7}{6!7^2} + \dots;$

и) $f(x) = x - \frac{x^3}{1!3} + \dots + (-1)^n \frac{x^{2n+1}}{n!(2n+1)} + \dots$

18. Определить условие сходимости, получить рекуррентную формулу и вычислить сумму членов ряда

$$y = x^n + \frac{1}{2}x^{n-1} + \frac{1}{3}x^{n-3} + \dots + \frac{1}{n}x + \frac{1}{n+1} + \frac{1}{n+2}x^{-1} + \dots$$

с точностью ε .

4.3. Вложенные циклы

Алгоритмы циклической структуры могут быть сложнее рассмотренных выше. Например, часто встречаются циклические вычисления со сложными циклами, т.е. когда цикл содержит в себе один или несколько циклов. Конструкция, в которой цикл содержит внутри себя другие циклы, называется вложенным циклом. Цикл, охватывающий другие циклы, называется внешним, остальные циклы - внутренними. Правила организации как внешнего, так и внутренних циклов такие же, как и для простого. Параметры этих циклов меняются не одновременно, т.е. при одном значении параметра внешнего цикла параметр внутреннего цикла принимает по очереди все значения.

Пример 1. Составить алгоритм для вычисления $z = (x^2 + 1) \arctg \frac{y-1}{x+1}$,

если x изменяется на отрезке $0 \leq x \leq 4$ с шагом 1, а y - на отрезке $2 \leq y \leq 3$ с шагом 0,5. Данное условие означает, что при каждом значении x необходимо перебирать все возможные значения y .

Решение. Из блок-схемы алгоритма (рис.4.12) следует, что вначале в блоках 1, 2 значениям x и y присваиваются начальные значения $x=0$ и $y=2$, затем вычисляется величина z (блок 3), результаты вычислений выводятся на печать (блок 4). Кроме z на печать целесообразно вывести и значения переменных x и y , которым это значение z соответствует.

В блоке 5 величине y присваивается новое значение, равное сумме предыдущего значения и величины шага изменения y . Для полученного значения y

проверяется условие $y \leq 3$ (блок 6), и если это условие выполняется, то вновь вычисляется значение z (блок 3) и печатаются значения x, y, z (блок 4). Затем значение y увеличивается на величину шага, и процесс будет повторяться до тех пор, пока y станет больше 3.

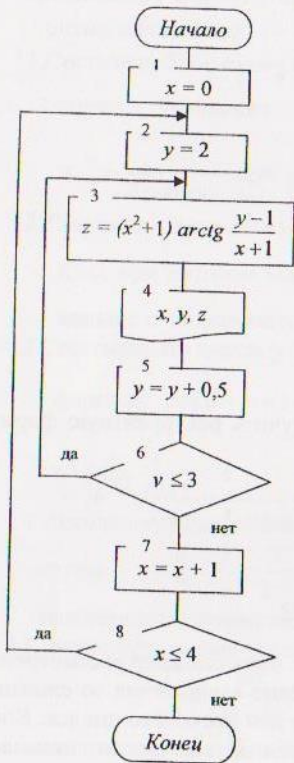


Рис. 4.12

Пример 2. Определить с точностью $\varepsilon = 0,01$ значение аргумента, при котором функция $y = ax - \ln x$ достигнет минимума, при x , изменяющемся от 0,2 до 10.

Решение. Можно было бы решать эту задачу, взяв шаг изменения аргумента равным 0,01. Однако это приведет к увеличению времени счета. Поэтому решение задачи разбивается на два этапа:

- 1) определение грубого значения минимума функции при большом шаге изменения аргумента, например 0,3;
- 2) повторение процесса в районе минимума при шаге изменения аргумента, равном 0,01.

Таким образом, при первом нахождении минимума шаг изменения аргумента h равен 0,3, а его начальное значение $x = 0,2$. При повторном нахождении минимума шаг равен 0,01, а $x_0 = x_{\min} - 0,3$.

Схема алгоритма решения задачи приведена на рис 4.13.

Во внутреннем цикле осуществляется поиск наименьшего значения

функции и значения аргумента, при котором оно достигается. Поскольку функция имеет один минимум, выход из цикла происходит при $y \geq y_{\min}$. После окончания внутреннего цикла проверяется условие $h = 0,01$. Если выполнение условия имеет место, то осуществляется выход из внешнего цикла. В противном случае задаются новое начальное значение переменной x , новый шаг h и внешний цикл повторяется еще один раз.

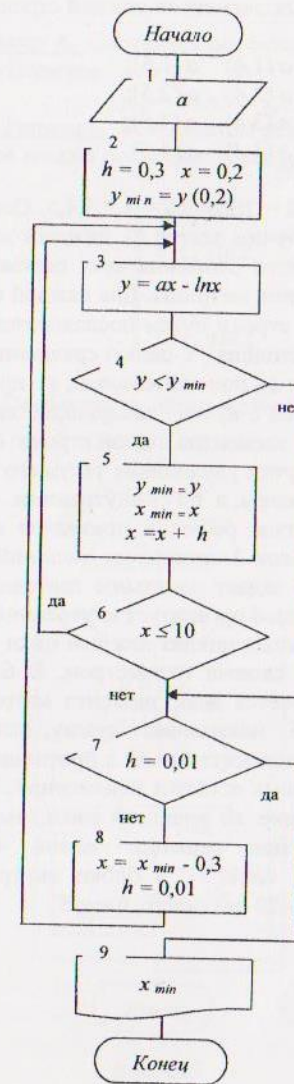


Рис. 4.13

Пример 3. Вычислить суммы положительных элементов каждой строки матрицы $A(4,5)$

$$\begin{matrix} a(1,1) & a(1,2) & a(1,3) & a(1,4) & a(1,5); \\ a(2,1) & a(2,2) & a(2,3) & a(2,4) & a(2,5); \\ a(3,1) & a(3,2) & a(3,3) & a(3,4) & a(3,5); \\ a(4,1) & a(4,2) & a(4,3) & a(4,4) & a(4,5). \end{matrix}$$

Элементы матрицы обозначим a_{kl} , где $k = 1,2,3,4$; $l = 1,2,3,4,5$. Согласно условию задачи в результате ее решения получим вектор C_k , каждый элемент которого получается путем последовательного сложения всех положительных элементов фиксированной строки k данной матрицы. Для каждой строки k необходимо перебрать все элементы этой строки путем последовательного изменения l от 1 до 5 с шагом 1 (l - номер столбца) с целью сравнения каждого элемента k -й строки с 0; если элемент a_{kl} положительный, то производится накопление суммы, иначе сравнивается с нулем следующий элемент. После получения суммы положительных элементов одной строки осуществляется переход к следующей строке путем увеличения текущего значения k на единицу. Цикл по k является внешним, а по l - внутренним. Схема

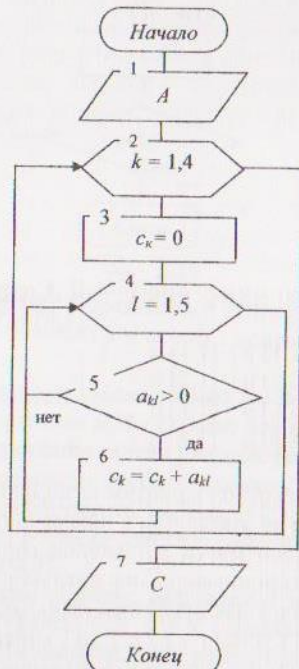


Рис.4.14

алгоритма решения приведена на рис. 4.14. Блок 2 организует внешний цикл, блок 3 задает начальное значение суммы, блок 4 организует внутренний цикл. В сложных циклах каждый цикл управляется своими параметром. В блоке 5 проверяется знак элемента матрицы, а блок 6 накапливает сумму, если элемент положительный, в противном случае сумма остается неизменной. Блоки, входящие во внешний цикл, выполняются при решении задачи 4 раза (напр., блок 7), а блоки внутреннего цикла - 20 раз (напр., блок 5).

Пример 4. Составить блок-схему алгоритма вычисления значений функции y по формуле

$$y_{ij} = \ln z_i \sin x_j, \text{ где } i = 1, 2, \dots, 10; \quad j = 1, 2, \dots, 15.$$

Решение. Результатом решения этой задачи будет множество чисел, которое можно представить следующей матрицей произведений $\ln z_i \sin x_j$

$$\begin{pmatrix} \ln z_1 \sin x_1 & \ln z_1 \sin x_2 & \dots & \ln z_1 \sin x_{15} \\ \ln z_2 \sin x_1 & \ln z_2 \sin x_2 & \dots & \ln z_2 \sin x_{15} \\ \dots & \dots & \dots & \dots \\ \ln z_{10} \sin x_1 & \ln z_{10} \sin x_2 & \dots & \ln z_{10} \sin x_{15} \end{pmatrix}$$

Особенность этого примера состоит в том, что при организации алгоритма с вложенным циклом во внешнем цикле по i одновременно с изменением

величины z_i следует вычислять значение промежуточной переменной $a = \ln z_i$ для того, чтобы при каждом значении i вычисление $\ln z_i$ производилось только один раз для всех x_j . Блок-схема алгоритма с вычислением значений промежуточной переменной a во внешнем цикле приведена на рис.4.15.

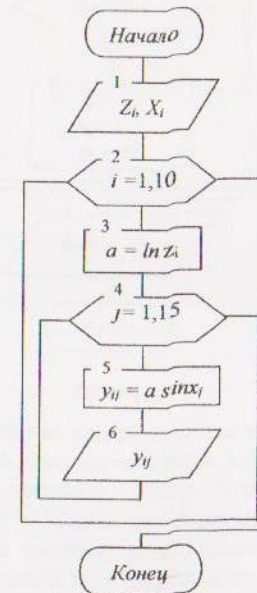


Рис.4.15

Пример 5. Составить блок-схему алгоритма вычисления функции z по формуле

$$z_{ij} = \sqrt{\cos x_i} \cdot \sin y_j, \text{ где } i = 1, 2, \dots, n; \quad j = 1, 2, \dots, m.$$

Так как на значения x_i не накладываются никакие ограничения, то $\cos x_i$ может быть и отрицательным числом, что исключает автоматическое повторение вычислений в цикле по i из-за того, что не будет существовать

действительного значения квадратного корня из $\cos x_i$. Следовательно, в цикле алгоритма по параметру i необходимо проверить выполнение условия $\cos x_i \geq 0$. Если это условие не выполняется, то согласно алгоритму, представленному на рис.4.16, выводится на печать (блок 8) сообщение о том, что для

данного i значение $\sqrt{\cos x_i}$ вычислить невозможно и осуществляется переход к вычислениям с новым значением i .

В блок-схеме этого алгоритма реализованы линейная, разветвляющаяся и циклические структуры.

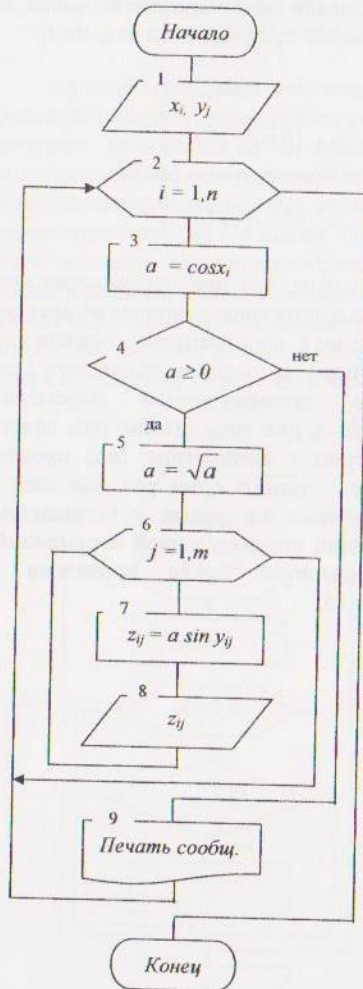


Рис.4.16

Пример 6. Составить блок-схему алгоритма упорядочения элементов вектора (одномерного массива) $B(b_1, b_2, \dots, b_n)$ в порядке возрастания их значений.

Решение. Последовательность действий, задаваемая алгоритмом, блок-схема которого представлена на рис. 4.17, заключается в следующем. Попарно сравниваются между собой все соседние элементы вектора B . Если $b_i > b_{i+1}$, то их необходимо поменять местами, иначе элементы b_i и b_{i+1} остаются на своих местах, затем сравнивается очередная пара элементов (т. е. $b_{i+1} > b_{i+2}$) и т. д. После первого просмотра всех элементов на позицию n будет поставлен наибольший элемент. При втором просмотре с выполнением указанных действий второй по величине элемент будет перемещен на позицию $n-1$ и т.д. После $(n-1)$ -го просмотра все элементы вектора B будут упорядочены в порядке их возрастания. В процессе попарного сравнения элементов целесообразно исключить из сравнения те упорядоченные элементы, которые уже перемещены на соответствующие позиции согласно их значениям. В соответствии с этим параметр внешнего цикла блок-схемы рис. 4.17 j ограничивается величиной $n-1$ (количество просмотров), а параметр внутреннего цикла ограничивается величиной $n-j$.

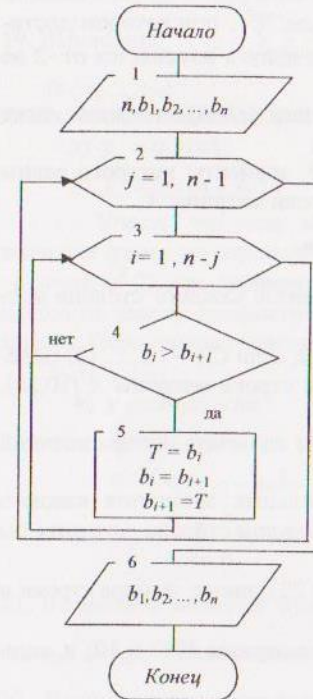


Рис.4.17

В блоке 4 производится попарное сравнение смежных элементов. Если условие $b_i > b_{i+1}$ выполняется, то в блоке 5 осуществляется обмен местами i -го и $i+1$ -го элемента. Здесь T - промежуточная переменная, необходимая для обмена b_i и b_{i+1} . В процессе каждого просмотра (т.е. изменения параметра внешнего цикла j) восстанавливается начальное значение параметра внутреннего цикла по i (переход от блока 3 к блоку 2), что позволяет сравнивать вновь попарные элементы, начиная с первого.

Контрольные вопросы и упражнения

1. Дать определение сложного цикла.
2. Какой цикл называется внешним (внутренним)?

3. Вычислить наибольшие значения функции $y_i = 2e^{b_i x - 5x^2}$, если b_i заданно массивом $(b_1, b_2, \dots, b_{20})$. Аргумент x изменяется от -2 до 2 с шагом $0,1$. Все наибольшие значения запомнить в массиве C .
4. Найти значение аргумента x для функции $y = ae^{bx+cx^2}$, при котором достигается максимум, с точностью до $\varepsilon = 0,005$. Значение x изменяется от -2 до 2 с шагом $0,2$. Функция имеет один максимум.
5. Вычислить сумму квадратов элементов матрицы $A(M, N)$, лежащих ниже (выше) главной диагонали.
6. Дана матрица $A(m, n)$. Получить вектор $C(m)$, элементы которого равны сумме квадратов элементов соответствующей строки матрицы A :

$$C_i = \sum_{j=1}^n a_{ij}^2, \quad i = 1, 2, \dots, m.$$

7. Определить количество положительных элементов каждого столбца матрицы $A(10 \times 20)$ и запомнить их в массиве M .
8. Вычислить компоненты вектора $C(c_1, c_2, \dots, c_{10})$, если $C_i (i=1, 2, \dots, 10)$ определяются как сумма элементов соответствующей строки матрицы $A(10, 20)$, стоящих на четных позициях.
9. Дана матрица $B(10, 20)$. Определить и вывести на печать номера позиций всех нулевых элементов заданной матрицы.
10. Найти среднее арифметическое положительных элементов каждого столбца матрицы $X(15 \times 25)$ при условии, что в каждом столбце есть хотя бы один положительный элемент.
11. Найти наименьший элемент матрицы $A(15 \times 25)$, также номера строки и столбца, в которых он находится.
12. Найти наибольшие элементы каждой строки матрицы $X(10 \times 10)$ и записать их в массив Y .
13. Вычислить суммы элементов каждой строки матрицы $X(20 \times 20)$, определить наименьшее значение этих сумм и номер соответствующей строки.
14. Найти минимальные элементы каждой строки матрицы $X(20 \times 20)$ и поместить их на главную диагональ, а диагональные элементы записать на место минимальных.
15. Определить в матрице $A(M, N)$ наименьший из наибольших элементов каждой строки и его координаты (номер строки и столбца).
16. Дана матрица $A(10 \times 15)$. Проверить знак произведения всех элементов каждой строки и вывести эти произведения на печать; прекратить данный процесс при выявлении отрицательного знака у произведения элементов.

17. Вычислить $z_i = \prod_{j=1}^{20} \frac{i+j}{x_j}$, где значения x_j заданы массивом

$(x_1, x_2, \dots, x_{40})$. Результаты запомнить в массиве z . При решении использовать приемы: во внутреннем цикле - накопление произведения, во внешнем цикле - запоминание результатов.

18. Вычислить значения функции $z = \sum_{i=1}^{15} \prod_{k=1}^{10} \frac{a_i + b_k}{2}$, где значения a_i заданным массивом $(a_1, a_2, \dots, a_{15})$, а значения b_k изменяются от 1 с шагом $0,1$.

19. Вычислить $z = \sum_{i=1}^{20} \sum_{k=1}^n \frac{x_i^k}{K}$, где значения x_i заданы массивом $(x_1, x_2, \dots, x_{20})$.

20. Вычислить значения функции y по формулам

$$a) y_j = a_i \sin x_j, \quad i = 1, 2, \dots, M; \quad j = 1, 2, \dots, N;$$

$$б) y_j = \lg \cos a_i \sin x_j, \quad i = 1, 2, \dots, 10, \quad j = 1, 2, \dots, 10.$$

Учесть, что $\cos a_i$ может быть и не положительным числом, тогда значение $\lg \cos a_i$ не определено.

- в) $y = xz/(x^3 - 2)$, где переменная x принимает 20 различных значений и изменяется по закону арифметической прогрессии $(x_{i+1} = x_i + \Delta x)$; значения x_0 и Δx заданы. Переменная z принимает 30 различных значений:

$$z_1, z_2, \dots, z_{30}.$$

- г) $y = z \ln |x|$, где

$$z = (z_1, z_2, \dots, z_8), \quad z_{i+1} = z_i + \Delta z; \quad z_1 \text{ и } \Delta z - \text{заданы};$$

$$x = (x_1, x_2, \dots, x_{10}), \quad x_{i+1} = x_i + \Delta x; \quad x_1 \text{ и } \Delta x - \text{заданы}.$$

$$д) y_j = \frac{1}{c_i} \sum_{k=1}^6 \frac{x_j^k}{K!}, \quad i = 1, 2, \dots, 10; \quad j = 1, 2, \dots, 10.$$

21. Вычислить значение функции $z = \sum_{i=1}^{20} a_i \prod_{k=1}^{10} \frac{a_i + b}{2}$, где a_i заданы массивом $(a_1, a_2, \dots, a_{20})$, b изменяется от 0 с шагом $0,1$.

22. Вычислить значение функции $z = \frac{a_i + b_j}{c_k}$; a_i, b_j, c_k заданы массивами из

$10, 8$ и 5 элементов соответственно.

23. Найти 3 наибольших элемента массива $(a_1, a_2, \dots, a_{30})$.

24. Упорядочить элементы массива $(x_1, x_2, \dots, x_{60})$, расположив их по убыванию в массиве Y .

25. Упорядочить в порядке убывания элементы каждой строки матрицы $A(M, N)$.

**БИБЛИОГРАФИЧЕСКИЙ СПИСОК РЕКОМЕНДУЕМОЙ
ЛИТЕРАТУРЫ**

1. Информатика: учеб. /Под ред. Н.Б. Макаровой. – Финансы и статистика. – 2003. – 768 с.
2. Могилев А.В., Пак Н.И., Хеннер Е.К. Информатика. – М.: Academia. – 2001. – 816 с.
3. Информатика. Базовый курс. /Симонович В.С. и др. - СПб.: Изд-во «Питер», 2000. – 640 с.
4. Острейковский В.А. Информатика: учеб. для вузов. – М.: Высш. ш к., 2000. – 511 с.
5. Лапчик М.П., Семакин И.Г., Хеннер Е.К. Методика преподавания информатики /Под. ред. М.П. Ланчика. - М.: Academia. – 2002. – 580 с.

ОГЛАВЛЕНИЕ

Введение.....	3
I. Основы алгоритмизации.....	3
II. Линейный вычислительный процесс.....	5
III. Разветвляющийся вычислительный процесс.....	10
IV. Циклический вычислительный процесс.....	17
4.1. Простые числа с заданным числом повторений	18
4.2. Итерационные циклы.....	24
4.3. Вложенные циклы.....	31
Библиографический список рекомендуемой литературы.....	40

**Основы алгоритмизации вычислительных
процессов**

Методические указания по курсу «Информатика»
для студентов 1-го курса всех специальностей

Составители: к. т. н., проф. Виктор Петрович Авдеев,
к. ф.-м. н., доц. Галина Тихоновна Венгерова,
к. т. н., доц. Владимир Исламович Гильмутдинов,
к. ф.-м. н., доц. Александр Давыдович Кононов,
к. т. н., доц. Андрей Александрович Кононов

Редактор Лантюхова Н.Н.

Подписано в печать 21.06.2005. Формат 60×84 1/16. Уч.-изд. л. 2,5.
Усл.-печ. л. 2,6. Бумага писчая. Тираж 400 экз. Заказ № 318

Отпечатано: отдел оперативной полиграфии
Воронежского государственного архитектурно-строительного университета
394006, Воронеж, ул. 20-летия Октября, 84