

**МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА
И ПРОДОВОЛЬСТВИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**Учреждение образования
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ
АГРАРНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Е.В. Севернева, Н.М. Жалобкевич

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

*Рекомендовано Учебно-методическим объединением высших учебных заведений Республики Беларусь по образованию в области сельского хозяйства в качестве учебно-методического комплекса для студентов высших учебных заведений группы специальностей
74 06 Агроинженерия*

МИНСК 2009

УДК
ББК
С

Рецензенты:

д-р физ.-мат. наук, проф. каф. ВМиП БГУИР *А.В. Аксенчик*;
канд. техн. наук, проф. БГАТУ *Р.И. Фурунжиев*

Севернева, Е.В.

С Основы алгоритмизации и программирования : учеб.-метод. комплекс / Е.В. Севернева, Н.М. Жалобкевич. – Минск : БГАТУ. – 114 с.

ISBN

Издание содержит методические материалы для самостоятельного изучения дисциплины, включая наименование и содержание тем лекций, перечень изучаемых и контрольных вопросов для самостоятельной работы студентов, вопросы для подготовки к экзамену и рекомендуемую литературу по всем темам, вопросы для обратной связи и самоконтроля, а также варианты индивидуальных заданий.

Предназначено для студентов агроинженерных специальностей, а также может быть использовано преподавателями и учащимися аграрно-технических колледжей.

УДК
ББК

ISBN

© БГАТУ, 2009

ВВЕДЕНИЕ

Издание «Основы алгоритмизации и программирования» является разделом учебной дисциплины «Информатика», в результате изучения которого студент должен:

- знать типовые блоки и типовые схемы алгоритмизации задач;
- применять типовые структуры алгоритмов для решения задач;
- уметь составлять схемы алгоритмов и программы на алгоритмическом языке Turbo-Pascal для решения практических задач;
- анализировать возможность применения типовой структуры алгоритма к решению конкретной задачи;
- знать сущность математической формализации численных методов решения задач;
- уметь решать задачи численными методами с применением алгоритмов и программирования на алгоритмическом языке Turbo-Pascal.

В предлагаемом издании содержание учебного материала представлено учебно-методическим комплексом, в котором сделан упор на управляемую самостоятельную работу студентов. С этой целью комплекс разбит на модули, наиболее адекватно отражающие логику и содержание курса:

- модуль 1– Основы программирования. Математическая формализация текстовых задач, составление алгоритмов их решения. Алгоритмический язык Turbo-Pascal. Алгоритмы линейной и разветвляющейся структуры и их реализация на языке Turbo-Pascal;
- модуль 2 – Алгоритмы циклической структуры и их реализация на языке Turbo-Pascal;
- модуль 3– Работа с двумерными массивами (матрицами);
- модуль 4 – Численные методы.

Каждый модуль содержит методические материалы для самостоятельного изучения дисциплины, включая наименование и содержание тем лекций, перечень изучаемых и контрольных вопросов для самостоятельной работы студентов, вопросы для подготовки к экзамену и рекомендуемую литературу по всем темам, вопросы для обратной связи и самоконтроля, а также варианты индивидуальных заданий. Каждая изучаемая тема в УМК пояснена примерами решения конкретных задач в виде схем алгоритмов и программами на алгоритмическом языке Turbo-Pascal.

МОДУЛЬ № 1

«Основы программирования. Математическая формализация текстовых задач. Составление алгоритмов их решения. Алгоритмический язык Turbo-Pascal. Алгоритмы линейной и разветвляющейся структуры и их реализация на языке Turbo-Pascal»

В результате изучения модуля студент должен:

- знать типовые блоки и типовые схемы алгоритмизации задач;
- применять структуры линейных и разветвляющихся алгоритмов для решения задач;
- анализировать возможность применения изучаемых типовых структур алгоритмов для решения практико-ориентированных задач;
- уметь математически формализовать поставленную задачу, находить метод решения ее, составлять схемы алгоритмов и программировать их на языке Turbo-Pascal;
- приобретать навыки самостоятельной работы с учебной литературой и другими источниками информации.

НАУЧНО-ТЕОРЕТИЧЕСКОЕ СОДЕРЖАНИЕ МОДУЛЯ

СЛОВАРЬ ОСНОВНЫХ ПОНЯТИЙ

Новые понятия

Новое понятие	Определение
Алгоритм	Под алгоритмом понимается «точное предписание, определяющее вычислительный процесс, ведущий от варьируемых начальных данных к искомому результату»
Свойства алгоритма:	
Определенность	Указания, составляющие алгоритм, должны быть четкими, понятными, однозначными, не допускающими двоякого толкования
Дискретность	Возможность разбиения алгоритмического процесса на отдельные этапы-шаги
Конечность	Число шагов алгоритмического процесса должно быть конечным

Результативность	Определяемый алгоритмом процесс через конечное число шагов обязательно приведет к результату, который является решением задачи
Массовость	Алгоритм должен быть применим к решению множества однотипных задач
Алгоритмы линейной структуры	Алгоритмы, в которых операции выполняются последовательно одна за другой, в естественном и единственном порядке.
Алгоритмы разветвляющейся структуры	Алгоритмы, вычислительные процессы в которых в зависимости от выполнения некоторого логического условия производятся по одному из нескольких заранее определенных направлений

Понятия для повторения

Понятие для повторения	Определение
Константы	Величины, значения которых постоянны и не изменяются при выполнении программы
Переменные	Именованные величины, которые могут изменять свое значение в процессе выполнения программы

ОСНОВНОЙ ТЕКСТ

Лекция 1

Понятие, свойства и способы описания алгоритма

План лекции:

1. Понятие и свойства алгоритма.
2. Способы описания алгоритма.

Под алгоритмом понимается «точное предписание, определяющее вычислительный процесс, ведущий от варьируемых начальных данных к искомому результату». Алгоритм включает систему правил, определяющих содержание и конечную последовательность действий (шагов и операций), выполняемых над некоторыми объектами с целью переработки исходных и промежуточных данных в

искомый результат. Таким образом, это предписание конкретному исполнителю о том, какие действия, над какими объектами и в каком порядке следует выполнять для решения поставленной задачи.

При разработке алгоритмов следует учитывать ряд требований, совокупность которых формирует его свойства: определенность, дискретность, конечность, результативность, массовость.

Указания, составляющие алгоритм, должны быть четкими и однозначными, не допускать произвольного или двойного толкования. Это свойство называют **определенностью**. Вычислительный процесс после выполнения заданной алгоритмом конечной последовательности действий должен заканчиваться выдачей результатов или сообщением о невозможности решить задачу. Эти взаимосвязанные свойства алгоритма называются **конечностью** и **результативностью**.

Для иллюстрации этих свойств алгоритма можно рассмотреть вычисление значения функции $\sin(x)$ методом разложения ее в ряд по степеням. Очевидно, что при расчете по подобной формуле окончательного ответа не будет, поскольку в условии задачи ничего не говорится о количестве членов ряда, которые необходимо учитывать при вычислениях. Без подобных указаний вычислительный процесс может продолжаться бесконечно. Чтобы этого не произошло, следует ввести некоторые ограничения, обеспечивающие свойство **конечности** алгоритма, в частности задать некоторое допустимое число шагов выполнения алгоритма. Например, суммирование продолжать до тех пор, пока значение очередного, учитываемого в сумме члена ряда не станет меньше некоторого ε , равного 10^{-10} . В этом случае за некоторое конечное число шагов будет получен результат, и алгоритм вычисления функции приобретет свойство **результативности**.

Алгоритмы должны обладать свойством **массовости**, чтобы их можно было использовать для решения множества однотипных задач с различными исходными данными. Например, алгоритм Евклида позволяет найти НОД для любой пары натуральных чисел.

Разработанные алгоритмы могут быть представлены на физическом носителе информации различными способами, наиболее известными из которых являются: словесный (средствами языка человеческого общения с тщательно отобранным набором слов и фраз), структурно-стилизованый (языком псевдокодов), графический (схемами из графических блок-символов) или программный (текстами программ).

Наиболее распространенным способом представления алгоритма является *графический*. В графическом представлении алгоритмы изображаются в виде блок-схемы, дополненной элементами словесной или математической записи. Схема алгоритма включает геометрические фигуры (блочные символы), соединенные между собой стрелками (линиями), указывающими порядок выполнения операций. Блочные символы стандартизированы и различаются по типу выполняемых действий.

В схеме начало и завершение алгоритма, а также вход и выход из вспомогательных алгоритмов отмечаются соответственно блочными символами «начало» и «конец» (рисунок 1а, б (блоки 1 и 2)). Эти блоки, в отличие от большинства других, используются по одному в алгоритме и отмечают начало и конец пути обработки информации. Каждая схема обязательно должна начинаться и заканчиваться этими символами.

Изображенные на рисунке 1в, г блочные символы в виде параллелограмма (блоки 3 и 4) используют для обозначения операций ввода-вывода данных.

Блок, отражающий вычислительный процесс, применяют для обозначения одной или группы операций, изменяющих значение, форму представления или размещения данных (рисунок 1д (блок 5)). Производимые операции в этом блоке записывают в любой форме с использованием математических формул, выражений и пояснений на естественном языке.

Логический блочный символ «решение» (рисунок 1 е–и (блоки 6, 7, 8, 9 соответственно)) используют для обозначения выбора направления выполнения алгоритма в зависимости от некоторых условий. В блоке указывают условие, вопрос или решение, определяющие дальнейшее направление выполнения алгоритма. Условия могут быть простыми (рисунок 1е (блок 6)) и составными (рисунок 1з (блок 8)). В них должны быть учтены абсолютно все возможные варианты следования процесса при решении задачи.

Из блока проверки условия может выходить два, три и более (блок 9) информационных потока, что отличает его от других блочных символов, имеющих не более одного выхода. Выходящие из блока линии должны снабжаться пояснениями о направлениях исполнения алгоритма при выполнении или невыполнении приведенного условия (например, «да», «нет», «<0», «=0», «>0», «=1», «+», «□–» или др.).

Блочный символ модификации (рисунок 1к, л (блоки 10 и 11)) символизирует начало циклических вычислений (заголовок цикла),

для управления которыми он используется. Внутри блока указывается переменная цикла и параметры, характеризующие закон ее изменения, например,

$$I = A_{\text{НАЧ}}, A_{\text{КОН}}, \Delta A,$$

где I – переменная цикла,

$A_{\text{НАЧ}}$ и $A_{\text{КОН}}$ – начальное и конечное значения переменной цикла,
 ΔA – шаг ее изменения (переменная цикла изменяется от $A_{\text{НАЧ}}$ до $A_{\text{КОН}}$ с шагом ΔA).

Если шаг равен 1, то ΔA можно не указывать. Кроме входящей линии, блок модификации имеет одну выходящую (на рисунке 1л обозначенную «Вых»), а также линии, отмечающие передачу вычислительного процесса на обработку для циклических вычислений «Цикл» и возврат в начало для изменения переменной цикла «Изм. пер».

Для обращения к вычислению по подпрограмме (стандартной или разработанной пользователем) в схеме используют блок-символ «предопределенный процесс» (рисунке 1м (блок 12)). Он как бы заменяет алгоритм подпрограммы (вспомогательный алгоритм) и указывает, что информационный поток передается подпрограмме. По завершении вычислительного процесса в подпрограмме результаты расчета возвращаются в основной алгоритм, в котором процесс вычислений возобновляется с блока, следующего за блоком обращения к подпрограмме. Блок «предопределенный процесс» используют при организации вспомогательных алгоритмов, оформленных автономно в виде отдельного модуля, или при обращении к библиотечным подпрограммам.

Схема является самым наглядным и простым способом представления алгоритма. В ней четко прослеживаются порядок выполнения действий, потоки информации и пути их следования, которые отмечаются линиями со стрелками (стрелки допускается опускать, если потоки направлены сверху вниз и слева направо). Линии по отношению к блокам могут быть входящими и выходящими. Количество входящих линий для всех блоков не ограничено – их может быть одна, две, три и более. Выходящая же линия для большинства блоков может быть только одна (исключение – блоки проверки условия). В схеме блоки, за исключением соединителей, могут нумероваться для простоты дальнейшего описания их работы, организации комментариев и использования соединителей. Номера проставляют в верхней части графического символа в разрыве его начертания, как это сделано на рисунке 1.

Внутри блоков и рядом с ними делаются записи и обозначения, уточняющие выполняемые функции. Эти записи могут производиться в любой удобной для разработчика форме. Они не имеют каких-либо существенных ограничений (на язык, обозначения, символы и др.), однако, должны быть понятны всем, кто будет пользоваться алгоритмом. Единственное ограничение накладывается на последовательность записей – они должны читаться (использоваться при работе алгоритма) слева направо и сверху вниз независимо от направления потоков информации.

Алгоритмы целесообразно разрабатывать поэтапно (по шагам). Сложные задачи следует разбивать на достаточно простые, легко воспринимаемые части.

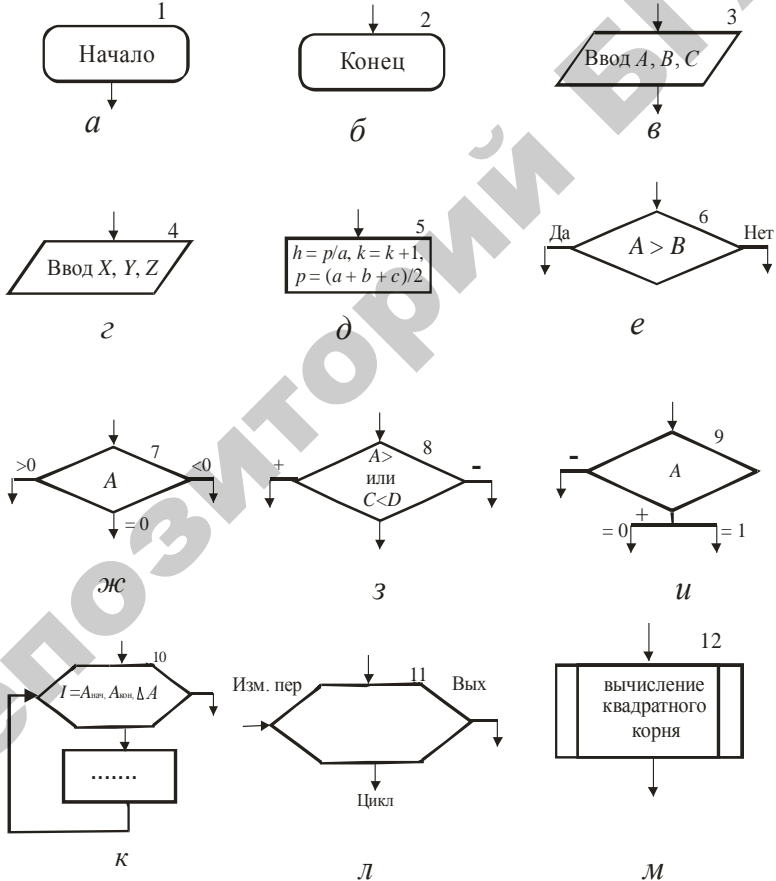


Рисунок 1 – Наиболее употребляемые в схемах алгоритмов блок-символы

Логика алгоритма должна опираться на минимальное число достаточно простых управляющих базовых структур. При разработке схем алгоритмов необходимо соблюдать некоторые требования:

- в схеме алгоритма все линии от блока «начало» до блока «конец» не должны иметь разрывов, не помеченных соединителями. Все линии, указывающие последовательность выполнения действий, должны быть замкнутыми;
- в схеме должны четко прослеживаться потоки информации. Блоки следует размещать таким образом, чтобы избежать пересечения линий. При передаче управления в схеме «снизу-вверх» или «справаналево» линии обязательно помечают стрелками;
- не допускается передача управления в никуда. «Источник» передачи управления и «получатель» должны быть четко обозначены.

Лекция 2

Основы программирования на языке Turbo-Pascal (TP)

План лекции:

1. Классификация данных.
2. Оператор присваивания.
3. Выражения и стандартные функции языка TP.
4. Операторы ввода данных.
5. Операторы вывода данных.
6. Структура программы.

Алгоритмический язык Turbo-Pascal (TP) для ПК разработан фирмой Borland и широко используется как для обучения основам программирования, так и для создания программных продуктов профессионального назначения.

Классификация данных

ПК выполняет действия с данными: читает данные из памяти, производит с ними нужные действия и результаты вновь заносит в память. Какие действия и с какими данными выполнять, куда записывать результат – указывается в программе. Данные могут представлять собой числа, буквы (символы), текст – в зависимости от решаемой задачи.

Одно и то же число на TP можно записать различными способами (таблица 1).

Таблица 1 – Запись чисел на ТР различными способами

Математическая запись	Целое	Запись на ТР	
		Действительное (вещественное)	
		В форме с фиксированной точкой	В форме с плавающей точкой
25	25	25.0	2.5E01
-892	-892	-892.0	8.92E02
0	0	0.0	0.0E00
-4,3	-	-4.3	-4.3E00
$-3,56 \times 10^8$	-	-356000000.0	-3,56E08
$27,84 \times 10^{-6}$	-	0.00002784	27.84E-06

Все данные в ТР относятся к определенному типу. Тип данных показывает, какие значения принимают данные, какие операции можно с ними выполнять и как данные представляются в памяти ПК. Типы данных указываются в специальном разделе программы, который называется **раздел объявлений (описаний)**.

В программе обычно используется много данных. Чтобы указать, с какими данными требуется выполнить действия, данным присваиваются **имена**. Имя может содержать буквы и цифры, не должно разделяться пробелами и должно начинаться буквенными символами, например **primer1**.

Иногда в программах на ТР используются данные, которые получают значение в начале программы и не изменяют своего значения в процессе ее выполнения. Такие данные называются **константами**. Константы указываются в разделе объявлений после слова **CONST**, например

CONST

```

a= 50;           {целая константа}
DELTA = 1.7E-8; {действительная константа}
FLAG = TRUE;    {логическая константа}
SUMBOL = 'h';   {символьная константа}
STROKA = 'PASCAL'; {строковая константа}
    
```

Константу $\pi = 3.14159$ указывать в разделе объявлений не нужно, в программе в качестве ее используется имя Pi.

Данные, значения которых могут изменяться при выполнении программы, называются **переменными**. Типы всех переменных, используемых в программе, указываются в разделе объявлений **VAR**, после имени переменной и двоеточия, например:

VAR

```

INDEX:INTEGER;   {переменная целого типа}
SUM,MIM:REAL;    {переменные действительного типа}
    
```

PRIZN:BOOLEAN; {переменная логического типа}
SYM:CHAR; {переменная символьного типа}
ST1,ST2:STRING; {переменные строкового типа}

Эти типы данных чаще всего используются в ТР.

Все данные, с которыми работает программа, хранятся в соответствующих полях памяти ПК. Эти поля выделяются при трансляции программы в соответствии с типом данных и закрепляются за данными в течение всего времени выполнения программы. Так происходит при статическом распределении памяти (динамическое выделение памяти здесь не рассматривается). Доступ к содержимому поля памяти осуществляется с помощью имени соответствующего данного.

Оператор присваивания

Для изменения содержимого переменной используется *оператор присваивания*. Оператор содержит операцию присваивания, которая обозначается двумя символами := (двоеточие и равенство). Слева от операции присваивания указывается имя той переменной, которой нужно присвоить новое значение, справа – значение, которое следует присвоить указанной переменной. Один оператор отделяется от другого с помощью символа ; (точка с запятой), например **INDEX:= 5; MIN:= MAX;**

Выражения и стандартные функции языка ТР

Действия, которые должен выполнить ПК, указываются в выражениях. **Выражение** задает порядок выполнения действий над данными и состоит из операндов, круглых скобок и знаков операций. **Операнды** – это константы, переменные и функции языка ТР. Круглые скобки используются для изменения последовательности выполнения операций в выражении. **Операции** – это действия, выполняемые над операндами. В арифметических выражениях используются арифметические операции, представленные в таблице 2.

Таблица 2 – Используемые в ТР арифметические операции

Операция	Действие	Тип операндов	Тип результата
+	Сложение	Целый действительный	Целый действительный
-	Вычитание	-“-	-“-
*	Умножение	-“-	-“-
/	Деление	-“-	-“-
Div	Целочисленное деление	Целый	Целый
Mod	Остаток от деления	-“-	-“-

Результат вычисления выражения заносится в переменную с помощью оператора присваивания: **Z:=(SUMMA + 5) * ADR - REZINTER / (ZNAM - 2.6);**

Рекомендуется следить за тем, чтобы тип переменной совпадал с типом выражения. При несовпадении типов ПК попытается преобразовать тип результата вычисления арифметического выражения к типу переменной. Если переменная действительного типа (**REAL**), а выражение целого типа (**INTEGER**), то оператор присваивания выполнится успешно. Если переменная целого типа (**INTEGER**), а выражение действительного типа (**REAL**), ПК выведет сообщение об ошибке преобразования типа.

Пример 1. Записать оператор для вычисления выражения:

$$Y = a^4 + b^3.$$

В языке TP отсутствует операция возведения в степень, поэтому вместо нее чаще всего используют многократное умножение операндов на свое собственное значение:

$$y := a * a * a * a + b * b * b.$$

В выражениях могут использоваться функции TP. Некоторые наиболее часто используемые функции языка TP приведены в таблице 3.

Если операнды выражений содержат другие функции, необходимо использовать известные математические соотношения, например:

- для вычисления логарифма с произвольным основанием a используется соотношение

$$\log_a(x) = \frac{\ln(x)}{\ln(a)};$$

- для возведения положительного числа x в действительную степень a используется соотношение

$$x^a = e^{a \ln(x)}.$$

Таблица 3 – Основные функции алгоритмического языка TP

Математическая запись	Действие	Запись на TP	Пример
$ x $	Вычисление абсолютного значения x	ABS(X)	ABS(-4)=4
x^2	Вычисление квадрата x	SQR(X)	SQR(5)=25
\sqrt{x}	Вычисление квадратного корня из x	SQRT(X)	SQRT(25)=5
e^x	Вычисление экспоненты x	EXP(X)	EXP(1)=2.7183...
$\ln x$	Вычисление натурального логарифма	LN(X)	LN(EXP(1))=1
$\sin x$	Вычисление синуса x	SIN(X)	SIN(PI/6)=0.5
$\cos x$	Вычисление косинуса x	COS(X)	COS(PI/3)=0.5
$\arctg x$	Вычисление арктангенса x	ARCTAN(X)	ARCTAN(1)=PI/4=0.7854
	Округление x до целого	ROUND(X)	ROUND(2.5)=3
	Вычисление целой части x , результат – действительный	INT(X)	INT(2.5)=2.0
	Вычисление целой части x , результат – целый	TRUNC(X)	TRUNC(2.5)=2
	Вычисление дробной части x	FRAC(X)	FRAC(2.5)=0.5
	True, если x – нечетный; False, если x – четный	ODD(X)	ODD(3)=TRUE

Пример 2. Вычислить значение выражения:

$$\varphi = \frac{(y^b + |b| - \sqrt[3]{a})(e^a - \sqrt{a} + \cos y)}{\lg a - x^2 \ln y + \arcsin a},$$

при $a = 0,56$, $b = 2,734$, $x = 1,8 \cdot 10^{-3}$, результат $y = 0,35 \times 10^2$.

Так как функция \arcsin в языке TP отсутствует, для ее вычисления используем известное соотношение:

$$\arcsin x = \arctg \frac{x}{\sqrt{1-x^2}} .$$

Для уменьшения вероятности ошибок и упрощения отладки программы запишем заданное выражение в виде нескольких операторов присваивания:

A := 0.56; B := -2.734; X := 1.8e-3; Y := 0.35e2;

FI := (EXP(B*LN(Y)) + ABS(B) – EXP(LN(A)/3));

FI := FI*(EXP(A) - SQRT(A) + COS(Y));

FI := FI/(LN(A)/LN(10) - X*X* LN(Y) + ARCTAN(A/SQRT(1-A*A)));

В этих операторах переменные **A, B, X, Y** являются исходными данными, а переменная **FI** – результатом вычислений. Наиболее часто численные значения исходных данных вводятся с клавиатуры, а результаты вычислений выводятся на экран дисплея. Такие действия выполняют операторы ввода и вывода.

Операторы ввода данных

Для **ввода данных** в Паскале применяют операторы **READ** или **READLN**, которые имеют одинаковый формат записи:

READ (список ввода);

В операторе после ключевого слова **READ** следует заключенный в круглые скобки список переменных, переменные друг от друга отделяются запятыми. Оператор завершается точкой с запятой. Например:

VAR A,B,C: REAL;

.....

READ (A,B,C);

После запуска программы на выполнение осуществляется ввод данных, например **0.5 6.25 -2.5E-2**.

При выполнении оператора **READ** программа приостанавливает свои действия в ожидании ввода значений, которые по очереди в соответствии с приведенным в операторе списком будут присвоены переменным. Значения переменных могут вводиться с клавиатуры во время исполнения программы или читаться из файла. При вводе числовых значений (типы **INTEGER, REAL, BYTE**) их отделяют друг от друга как минимум одним пробелом или размещают в разных строках. В пределах одного числа пробелы между цифрами недопустимы. Тип вводимых значений должен соответствовать типу

переменных, которым они будут присвоены. В Паскале допускается ввод данных типов **INTEGER**, **REAL**, **CHAR**, **BYTE**.

Оператор **READLN** по своим действиям аналогичен оператору **READ**, за исключением того, что по окончании чтения значения последней переменной из списка ввода он дает указание на переход к началу новой строки для чтения данных последующими операторами **READ** или **READLN**. Когда оператор **READLN** может быть применен и без списка переменных, тогда будет произведен переход на новую строку. Например:

```
VAR A,B: REAL;  
    I,K: INTEGER;  
.....  
READ (A, B);  
READLN;           или   READLN (A, B);  
READ (I, K);      READ (I, K);
```

После запуска программы на выполнение данные для ввода должны быть расположены следующим образом:

```
1.5 7.48  
6 -1
```

Операторы вывода данных

Вывод данных организуется операторами **WRITE** и **WRITELN**, формат записи которых имеет вид: **WRITE (список вывода)**;

В списке вывода приводят разделенные запятыми переменные и выражения (типа **INTEGER**, **REAL**, **CHAR** и **BOOLEAN**), заключенные в кавычки строки символов. Например:

```
WRITE (DAY[3], Alfa/Sin(X), LETTERS, 'Результат =');
```

По умолчанию в строке вывода для данных целого типа отводится 13 позиций (мест для размещения цифр и знака “±” числа). Вещественные числа выводятся в экспоненциальном виде с плавающей точкой по формату $\pm \#.#####E\pm##$ (знаком # отображено место для цифр мантииссы и степени числа). Формат выводимых данных может быть изменен явным указанием числа отводимых позиций. Для этого после идентификаторов переменных (выражений, констант) в списке вывода указывается отведенная ширина поля для выводимой информации и точность представления числа (только для вещественных чисел). Эти две величины должны быть целыми и положительными, отделяться от имен переменных (выражений, констант) и друг от друга двоеточием. Они соответственно указывают количество позиций в строке, отводимых под значение переменной (выражения, константы), и количество мест – под цифры дробной части числа.

Например, **WRITE ('I=', I:3, 'K=', K:4:2);** – на экране появится **I=-12 K=3.25**

В отличие от **WRITE**, оператор **WRITELN** автоматически организует перевод курсора на новую строку. При его использовании последующий оператор **WRITE** или **WRITELN**, если он присутствует в программе, начинает вывод информации с новой строки. Оператор **WRITELN** может применяться и без списка вывода, что приведет к переходу курсора на новую строку (если следует после **WRITE**) или к пропуску пустой строки (если следует после **WRITELN**). Например, после выполнения операторов:

```
WRITELN ('ЗАДАЧА');  
WRITELN ('I=', I:3, 'K=', K:4:2);  
WRITELN ('Z=', Z:4:1);  
WRITELN ('Y=', Y);
```

на экране появится следующая информация:

```
ЗАДАЧА  
I=-12 K= 3.25  
Z=-9.5  
Y=3.7800E+02.
```

Очистка экрана выполняется оператором **CLRSCR**, который удаляет всю имеющуюся на экране информацию и устанавливает курсор в его левый верхний угол.

Структура программы

Программа в Паскале структурно состоит из заголовка, области описания данных, разделов процедур и функций, а также операторов, составляющих ее тело:

```
PROGRAM имя;  
LABEL метка [,метка]...;  
CONST имя константы = значение константы;  
TYPE имя типа = значение констант типа;  
VAR имя [,имя]... : тип данных;  
PROCEDURE имя процедуры [список формальных параметров с указанием типа];  
    [Блок процедуры];  
FUNCTION имя функции [список формальных параметров с указанием типа] : тип значения функции;  
    [Блок функции];  
BEGIN {начало тела программы}  
    оператор [; оператор]...
```

END.

В заголовке программы после служебного слова **PROGRAM** указывают ее имя, образованное по тем же правилам, что и имя переменных. Заголовок оформляется в виде отдельной строки и оканчивается “;”.

В Паскале в обязательном порядке указывается тип всех используемых в программе данных. Область описания данных включает разделы описания меток (**LABEL**), констант (**CONST**), типов (**TYPE**) и переменных (**VAR**). Каждый из них, а также операторы внутри разделов отделяются друг от друга точкой с запятой “;”. Если в программе данные каких-либо типов отсутствуют, то и соответствующий раздел описания данных не приводится.

Любой оператор в программе может быть выделен меткой, которая обозначается либо натуральным числом, либо именем, образованным по тем же правилам, что и имя переменной. Метка отделяется от оператора двоеточием, например: **30 : WRITELN ('C=', C);**.

На метки в программе могут быть сделаны ссылки, в частности в операторе безусловного перехода. Все используемые метки должны быть перечислены в разделе **LABEL** области описания данных, например: **LABEL 5, 20, 30, 100;**

Если в программе используются константы, имеющие достаточно громоздкую запись, либо сменные константы (для разных вариантов программы), то такие константы целесообразно описать в разделе **CONST**, например: **CONST N=100;**

Следует отметить, что в разделе **CONST** используется знак равенства “=”, а не присваивания “:=”.

Все переменные, встречающиеся в программе, должны быть описаны в разделе **VAR**, где после имени переменной (или разделенных запятыми нескольких имен) и двоеточия приводится одно из служебных слов: **INTEGER** (целочисленный), **REAL** (вещественный), **BOOLEAN** (логический), **CHAR** (символьный).

В разделе **TYPE** описываются типы переменных, отличные от стандартных. Например: **TYPE COLOR=(RED, GREEN, BLUE, YELLOW);**

За областью описания данных следует раздел операторов. Он начинается с ключевого слова **BEGIN** и заканчивается словом **END**, после которого ставится точка. Программа при отсутствии явных команд, изменяющих последовательность ее действий, выполняется построчно по мере следования операторов (слева-направо и сверху-вниз). В каждой строке программы может присутствовать несколько операторов, отделенных друг от друга точкой с запятой “;”. Перед

END (после последнего используемого оператора) разделитель операторов “;”, как правило, не ставят, но его наличие не является ошибкой.

Раздел описания переменных **VAR** и раздел операторов должны присутствовать всегда, остальные разделы могут отсутствовать.

Лекция 3

Алгоритмы линейной структуры и их программирование

План лекции:

1. Схемы алгоритмов линейной структуры.
2. Примеры решения задач.

Линейными называют алгоритмы, в которых операции выполняются последовательно одна за другой, в естественном и единственном порядке.

Рассмотрим реализацию простейшего линейного алгоритма и приведем полный текст программы на языке Turbo-Pascal. В таких алгоритмах все блоки имеют последовательное соединение логической связью передачи информационных потоков. В них могут использоваться все блоки, за исключением блоков проверки условия и модификации. Линейные алгоритмы, как правило, являются составной частью любого алгоритмического процесса.

Пример 3 Вычислить значение функции:

$$Y = \arctg \left[\sqrt{\frac{ax^2 + b}{c}} + \left(\frac{ax^2 + b}{c} \right) \right].$$

Очевидно, что функцию Y целесообразно вычислять в такой последовательности: предварительно введя исходные данные a, b, c и присвоив значение переменной x , вначале найдем значение выражения, которое обозначим переменной z , и далее определим выражение Y . Используя общепринятые символы блоков (рисунок 1), изобразим схему разрабатываемого алгоритма (рисунок 2) и составим программу: при значении аргумента $x = 1,5$ и произвольных значениях переменных a, b, c .

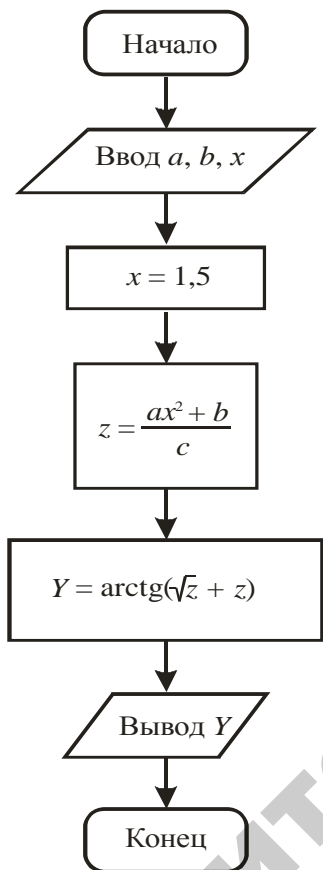


Рисунок 2 – Схема алгоритма линейной структуры примера 3

После запуска программы на выполнение вводим численные значения исходных данных a , b , c с клавиатуры, разделяя числа пробелами. После выполнения программы на экране появится результат вычислений.

Рассмотрим реализацию простейшего линейного алгоритма на языке TP и приведем полный текст программы.

Пример 4. Вычислить значение функции Y по формуле $Y = \sin a^2 + 4ax^3 + |a - b|$ для заданных значений переменных a , x , b ($a = 1,5$, $x = 2$, $b = 5$).

```

PROGRAM PRIMER2;
VAR B,C:INTEGER;
A,X,Y,Z:REAL;
BEGIN
  WRITELN('ВВЕДИТЕ A,B,C');
  READLN(A,B,C);
  X:=1.5;
  Z:=(A*X*X+B)/C;
  Y:=ARCTAN(SQRT(Z) + Z);
  WRITELN('Y=',Y:8:4);
END.
  
```

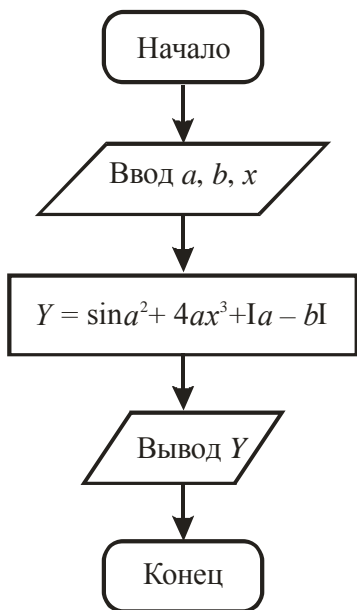


Рисунок 3 – Схема алгоритма линейной структуры примера 4

```

PROGRAM PRIMER4;
VAR X,B: INTEGER;
    A,Y: REAL;
BEGIN
  WRITELN ('ВВЕДИТЕ A, B, X');
  READLN (A,B,X);
  Y:=SIN(A*A)+4*A*X*X*X+ABS(A-B);
  WRITELN ('Y=', Y:8:4);
END.
  
```

После запуска программы на выполнение вводим численные значения исходных данных A , B , X с клавиатуры, разделяя числа пробелами: **1.5 2 5**.

После выполнения программы на экране появится результат вычислений в виде: **Y= 52.2781**.

Лекция 4 Программирование разветвляющихся алгоритмов

План лекции

1. Схемы алгоритмов разветвляющейся структуры
2. Оператор безусловного перехода GOTO
3. Оператор условного перехода IF

Оператор безусловного перехода GOTO

Общий вид оператора: **GOTO <метка>;**

Метка ставится перед оператором, которому передается управление и отделяется от него двоеточием. Метка должна быть описана в разделе **LABEL**. Например:

```
.....  
GOTO 50;  
10 S:=S+A;  
.....  
50 WRITELN (Y);  
.....
```

Необходимо помнить, что оператор, следующий непосредственно за **GOTO**, всегда должен иметь метку, в противном случае он никогда не будет выполняться.

Оператор условного перехода **IF**

Для программирования разветвляющихся алгоритмов используются операторы перехода, которые позволяют изменять последовательность выполнения операторов программы. Операторы перехода, обеспечивающие выполнение той или иной ветви алгоритма в зависимости от выполнения некоторого условия, называются условными.

Условный оператор **IF** записывается в виде:

```
IF <логическое выражение> THEN <оператор 1>  
                                  ELSE <оператор 2>;.
```

Оператор условного перехода **IF** может быть представлен схемой (рисунок 4). Если логическое выражение истинно, то выполняется оператор 1, если нет – оператор 2. В качестве операторов 1, 2 могут использоваться другие операторы **IF**. В этом случае каждое **ELSE** относится к ближайшему слева **IF**. Рекомендуется слово **ELSE** писать под соответствующим ему словом **THEN** для удобства восприятия программы. Перед **ELSE** “;” никогда не ставится.

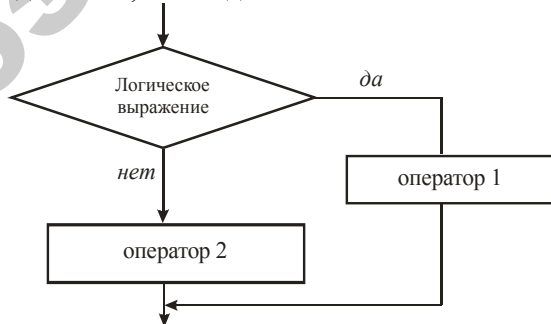


Рисунок 4 – Схема оператора условного перехода **IF**

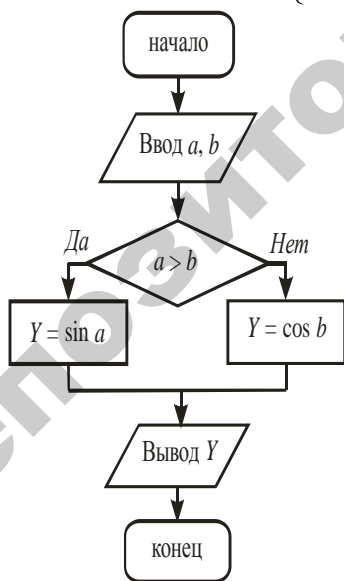
В логических выражениях используются операции отношения (таблица 4). Результатом вычисления логического выражения могут быть только два значения: истина (**TRUE**) или ложь (**FALSE**).

Таблица 4 – Операции отношения алгоритмического языка TP

Операция	Действие	Выражение	Результат
=	Равно	$A=B$	True, если $A=B$
\diamond	Не равно	$A \diamond B$	True, если $A \diamond B$
<	Меньше	$A < B$	True, если $A < B$
>	Больше	$A > B$	True, если $A > B$
\leq	Меньше или равно	$A \leq B$	True, если $A \leq B$ или $A=B$
\geq	Больше или равно	$A \geq B$	True, если $A \geq B$ или $A=B$

Пример 5. При заданных значениях переменных a и b вычислить Y по одной из формул:

$$Y = \begin{cases} \sin a, & \text{если } a > b; \\ \cos b, & \text{если } a \leq b. \end{cases}$$



PROGRAM PRIMER5;

VAR A,B,Y: REAL;

BEGIN

Writeln ('ВВЕДИТЕ A, B');

Readln (A,B);

IF A>B THEN Y:=SIN(A)

ELSE Y:=COS(B);

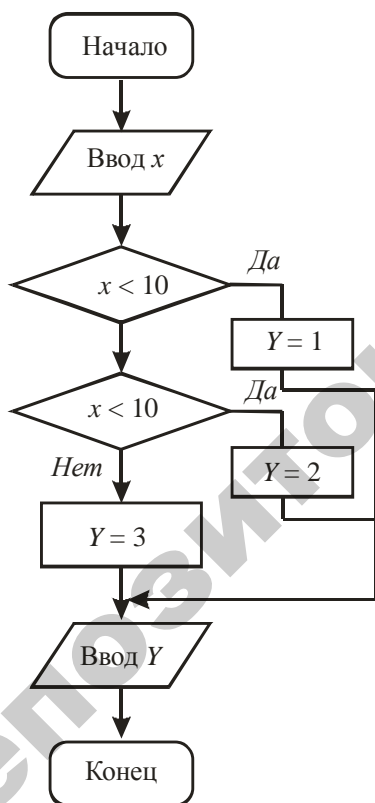
Writeln ('Y=',Y:10:6);

END.

Рисунок 5 – Схема алгоритма решения примера 5

Пример 6. При заданном значении x вычислить Y по одной из формул:

$$Y = \begin{cases} 1, & \text{а́ннèè } x < 0; \\ 2, & \text{а́ннèè } 0 \leq x < 10; \\ 3, & \text{а́ннèè } x \geq 10. \end{cases}$$



```

PROGRAM PRIMER6;
VAR X,Y: INTEGER;
BEGIN
  WRITELN ('ВВЕДИТЕ X');
  READLN (X);
  IF X<0 THEN Y:=1
  ELSE IF X<10 THEN Y:=2
  ELSE Y:=3;
  WRITELN ('Y=',Y:1);
END.
  
```

Рисунок 6 – Схема алгоритма решения примера 6

У оператора **IF** ветвь **ELSE** может отсутствовать, тогда оператор записывается в следующем виде:

IF <логическое выражение> THEN <оператор 1>;
и реализующая его схема имеет вид:

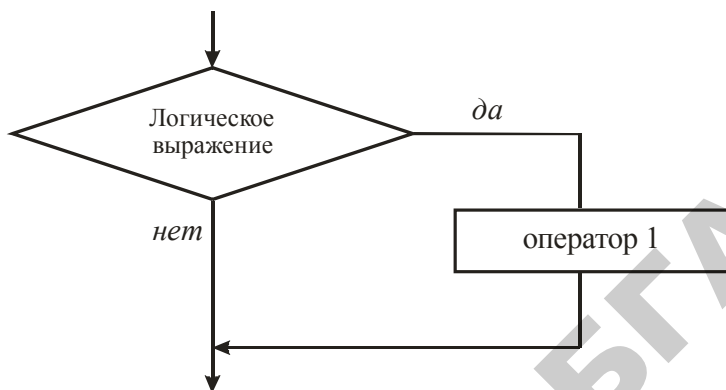


Рисунок 7 – Схема условного оператора IF при отсутствии ветви ELSE

В логических выражениях для записи сложных условий, кроме операций отношения, могут использоваться логические операции (таблица 5).

Таблица 5 – Логические операции алгоритмического языка ПР

Операция	Действие	Выражение	<i>A</i>	<i>B</i>	Результата
Not	Отрицание	Not A	True		False
And	И	A and B	True	True	True
			True	False	False
			False	True	False
			False	False	False
Or	ИЛИ	A or B	True	True	True
			True	False	True
			False	True	True
			False	False	False
Xor	Исключающее или	A xor B	True	True	False
			True	False	True
			False	True	True
			False	False	False

Вопросы для самоконтроля

1. Назовите этапы подготовки и решения задач на ПЭВМ.
2. Дайте определение алгоритма.
3. Какие вы знаете свойства алгоритма?
4. Перечислите известные вам способы описания алгоритма.
5. Что такое схема алгоритма?
6. Как обозначаются операции ввода и вывода данных в схеме алгоритма?
7. Какой геометрической фигурой обозначается блок «проверка условия» в схеме алгоритма?

МАТЕРИАЛЫ К ЛАБОРАТОРНОЙ РАБОТЕ

Тема: Алгоритмы разветвляющейся структуры.

Цель: уметь составлять схемы алгоритмов разветвляющейся структуры и программы на языке Turbo-Pascal для решения практических задач.

Студенту следует для трех задач, условия которых приведены ниже, выполнить следующие задания:

- составить схему алгоритма решения задачи;
- записать пояснения к схеме алгоритма;
- составить программу на языке Turbo-Pascal;
- проверить выполнение составленной программы на контрольном примере, приняв упрощенные значения исходных данных по своему усмотрению.

Наиболее наглядным способом составления алгоритма является графический, т. е. изображение алгоритма решения задачи в виде схемы. При составлении алгоритмов разветвляющейся структуры необходимо указать дальнейшее направление вычислительного процесса по одному из нескольких заранее определенных направлений в зависимости от выполнения некоторого логического условия.

При составлении схем алгоритма следует:

- четко определить, что является исходными данными и в каком виде должен получиться результат решения задачи;
- словесно сформулировать действия, которые необходимо выполнить для получения результата решения задачи, затем записать последовательность действий с помощью блоков в виде схемы алгоритма.

ПРИМЕРЫ РАЗНОУРОВНЕВЫХ ЗАДАНИЙ ДЛЯ КОНТРОЛЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО МОДУЛЮ

I уровень

Вариант 1

1. Вычислить R при любых заданных значениях x и a :

$$R = \begin{cases} \frac{a^2}{\sin^2 \sigma + 1}, & \text{если } x < 0; \\ \frac{a^6}{|x-4|}, & \text{если } x \geq 0. \end{cases}$$

2. Даны три числа a , b , c . Напечатать те из них, которые меньше 15.
3. Взвешивание поросенка показало, что его масса за n дней увеличилась с m_1 до m_2 кг. Определить, достигнет ли среднесуточный привес поросенка запланированной нормы d кг.

Вариант 2

1. Вычислить Z при любых заданных значениях a , b , t :

$$Z = \begin{cases} \sqrt{at^2 + b \sin t + 1}, & \text{при } t < 0,1 \\ at + b, & \text{при } t = 0,1 \\ \sqrt{at^2 + b \sin t + t}, & \text{при } t > 0,1 \end{cases}$$

2. Даны три числа a , b , c . Напечатать те из них, которые больше 0, но меньше 10.
3. Вывести на экран номера точек с координатами $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$, которые лежат на оси OX , и определить их количество.

Вариант 3

1. Вычислить **S** при любых заданных значениях **a, b, c, f**:

$$S = \begin{cases} \frac{a}{c} + bc^2 + f, & \text{при } c < 3; \\ c - \cos f, & \text{при } 3 \leq c \leq 7; \\ ac + bc^3, & \text{при } c > 7. \end{cases}$$

2. Даны две фигуры: квадрат со стороной **a** и круг с радиусом **r**. Определить, какая из фигур имеет бóльшую площадь, и вывести эту площадь на печать.

3. Определить, попадет ли команда “Динамо” в лигу сильнейших (т. е. наберет не менее 20 очков). За выигрыш команда получает 2 очка, за ничью – 1 очко, за проигрыш – 0 очков. Команда **m** встреч выиграла, **k** – проиграла, а **p** завершила вничью.

Вариант 4

1. Вычислить **D** при любых заданных значениях **a, c, x**:

$$D = \begin{cases} \frac{c^5}{x^2 + 2}, & \text{если } x < 0; \\ \frac{a + \sin x}{|x - 2|}, & \text{если } x \geq 0. \end{cases}$$

2. Даны три числа **a, b, c**. Напечатать те из них, которые больше заданной величины **z**. Если таких чисел нет – вывести сообщение об этом.

3. Определить, можно ли огородить изгородью длиной **z** два земельных участка, имеющих форму прямоугольника со сторонами **c, d** и форму квадрата со стороной **a**.

Вариант 5

1. Вычислить Z при любых заданных значениях a, b, t :

$$Z = \begin{cases} t^2 + 2a - b, & \text{при } t < 0,1; \\ at + b, & \text{при } t = 0,1; \\ \frac{b}{|t - 1|}, & \text{при } t > 0,1. \end{cases}$$

2. Даны три неравных между собой числа a, b, c . Наибольшее из них разделить на сумму двух оставшихся. Вывести на экран полученный результат и наибольшее число.

3. Определить и вывести на экран номера точек с координатами $A(x_1, y_1), B(x_2, y_2), C(x_3, y_3)$, которые лежат на оси OY .

II уровень

Вариант 6

1. Вычислить C при любых заданных значениях x, a, b :

$$C = \begin{cases} a^3 - b^2 + 2\sin x, & \text{если } x > a; \\ a^2 + b^2 - \operatorname{tg} x, & \text{если } x = a; \\ a - 2b^4 + \cos x, & \text{если } x < a. \end{cases}$$

2. Определить количество неотрицательных чисел среди трех заданных a, b, c . Если таких чисел нет – вывести об этом сообщение.

3. Деревня А находится на расстоянии c км от железнодорожной станции и на расстоянии r км от другой деревни В (рисунок 8).

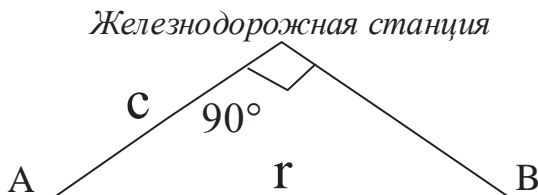


Рисунок 8

Определить, какая из деревень находится ближе к станции, напечатать сообщение об этом.

Вариант 7

1. Вычислить **D** при любых заданных значениях **x, a**:

$$D = \begin{cases} (x - a)^2, & \text{если } x > 0; \\ 2x + a, & \text{если } x = 0; \\ \frac{a^2 - x}{8}, & \text{если } x < 0. \end{cases}$$

2. Определить, является ли число **b** наименьшим из четырех, не равных между собой **a, b, c, d**.

3. Даны две фигуры: квадрат со стороной **a** и прямоугольник со сторонами **c, d**. Определить, какая из фигур имеет больший периметр, и вывести сообщение об этом на печать.

Вариант 8

1. Вычислить **C** при любых заданных значениях **x, a**:

$$C = \begin{cases} ax^2 - 4, & \text{если } x < -2; \\ x^4 - 1, & \text{если } -2 \leq x < 0; \\ x^3 + 8x, & \text{если } 0 \leq x < 2; \\ 4\sin x, & \text{если } x \geq 2. \end{cases}$$

2. Определить, могут ли произвольные числа **a, b, c** служить длинами сторон треугольника.

3. Имеются 3 клубня шарообразной формы радиусами **r1, r2, r3**. Определить, сколько из них пройдет через круглое отверстие сепарирующего решета площадью **s**.

Вариант 9

1. Вычислить **D** для любых произвольных значений **x, c**:

$$D = \begin{cases} \cos^3 x, & \text{иде } x < c, \\ 0,6c^2 + |x - 3|, & \text{иде } x \geq c. \end{cases}$$

2. Составить алгоритм вычисления действительных корней квадратного уравнения $ax^2 + bx + c = 0$ по формулам:

$$x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}, \text{ где } D = b^2 - 4ac.$$

3. Даны три действительных числа **a**, **b**, **c**. Вывести на печать те из них, которые принадлежат отрезку $[1,5]$, если таких чисел нет, то вывести на печать сообщение об этом.

Вариант 10

1. Вычислить **C** при любых заданных значениях **x**, **a**, **b**:

$$C = \begin{cases} a^2 - b^2 + 2\sin x^2, & \text{если } x > a; \\ a^2 + b - \operatorname{tg} x, & \text{если } x = a; \\ a - 2b^4 + \cos x, & \text{если } x < a. \end{cases}$$

2. Даны три числа **a**, **b**, **c**. Определить, есть ли среди них отрицательные, вывести соответствующее сообщение на экран.

3. При осмотре леса лесник определил численность деревьев на контрольном участке: сосен – **N1**, елей – **N2**, берез – **N3**. Из них здоровыми являются соответственно **M1**, **M2**, **M3**, остальные повреждены. Определить, какого вида деревьев больше всего повреждено и вывести сообщение на печать.

Вариант 11

1. Вычислить **F** при любых заданных значениях **x**, **c**:

$$D = \begin{cases} \cos x^3, & \text{и́дè } x < c, \\ 6c^2 + |x-9|, & \text{и́дè } x \geq c. \end{cases}$$

2. Вычислить $Q = \max(x, y, z) + x^3$. Значение **x**, **y**, **z** заданы.

3. Суточная норма кормления одной коровы составляет **a** кг сена, одной лошади – **b** кг сена. Определить, можно ли прокормить **k** коров и **m** лошадей в течение **n** дней, располагая массой **p** кг сена, если нет, то определить, сколько кг сена не хватает.

Вариант 12

1. Вычислить **S** при любых заданных значениях **x**, **c**:

$$S = \begin{cases} \operatorname{tg} 0,1x, & \text{ï ðè } x < c, \\ 1 + \ln |x|, & \text{ï ðè } x \geq c. \end{cases}$$

2. Среди 3-х точек с координатами (x_1, y_1) , (x_2, y_2) , (x_3, y_3) определить количество точек, лежащих в 3-ей четверти.

3. В колхозе имеется 3 комбайна ККУ-2А со средней производительностью **s** га в день и 5 комбайнов КПК-3 со средней производительностью **p** га в день. Будет ли выполнена колхозом уборка поля площадью **s** га в запланированный срок, составляющий 10 дней?

Вариант 13

1. Вычислить **D** при любых заданных значениях **a**, **x**:

$$D = \begin{cases} |x| + a - 4, & \text{если } a > 2; \\ \frac{x^2 + 2ax}{3}, & \text{если } a = 2; \\ 4x + a, & \text{если } a < 2. \end{cases}$$

2. Даны две фигуры: квадрат со стороной **a** и круг с длиной окружности **L**. Определить, какая из фигур имеет большую площадь, и вывести эту площадь на печать.

3. Имеется 10 круглых луковиц диаметром **d1**, 6 луковиц диаметром **d2** и 15 луковиц диаметром **d3**. Радиус отверстия в сепарирующем решете **p**. Определить, сколько луковиц пройдет сквозь отверстие решета.

Вариант 14

1. Вычислить **C** при любых заданных значениях **d**, **x**:

$$C = \begin{cases} \cos x^2 + d + 1, & \text{если } d > 1; \\ \frac{2x^2 + 4dx}{2}, & \text{если } d = 1; \\ 8x - d, & \text{если } d < 1. \end{cases}$$

2. Заданы 4 неравных между собой числа **a**, **b**, **c**, **d**. Определить наибольшее из них и вывести на печать.

3. В квадрат со стороной x вписан круг. Определить, площадь какой фигуры больше: квадрата или круга и вывести сообщение на печать.

Вариант 15

1. Вычислить F при любых заданных значениях x, c :

$$F = \begin{cases} |x| + x - 4, & \text{если } x > 0.2 \\ \frac{c^2 + 2cx}{2}, & \text{если } x = 0.2 \\ \cos^2 x - 1, & \text{если } x < 0.2 \end{cases}$$

2. Среди 3-х точек с координатами $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ определить количество точек, лежащих в 4-ой четверти.

3. Каждая из трех доярок надоила за месяц (30 дней) соответственно P_1, P_2, P_3 литров молока. Определить, у какой доярки наибольший среднесуточный надой молока и вывести на экран ее номер.

Вариант 16

1. Вычислить T при любых заданных значениях a, d, b, c :

$$T = \begin{cases} 5, & \text{если } d < a; \\ 4, & \text{если } a \leq d \leq b; \\ 3, & \text{если } b < d < c; \\ 2, & \text{если } d \geq c. \end{cases}$$

2. Определить количество положительных чисел среди четырех a, b, c, d .

3. Определить, можно ли огородить изгородью длиной g земельный участок, имеющий форму равнобедренной трапеции с основаниями c, d и высотой h .

Вариант 17

1. Вычислить **D** при любых заданных значениях **x**, **a**, **z**:

$$D = \begin{cases} |ax - 3|, & \text{если } x < 2; \\ 8x - 1, & \text{если } 2 \leq x < 3; \\ \frac{4}{z} + a^3, & \text{если } x \geq 3. \end{cases}$$

2. Определить, какая из сторон **k**, **l**, **m** прямоугольного треугольника является его гипотенузой, и вывести ее значение на печать.

3. Определить, какая из трех точек с координатами $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$ наиболее удалена от начала координат и вывести на печать ее номер.

Вариант 18

1. Вычислить **Z** при любых заданных значениях **a**, **b**, **x**, **c**:

$$Z = \begin{cases} ax^2/2 + bx - c, & \text{если } x \leq 1; \\ x + 1,5, & \text{если } 1 < x < 2; \\ 2x^2 - b, & \text{если } x \geq 2. \end{cases}$$

2. Проверить, является ли треугольник со сторонами **a**, **b**, **c** равнобедренным и вывести на экран соответствующее сообщение.

3. Даны две фигуры: квадрат с диагональю **d** и прямоугольная трапеция со сторонами **x**, **y**, **z** (рисунок 9). Определить, какая из фигур имеет больший периметр, сообщение вывести на печать.

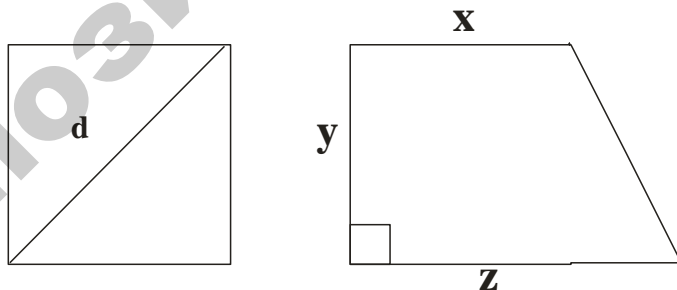


Рисунок 9 – Квадрат с диагональю **d** и прямоугольная трапеция со сторонами **x**, **y**, **z**

Вариант 19

1. Вычислить **T** для любых произвольных значений **x** и **c**:

$$T = \begin{cases} \ln |0,3x|, & \text{ї ðè } x < c, \\ \sqrt{4-x}, & \text{ї ðè } x \geq c. \end{cases}$$

2. В прямоугольном треугольнике заданы три стороны **k**, **l**, **m**. Вывести на печать значение гипотенузы и найти площадь треугольника.

3. Заданы четыре переменные **x**, **y**, **z**, **d**. Найти и вывести на печать количество переменных, попавших в интервал от -5 до 5 , если таких нет – дать сообщение.

Вариант 20

1. Вычислить **D** при любых заданных значениях **x**, **a**, **b**:

$$D = \begin{cases} ax + b, & \text{при } x \leq 0; \\ |b| - 2, & \text{при } 0 < x \leq 1; \\ |a| - x^2, & \text{при } x > 1. \end{cases}$$

2. Вычислить $Q = \min(a, b, c) + z$. Значения **a**, **b**, **c**, **z** заданы.

3. Даны две фигуры: квадрат с диагональю **d** и прямоугольная трапеция со сторонами **x**, **y**, **z**. Определить, какая фигура имеет большую площадь, сообщение вывести на печать (см. рисунок 9).

Вариант 21

1. Вычислить **Y** при любых заданных значениях **a**, **x**, **b**:

$$Y = \begin{cases} \sin bx, & \text{при } x < a; \\ \cos ax, & \text{при } a \leq x \leq b; \\ e^x + ab, & \text{при } x > b. \end{cases}$$

2. Среди 3-х точек с координатами (x_1, y_1) , (x_2, y_2) , (x_3, y_3) определить количество точек, лежащих в 1-ой четверти.

3. Имеются четыре результата **t1**, **t2**, **t3**, **t4** (сек.) в беге на 100 метров. Вывести на печать результат и номер победителя.

Вариант 22

1. Вычислить Z при любых заданных значениях a :

$$Z = \begin{cases} a \sin x^2, & \text{если } a \leq -3; \\ 1, & \text{если } -3 < a < 4; \\ \cos(a + 1), & \text{если } a \geq 4. \end{cases}$$

2. Среди 3-х точек с координатами (x_1, y_1) , (x_2, y_2) , (x_3, y_3) определить количество точек, лежащих в 2-ой четверти и вывести на экран их координаты.

3. Четыре студента сдали экзамен по физике и получили соответственно оценки x , y , z , d . Определить и вывести на печать номера студентов, оценка которых превышает средний балл по предмету.

Вариант 23

1. Вычислить S при любых заданных значениях a , b , c :

$$S = \begin{cases} \frac{a}{c} + bc^2 + \cos b^2, & \text{при } c < 3; \\ \cos^3 a - c, & \text{при } 3 \leq c \leq 7; \\ ac + bc^3 + 2, & \text{при } c > 7. \end{cases}$$

2. Даны две фигуры: квадрат с периметром z и круг с длиной окружности c . Определить, какая из фигур имеет большую площадь, и вывести эту площадь на печать.

3. Четыре студента сдали экзамен по физике и получили соответственно оценки a , b , c , d . Определить и вывести на печать количество и номера студентов, которые получили по экзамену оценку 9.

Вариант 24

1. Вычислить Z при любых заданных значениях x :

$$Z = \begin{cases} x^3 - 3x + 8, & \text{при } x \leq 0; \\ 4, & \text{при } 0 < x \leq 1; \\ 1 / (x^3 - 3x + 8), & \text{при } x > 1. \end{cases}$$

2. Среди трех чисел a , b , c есть пара равных. Заменить их нулями и все числа вывести на печать.

3. Имеются четыре результата **z1, z2, z3, z4** (м) по прыжкам в высоту. Вывести на печать количество и номера спортсменом, которые прыгнули выше заданного норматива **L(m)**.

III уровень

Вариант 25

1. Вычислить **U** при любых заданных значениях **x, a, b**:

$$U = \begin{cases} x^2 - 3a + \cos^2 x, & \text{при } x \leq 3; \\ |b - a|, & \text{при } 3 < x \leq 5; \\ (x^3 - 4), & \text{при } x > 5. \end{cases}$$

2. Определить, является ли треугольник со сторонами **a, b, c** прямоугольным (для определения использовать теорему Пифагора).

3. Определить, какая из трех точек с координатами **A(x₁, y₁)**, **B(x₂, y₂)**, **C(x₃, y₃)** лежит ближе к началу координат и вывести на печать ее номер.

Вариант 26

1. Вычислить **F** для любых произвольных значений **x** и **c**:

$$F = \begin{cases} \operatorname{tg}(x-14), & \text{и } \text{д} \text{е } x < c, \\ \sqrt{|x-10|}, & \text{и } \text{д} \text{е } x \geq c. \end{cases}$$

2. Задана окружность с центром в начале координат и точка **A** с координатами **(x, y)**, лежащая на окружности. Определить, можно ли в этот круг вписать квадрат со стороной **d**.

3. Надои молока, полученные каждой из трех доярок Петровой, Ивановой, Сидоровой за 30 дней, составляют соответственно **P1, P2, P3** литров. Вывести на печать фамилии и надои тех доярок, среднесуточные надои которых достигают планового показателя, равного **R** литров.

Вариант 27

1. Вычислить **R** при любых заданных значениях **x, a, b**:

$$R = \begin{cases} \frac{|x-a|}{|x-b|}, & \text{если } x < a; \\ x-a+b, & \text{если } x = a; \\ ba - \cos^2 x, & \text{если } x > a. \end{cases}$$

2. Бригада из четырех рабочих собирала картофель, каждый собрал соответственно **a, b, c, d** (кг). Определить, превышает ли средняя выработка в бригаде заданную величину **S** (кг) и вывести на печать номера рабочих, которые не достигли средней выработки по бригаде.

3. Даны три неравных между собой числа **x, y, z**. Наибольшее из них разделить на сумму двух других.

Вариант 28

1. Вычислить **D** для любых произвольных значений **x** и **c**:

$$D = \begin{cases} \sqrt{9+x}, & \text{ï ðè } x < c; \\ 0,8x \ln x, & \text{ï ðè } x \geq c \end{cases}$$

2. Определить, поместится ли круг с радиусом **r** в квадрат со стороной **a**.

3. Три студента сдали три экзамена и получили соответственно оценки: I студент – **x1, x2, x3**; II студент – **y1, y2, y3**; III студент – **z1, z2, z3**. Вывести на печать номера тех студентов, которые будут получать стипендию, если для этого необходимо иметь средний балл, равный 7,5.

Вариант 29

1. Вычислить **G** для любых произвольных значений **x** и **c**:

$$G = \begin{cases} \frac{1}{\cos x}, & \text{ï ðè } x < c; \\ 0,5x \operatorname{tg} x, & \text{ï ðè } x \geq c. \end{cases}$$

2. Вычислить **Q** = max (x, y, z) + 4. Значение **x, y, z** заданы.

3. Заданы три точки с координатами **A(x1, y1), B(x2, y2), C(x3, y3)** и круг с центром в начале координат и радиусом **R**. Определить, сколько точек принадлежит кругу.

Вариант 30

1. Вычислить \mathbf{B} для любых произвольных значений x и c :

$$\mathbf{B} = \begin{cases} \sqrt{|6-x|}, & \text{ї ðè } x < c; \\ \sin \frac{x}{2}, & \text{ї ðè } x \geq c. \end{cases}$$

2. Вычислить $\mathbf{Q} = \min(x, y, z) + \mathbf{d}$. Значение x, y, z, \mathbf{d} заданы.

3. Цех по переработке молочной продукции за одну смену выпускает пастеризованного молока \mathbf{a} кг, кефира – \mathbf{b} кг. Определить, выполнит ли цех плановые показатели за месяц (30 дней) по каждому из видов продукции, составляющих соответственно \mathbf{s} кг и \mathbf{p} кг, если в первую декаду он работал в одну смену, во вторую – в полторы, в третью – в две смены.

ЗАДАНИЯ ДЛЯ УПРАВЛЯЕМОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Самостоятельно изучить и выполнить раздел дисциплины «Программирование алгоритмов линейной структуры» по плану:

- 1) составить схемы алгоритмов решения задач (используя графический способ описания алгоритма): детальную и укрупненную. Записать пояснения к схемам алгоритмов;
- 2) составить программы на языке Turbo-Pascal;
- 3) проверить выполнение составленных программ на конкретном примере, приняв упрощенные значения исходных данных по своему усмотрению.

Вариант 1

1. Составить схему алгоритма и программу для вычисления выражения:

$$t = \frac{2 \cos(x - \frac{\pi}{6})}{0,5 + \sin^2 y} \left(1 + \frac{z^2}{3 - z^2/5}\right).$$

2. Садовый участок имеет форму полукруга, диаметр которого равен D (м). Участок по всему периметру необходимо огородить забором. Определить, какое количество столбиков и сколько руло-

нов сетки необходимо заказать для забора, если расстояние между столбиками должно быть Z (м), а длина одного рулона сетки – G (м). В программе результаты расчета округлить до ближайшего целого числа в большую сторону.

Вариант 2

1. Составить схему алгоритма и программу для вычисления выражения:

$$u = \frac{\sqrt[3]{8 + |x - y|^2 + 1}}{x^2 + y^2 + 2} - (\operatorname{tg} z + 1)^x.$$

2. Картофельное поле имеет форму треугольника со сторонами A (м), B (м) и C (м). На ручной уборке картофеля задействована группа из N студентов, причем ребята составляют $M\%$ от общего числа студентов, остальной состав группы – девушки. Скорость уборки картофеля ребятами за 1 час работы составляет R (м²), а девушками – D (м²). Сколько дней необходимо запланировать на уборку поля, если продолжительность рабочего дня составляет T часов, из которых P часов используется на обеденный перерыв. В программе результат расчета округлить до ближайшего целого числа в большую сторону.

Вариант 3

1. Составить схему алгоритма и программу для вычисления выражения:

$$v = \frac{1 + \sin^2(x + y)}{\left| x - \frac{2y}{1 + x^2 y^2} \right|} x^{|y|}.$$

2. Длина, ширина и высота дома – A , B и H (м) соответственно. Дом имеет крышу, которая в поперечном сечении образует над потолком равнобедренный треугольник высотой V (м). Поверхность в торце дома, заключенная между крышей и потолком, называется фронтоном. В доме имеется одна дверь размером $C \times D$ (м) и N окон размером $E \times F$ (м). Стены дома планируется окрасить в зеленый цвет, фронтоны – в желтый, а крышу – в коричневый. Окна и двери решено не перекрашивать. Определить необходимое количество краски каждого цвета (в кг). Расход краски составляет Q (грамм на 1 м²).

Вариант 4

1. Составить схему алгоритма и программу для вычисления выражения

$$w = |\sin x - \sin y| \left(1 + a + \frac{a^2}{2} + \frac{b^3}{3} + \frac{z^4}{4}\right).$$

2. Автохозяйство имеет N легковых и M грузовых автомобилей. Плановый пробег за месяц эксплуатации одного легкового автомобиля составляет $L1$ (км), а грузового – $L2$ (км). Норма расхода бензина для легковых автомобилей составляет $Q1$ (литров на 100 км), а для грузовых – $Q2$ (литров на 100 км). Определить общую годовую потребность автохозяйства в бензине (в литрах), если фактически легковые автомобили простаивают без пробега $P1$ % времени в году, а грузовые – $P2$ %.

Вариант 5

1. Составить схему алгоритма и программу для вычисления выражения:

$$a = \left(x - \frac{y}{2}\right) * \left(\frac{1}{y\sqrt{|x|}}\right) + \sin^2 y.$$

2. Фермер имеет в своем хозяйстве N коров и M телят. Суточное потребление сена на одну корову составляет $Q1$ (кг), на одного теленка – $Q2$ (кг). На зиму фермер заготовил $K1$ тюков сена цилиндрической формы диаметром D (м) и высотой H (м) и $K2$ тюков сена в форме параллелепипеда размером $A \times B \times C$ (м). Удельный вес сена в тюках составляет R (кг/м³). Рассчитать количество дней, в течение которых фермер сможет прокормить животных. В программе предусмотреть округление результата до ближайшего меньшего целого числа.

Вариант 6

1. Составить схему алгоритма и программу для вычисления выражения:

$$B = \sqrt{x + 3y + z} (\sin z - |x - y|).$$

2. На уборке зерновых задействовано два комбайна, имеющие производительность работы $Q1$ и $Q2$ (га/час), которые вместе должны убрать площадь S (га). Первый комбайн убирает K % всей пло-

шадии, остальное – второй комбайн. Часовой расход топлива при работе комбайнов составляет R_1 и R_2 (л/ час) соответственно. Определить, сколько тонн топлива потребуется для уборки урожая, если удельный вес топлива составляет P (кг/л), и затраты хозяйства на топливо, если стоимость 1 тонны топлива составляет C (руб.).

Вариант 7

1. Составить схему алгоритма и программу для вычисления выражения:

$$d = 5 \operatorname{tg}(x) - \cos(x) \frac{x + 3|x - y| + x^2}{|x - y|z + x^2}.$$

2. Рожью засеяна площадь S_1 (га), пшеницей – S_2 (га). Урожайность ржи составляет Q_1 (ц/га), пшеницы – Q_2 (ц/га). Для доставки зерна с поля на элеватор используются автомобили грузоподъемностью P тонн. Продолжительность рабочего дня составляет T часов, на выполнение одного рейса (с поля на элеватор и обратно) автомобилю требуется R часов. Определить, сколько автомобилей необходимо привлечь к работе, чтобы обеспечить доставку зерна с поля за N дней. Указание: при расчете числа рейсов, которые выполняет один автомобиль за день, в программе предусмотреть округление результата до ближайшего меньшего целого числа.

Вариант 8

1. Составить схему алгоритма и программу для вычисления выражения

$$a = \frac{|x - y|^{x+y}}{\operatorname{tg}(x) + \operatorname{tg}(z)} + \sqrt[3]{x^6 + y}.$$

2. Посевная площадь картофеля в хозяйстве составляет S (га). Для его уборки используется N комбайнов производительностью Q_1 (га/час). Однако для уборки картофеля в запланированные D дней этого количества комбайнов недостаточно. Соседнее хозяйство предлагает в помощь комбайны производительностью Q_2 (га/час). Определить, сколько комбайнов соседей необходимо привлечь к уборке картофеля, если продолжительность рабочего дня составляет T часов, а время для ежедневного технического обслуживания каждого комбайна – E часов. В программе предусмотреть округление результата расчета до ближайшего большего целого числа.

Вариант 9

1. Составить схему алгоритма и программу для вычисления выражения:

$$Q = \left| x^3 - \sqrt{\frac{y}{x}} \right| + (y-x) \frac{\cos y - z}{1 + (y-x)^2}.$$

2. Для уборки зерна используется R комбайнов производительностью F (га/ час) и L комбайнов производительностью W (га/час). Продолжительность рабочего дня комбайнов составляет K часов, время для ежедневного технического обслуживания каждого комбайна – E часов. Определить, какую площадь уборут комбайны за M дней, если 1 комбайн производительностью F не работал два дня.

Вариант 10

1. Составить схему алгоритма и программу для вычисления выражения:

$$a = 2^{-x} \sqrt{x + \left(\frac{1}{y}\right) + \sin^2 y}.$$

2. Фермер построил две теплицы одинаковой длины L (м). Первая теплица имеет в поперечном сечении форму прямоугольника со сторонами размером A на B (метров), вторая – полукруг радиуса R (м). Определить, сколько рулонов полиэтиленовой пленки необходимо фермеру для укрытия всей поверхности теплиц (включая торцы), если известно, что ширина одного рулона пленки составляет H (м), а длина пленки в рулоне – D (м). В программе результат расчета округлить до ближайшего большего целого числа.

Вариант 11

1. Составить схему алгоритма и программу для вычисления выражения:

$$b = y^3 \sqrt[3]{|x|} + \cos^3(y) + (\operatorname{tg}^2 z + 1).$$

2. Садовый участок имеет форму равнобедренной трапеции, большее основание которой равно A (м), меньшее – B (м), высота – H (м). Участок по всему периметру необходимо огородить забором. Определить, какое количество столбиков и сколько рулонов сетки

необходимо заказать для забора, если расстояние между столбиками должно быть S (м), а длина одного рулона сетки – R (м).

Вариант 12

1. Составить схему алгоритма и программу для вычисления выражения:

$$c = \frac{\operatorname{tg}(z) - x^3}{|x| + \frac{1}{y^2 + 1}}.$$

2. Для уборки урожая используется K комбайнов производительностью A (га/час) и P комбайнов производительностью B (га/час). Продолжительность рабочего дня комбайнов составляет Z часов, время для ежедневного технического обслуживания каждого комбайна – S часов. Определить, какую площадь уборут комбайны за K дней, если 1 комбайн производительностью A не работал два дня, а 1 комбайн производительностью B не работал три дня.

Вариант 13

1. Составить схему алгоритма и программу для вычисления выражения:

$$F = \frac{\sqrt{z + \sqrt[3]{x-1}}}{|y-x|(\sin^2 z + \operatorname{tg} z)}.$$

2. Длина, ширина и высота дома равна K (м), R (м) и N (м) соответственно. Дом имеет крышу, которая в поперечном сечении образует над потолком равносторонний треугольник. Поверхность в торце дома, заключенная между крышей и потолком, называется фронтоном. В доме имеется одна дверь размером A на B (метров) и 8 окон размером X на Y (метров). Стены дома планируется окрасить в зеленый цвет, фронтоны и двери – в желтый, а крышу – в коричневый. Окна решено не перекрашивать. Определить необходимое количество краски каждого цвета (в килограммах), если расход краски составляет Q (грамм на 1 м^2).

Вариант 14

1. Составить схему алгоритма и программу для вычисления выражения:

$$g = \frac{y^{x+1}}{\sqrt[3]{y-2+3}} + \frac{x+y/2}{2|x+y|} (x+1).$$

2. Хозяйство имеет A легковых и B грузовых автомобилей. Средний пробег за месяц эксплуатации одного легкового автомобиля составляет K (км), а грузового – M (км). Норма расхода бензина для легковых автомобилей составляет $Z1$ (литров на 100 км), а дизтоплива для грузовых – $Z2$ (литров на 100 км). Рассчитать годовую потребность автохозяйства в бензине и дизтопливе, общие затраты на топливо, если стоимость 1 литра бензина составляет $C1$ (руб/л), а дизтоплива – $C2$ (руб/л).

Вариант 15

1. Составить схему алгоритма и программу для вычисления выражения:

$$h = \frac{x^{y+1}}{1+x|y-\text{tgz}|} (1+|y-x|) + \frac{|y-x|^2}{3}.$$

2. Фермер имеет в своем хозяйстве N коров и M телят. Суточное потребление сена на одну корову составляет G (кг), на одного теленка – F (кг). На зиму фермер заготовил скирду сена длиной L (м) и поперечным сечением в форме равнобедренной трапеции с основаниями A и B (м) и высотой H (м). Удельный вес сена составляет V (кг/м³). Рассчитать количество дней, в течение которых фермер сможет прокормить животных своим запасом сена. В программе предусмотреть округление результата до ближайшего меньшего целого числа.

Вариант 16

1. Составить схему алгоритма и программу для вычисления выражения:

$$t = \left(1 + \frac{x^2}{3 - \frac{x}{5}}\right) \frac{2 \cos\left(x - \frac{\pi}{6}\right)}{1,5 + \sin y^2}$$

2. В хозяйстве на уборке зерновых задействовано N комбайнов, имеющих производительность работы F (га/час) и M комбайнов с про-

изводительностью работы Q (га/час). Первому типу комбайнов предстоит убрать площадь S_1 (га), второму – S_2 (га). Часовой расход топлива при работе комбайнов составляет X_1 и X_2 (л/час) соответственно. Определить, сколько тонн топлива потребуется для уборки урожая, если удельный вес топлива составляет P (кг/л) и затраты хозяйства на топливо, если стоимость 1 тонны топлива составляет Z руб.

Вариант 17

1. Составить схему алгоритма и программу для вычисления выражения:

$$u = (\operatorname{tg}^2 z + 1)^x - \frac{\sqrt[3]{8 + |x - y|^2 + 1}}{x^2 + y^2 + 2}.$$

2. В хозяйстве рожью засеяна площадь A (га), пшеницей – B (га). Урожайность ржи составляет C (ц/га), пшеницы D – (ц/га). Для доставки зерна с поля на элеватор используется N автомобилей грузоподъемностью P_1 тонн и M автомобилей грузоподъемностью P_2 тонн. Продолжительность рабочего дня составляет T часов, на выполнение одного рейса (с поля на элеватор и обратно) каждому автомобилю для доставки зерна с поля требуется R часов. Определить, сколько дней потребуется для доставки зерна с поля имеющимися автомобилями. Указание: при расчете числа рейсов, которые выполняют один автомобиль за день, в программе предусмотреть округление результата до ближайшего меньшего целого числа.

Вариант 18

1. Составить схему алгоритма и программу для вычисления выражения:

$$t = \left(1 + \frac{z^2}{9 - \frac{z^2}{5}}\right) \frac{\cos\left(x - \frac{\pi}{6}\right)}{0,5 + \sin^2 y}.$$

2. Тепличное хозяйство построило N новых теплиц длиной C (м). Поперечное сечение теплиц имеет форму полукруга радиусом R (м). Один торец теплицы имеет входную дверь размером $A \times B$ (метров). Необходимо укрыть всю поверхность теплиц (за исключением дверей) полиэтиленовой пленкой. Определить необходимое количество рулонов пленки, если ширина пленки в рулоне – H (м), длина –

D (м). В программе результат расчетов округлить до ближайшего большего целого числа.

Вариант 19

1. Составить схему алгоритма и программу для вычисления выражения:

$$w = \left(1 + z + \frac{z^2}{2} + \frac{z^3}{3} + \frac{z^4}{4}\right) |\cos x - \cos y|.$$

2. На ручной уборке картофеля задействована группа из N студентов, причем ребята составляют $C\%$ от общего числа студентов, остальной состав группы - девушки. Скорость уборки картофеля ребятами за 1 час работы составляет Z м², а девушками - A (м²). Продолжительность рабочего дня студентов составляет M часов, из которых T часов используется на обеденный перерыв. Определить, сколько гектаров поля уборут студенты за K дней.

Вариант 20

1. Составить схему алгоритма и программу для вычисления выражения:

$$a = \frac{\sqrt{y + \sqrt[3]{x-1}}}{(\sin^2 z + 1)|x-y|}.$$

2. Участок имеет форму квадрата с диагональю L м. Участок по всему периметру необходимо огородить забором. Определить, какое количество столбиков и сколько штакетин необходимо заказать для забора, если расстояние между столбиками должно быть C м, а на один метр забора дается M штакетин. В программе результаты округлить до ближайшего целого числа в большую сторону.

Вариант 21

1. Составить схему алгоритма и программу для вычисления выражения:

$$B = (\sin^2 z - |x - y|) + \frac{z^3 + x}{\cos y - 1}.$$

2. Автохозяйство имеет N автомобилей. Среднедневной пробег одного автомобиля составляет L км. Норма расхода бензина для автомобилей составляет Q (литров на 100 км). Определить, на сколько дней хватит бензина, если закуплено M тонн топлива (вес одного литра бензина – R кг). В программе результат расчета округлить до ближайшего меньшего целого числа.

Вариант 22

1. Составить схему алгоритма и программу для вычисления выражения:

$$Q = \frac{1}{4} \cos(x) \frac{x + 3|x - y| + x^2}{|x - y|z + x^2} - 5 \operatorname{tg}(x).$$

2. В хозяйстве на уборке зерновых задействовано два комбайна, имеющие производительность работы F и X (га/час) соответственно. Первый комбайн убирает прямоугольное поле размером $A \times B$ (метров), площадь уборки для второго комбайна - в два раза больше. Часовой расход топлива при работе комбайнов составляет C и D (л/час) соответственно. Определить, сколько тонн топлива потребуется для уборки урожая, если удельный вес топлива составляет Q (кг/л), и затраты хозяйства на топливо, если стоимость 1 тонны топлива составляет W (руб.).

Вариант 23

1. Составить схему алгоритма и программу для вычисления выражения:

$$a = \sqrt[3]{z^6 + y} + \frac{|x - y|^{x+y}}{\operatorname{tg}(x) + \operatorname{tg}(z)}.$$

2. В хозяйстве рожью засеяна площадь X (га), пшеницей – на K % меньше. Урожайность ржи составляет Y (ц/га), пшеницы – на M % больше (ц/га). Для доставки зерна с поля на элеватор используются автомобили грузоподъемностью P тонн. Продолжительность рабочего дня составляет T часов, на выполнение одного рейса (с поля на элеватор и обратно) автомобилю требуется R часов. Определить, сколько автомобилей необходимо привлечь к работе, чтобы обеспечить доставку зерна с поля за N дней. Указание: при расчете числа

рейсов, которые выполняет один автомобиль за день, в программе предусмотреть округление результата до ближайшего меньшего целого числа.

Вариант 24

1. Составить схему алгоритма и программу для вычисления выражения:

$$F = \frac{\cos y - \frac{z}{y-x}}{1 - (y-x)^2} (y-x) + \sin^2 x.$$

2. Для уборки картофеля используется N комбайнов производительностью $Q1$ (г/час) и M комбайнов производительностью $Q2$ (га/час). Продолжительность рабочего дня комбайнов составляет T часов, время для ежедневного технического обслуживания каждого комбайна – E часов. Определить, какую площадь уборут комбайны за K дней, если все комбайны производительностью $Q2$ не работали два дня.

Вариант 25

1. Составить схему алгоритма и программу для вычисления выражения:

$$a = 2^{-x} + \frac{\sqrt{2x + \sin y}}{\cos^2 y + 3z}.$$

2. В хозяйстве M теплиц длиной Z (м). Поперечное сечение теплиц имеет форму полукруга радиусом R (м). Для укрытия всей поверхности теплиц (включая торцы) хозяйство закупило F рулонов полиэтиленовой пленки шириной A (м). Длина пленки в одном рулоне – Q (м). Однако оказалось, что закупленной пленки недостаточно. Определить, сколько теплиц будет укрыто и сколько рулонов пленки необходимо купить дополнительно, чтобы укрыть оставшиеся. В программе результаты расчетов округлить до целых чисел.

Вариант 26

1. Составить схему алгоритма и программу для вычисления выражения:

$$b = \frac{|x-y|}{e^{|x-y|} + \frac{x}{2}} \cos^3 y + y\sqrt{|x|}.$$

2. Картофельное поле разбито на прямоугольные участки размером $A \times B$ (метров). На ручной уборке картофеля задействована группа из Z студентов, причем ребята составляют 68 % от общего числа студентов, остальной состав группы – девушки. Скорость уборки картофеля ребятами за 1 час работы составляет V (m^2), а девушками – W (m^2). Продолжительность рабочего дня студентов составляет Q часов, из которых P часов используется на обеденный перерыв. Определить, сколько участков поля уборут студенты за K дней.

Вариант 27

1. Составить схему алгоритма и программу для вычисления выражения:

$$\tilde{n} = \frac{y(tgz - \frac{\pi}{6})}{|x| + \frac{1}{y^2 + 1}} - 3^x.$$

2. В хозяйстве K автомобилей. Среднедневной пробег одного автомобиля составляет W (км). Норма расхода бензина для автомобилей составляет A (литров на 100 км). Стоимость одного литра бензина – C (руб/л). Определить, на сколько дней хватит бензина, если на его покупку разрешено использовать денежную сумму в размере S рублей. В программе результат расчета округлить до ближайшего меньшего целого числа.

Вариант 28

1. Составить схему алгоритма и программу для вычисления выражения:

$$f = \frac{\sqrt[4]{y + \sqrt[3]{x-1}}}{(\sin^2 z + tgz)|x-y|}.$$

2. Фермер имеет в своем хозяйстве D коров и F телят. Суточное потребление сена на одну корову составляет K (кг), на одного теленка – T (кг). Планируемый срок зимовки скота составляет Z дней.

Сено на зиму фермер складывает в скирду, которая в поперечном сечении имеет форму полукруга диаметром H (м). Удельный вес сена в скирде составляет Q (кг/м³). Рассчитать длину стога, необходимого фермеру, чтобы прокормить свое хозяйство зимой.

Вариант 29

1. Составить схему алгоритма и программу для вычисления выражения:

$$g = \frac{x + \frac{y}{2}}{2|x + y|} (x + 1) + \frac{y^{x+1}}{3 + \sqrt[3]{|y - 2|}}.$$

2. В хозяйстве на уборке зерновых задействовано Z комбайнов, имеющих производительность работы Q (га/час), и M комбайнов с производительностью работы W (га/час). Первому типу комбайнов предстоит убрать площадь S (га), второму – на $K\%$ больше. Часовой расход топлива при работе комбайнов составляет D и G (л/час) соответственно. Определить, сколько тонн топлива потребуется для уборки урожая, если удельный вес топлива составляет P (кг/л), и затраты хозяйства на топливо, если стоимость 1 тонны топлива составляет T руб.

Вариант 30

1. Составить схему алгоритма и программу для вычисления выражения:

$$h = \frac{|y - x|^3}{3} - (1 + \sin x) \frac{1 + e^y}{1 + x|y - 1|}.$$

2. Для уборки урожая используется K комбайнов производительностью A (га/час) и P комбайнов производительностью B (га/час). Продолжительность рабочего дня комбайнов составляет Z часов, время для ежедневного технического обслуживания каждого комбайна – S часов. Определить, какую площадь уборут комбайны за K дней, если 1 комбайн производительностью A не работал два дня, а 1 комбайн производительностью B не работал три дня.

Вопросы для самоконтроля

1. Какие алгоритмы называют линейными?
2. Перечислите основные свойства алгоритма.
3. Какие алгоритмы называют разветвляющимися?
4. Что такое ветвь в блоке проверки условия?
5. Оператор условного перехода if, все варианты его использования.

Форма контроля – представить преподавателю отчет в письменном виде по изученному разделу.

МОДУЛЬ № 2

«Алгоритмы циклической структуры и их реализация на языке Turbo-Pascal. Операторы FOR, WHILE, REPEAT»

В результате изучения модуля студент должен:

- знать операторы ввода и вывода информации, операторы организации циклической структуры и оператор условного перехода;
- уметь составлять схемы алгоритмов и программы реализации алгоритмов циклической структуры на алгоритмическом языке Turbo-Pascal и проводить по ним расчет на компьютере;
- приобрести навыки самостоятельного выполнения индивидуальных заданий с проведением расчетов на компьютере.

НАУЧНО-ТЕОРЕТИЧЕСКОЕ СОДЕРЖАНИЕ МОДУЛЯ

СЛОВАРЬ ОСНОВНЫХ ПОНЯТИЙ

Новые понятия

Новое понятие	Определение
Цикл	Участок алгоритма, многократно повторяющийся для различных значений входящих в него величин
Параметр цикла	Величины, которые используются для подготовки очередного повторения цикла
Рабочая часть цикла (тело цикла)	Группа предписанных условий и действий, которая, повторяясь, дает искомый результат
Массив данных	Совокупность однотипных данных, имеющих общее имя и разные порядковые номера
Индекс (порядковый номер)	Указывает место (положение) элемента в массиве
Простые циклы	Циклы, в теле которых нет разветвлений и других встроенных в них циклов
Сложные циклы	Циклы, в теле которых имеются разветвления и другие встроенные в них циклы
Детерминированные	Циклы, в которых число повторений заранее известно из исходных данных или определено в ходе решения задачи
Итерационные	Циклы, в которых число повторений заранее не известно из исходных данных

Понятия для повторения

Понятие для повторения	Определение
Алгоритм	Под алгоритмом понимается «точное предписание, определяющее вычислительный процесс, ведущий от варьируемых начальных данных к искомому результату»
Алгоритмы разветвляющейся структуры	Алгоритмы, вычислительные процессы в которых в зависимости от выполнения некоторого логического условия производятся по одному из нескольких заранее определенных направлений

ОСНОВНОЙ ТЕКСТ

Лекция 1

Программирование циклических алгоритмов. Операторы цикла

План лекции:

- 1 Программирование циклических алгоритмов. Операторы цикла.
2. Оператор цикла FOR.
3. Оператор цикла WHILE.
4. Оператор цикла REPEAT.

В алгоритмическом языке Turbo-Pascal имеются три вида операторов цикла, которые используются для реализации алгоритмов циклической структуры:

- 1) оператор **FOR** – оператор цикла с параметром;
- 2) оператор **WHILE** – оператор цикла с предварительным условием;
- 3) оператор **REPEAT** – оператор цикла с последующим условием.

Оператор цикла FOR

Оператор цикла **FOR** используют для программирования детерминированных циклов, т. е. в том случае, когда заранее известно число повторений цикла, а параметр цикла либо возрастает с шагом 1, либо убывает с шагом -1 .

Общий вид оператора **FOR**:

FOR I:=N1 TO N2 DO <оператор цикла>;

или
FOR I:=N1 TO N2 DO BEGIN
 <блок операторов цикла>;
END;

где I – параметр цикла;

N1 – начальное значение параметра цикла;

N2 – конечное значение параметра цикла.

Параметры I, N1, N2 должны быть одного и того же скалярного типа, но не **REAL**. Параметр цикла I принимает последовательные значения от N1 до N2, увеличиваясь на 1. Схема, реализующая действие оператора **FOR** имеет вид, представленный на рисунке 1.

Например, при реализации следующего оператора:

FOR K:=1 TO 50 DO WRITELN (K:3);

на печать будут выданы натуральные числа от 1 до 50.

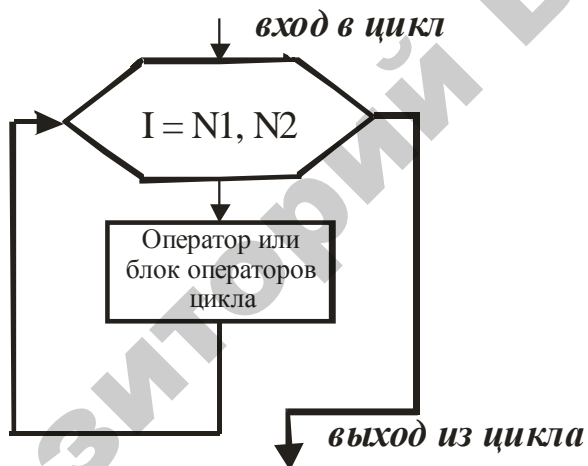


Рисунок 1 – Схема оператора цикла FOR

Если параметр цикла изменяется от большей величины к меньшей, уменьшаясь на 1 (т. е. шаг изменения параметра –1), то оператор цикла **FOR** будет иметь вид:

FOR I:=N1 DOWNTO N2 DO <оператор цикла>;

Например, при выполнении оператора:

FOR K:=100 DOWNTO 50 DO WRITELN (K:3);

на печать будут выданы натуральные числа от 100 до 50.

Пример 1 Вычислить значение факториала $P=n!$ при заданном n . Факториал n – это произведение первых n натуральных чисел, т. е. $P=1*2*3*…*n$.

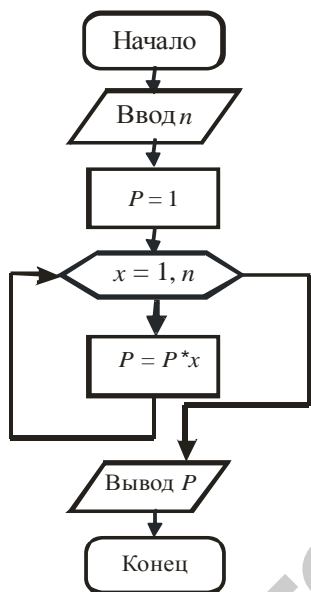


Рисунок 2 – Схема алгоритма решения примера 1

PROGRAM PRIMER1;

VAR X,N,P: INTEGER;

BEGIN

WRITELN ('ВВЕДИТЕ ЧИСЛО N');

READLN (N);

P:=1;

FOR X:=1 TO N DO P:=P*X;

WRITELN ('ФАКТОРИАЛ=',P);

END.

Оператор цикла WHILE

Операторы **WHILE** и **REPEAT** используются для программирования итерационных циклов, т. е. если необходимо произвести некоторые повторяющиеся вычисления, но число повторов заранее неизвестно, а также для программирования детерминированных циклов, если шаг изменения параметра цикла отличен от 1 и -1.

Общий вид оператора **WHILE**:

WHILE <логическое выражение> **DO** <оператор цикла>;

или

WHILE <логическое выражение> **DO BEGIN**

**<блок операторов цикла>;
END;**

Схема, реализующая оператор **WHILE**, является схемой итерационного цикла с предусловием и имеет следующий вид (рисунок 3):

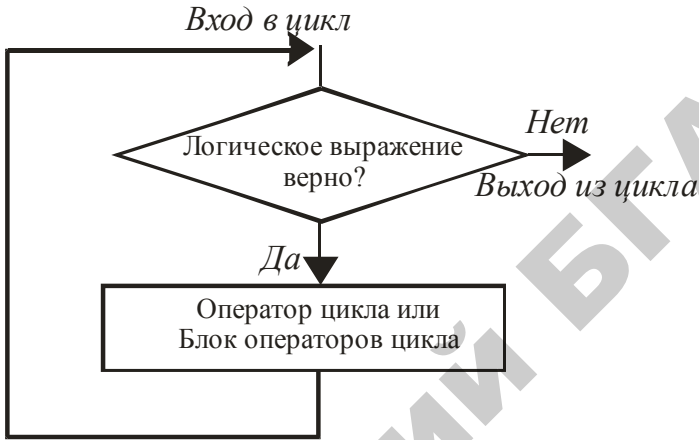
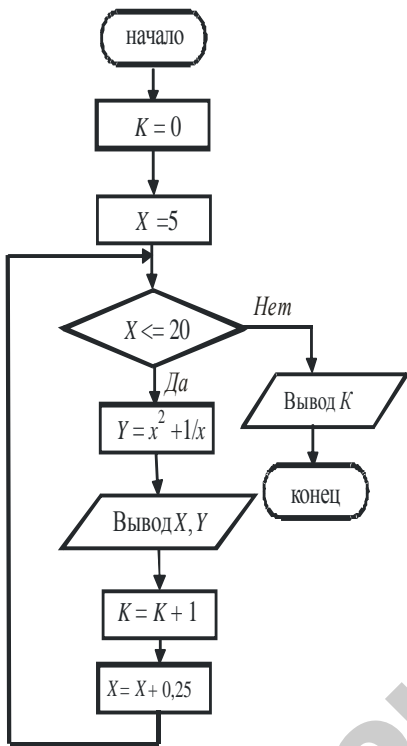


Рисунок 3 – Схема оператора цикла WHILE

Оператор цикла **WHILE** действует следующим образом: проверяется логическое выражение, и пока оно истинно (ветвь “Да”), выполняются все операторы цикла, записанные после слова **DO**. Как только логическое выражение становится ложным (ветвь “Нет”) происходит выход из цикла. Если с самого начала логическое условие ложно, то цикл не выполняется.

Пример 2 Вычислить и вывести на печать все значения аргумента x и функции $y = x^2 + 1/x$, при изменении x от 5 до 20 с шагом 0,25. Найти количество полученных значений y .



PROGRAM PRIMER2;

VAR K: INTEGER;

X,Y: REAL;

BEGIN

K:=0;

X:=5;

WHILE X<=20 DO BEGIN

Y:=X*X+1/X;

WRITELN (X:6:2,Y:12:6);

K:=K+1;

X:=X+0.25;

END;

WRITELN ('КОЛИЧЕСТВО=', K:3);

END.

Рисунок 4 – Схема алгоритма решения примера 2

Оператор цикла REPEAT

Общий вид оператора REPEAT:

REPEAT

<блок операторов цикла>;

UNTIL <логическое выражение>;

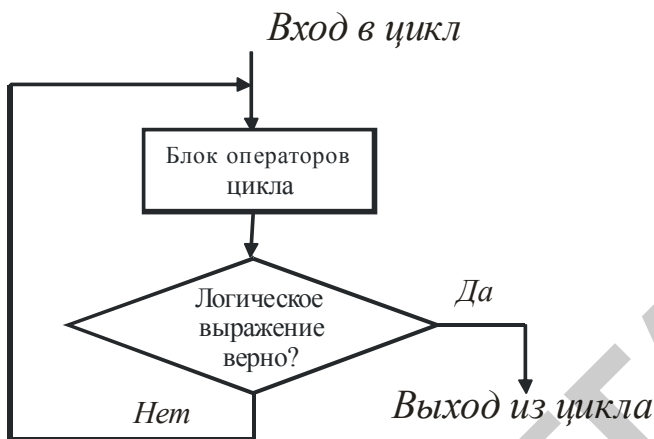
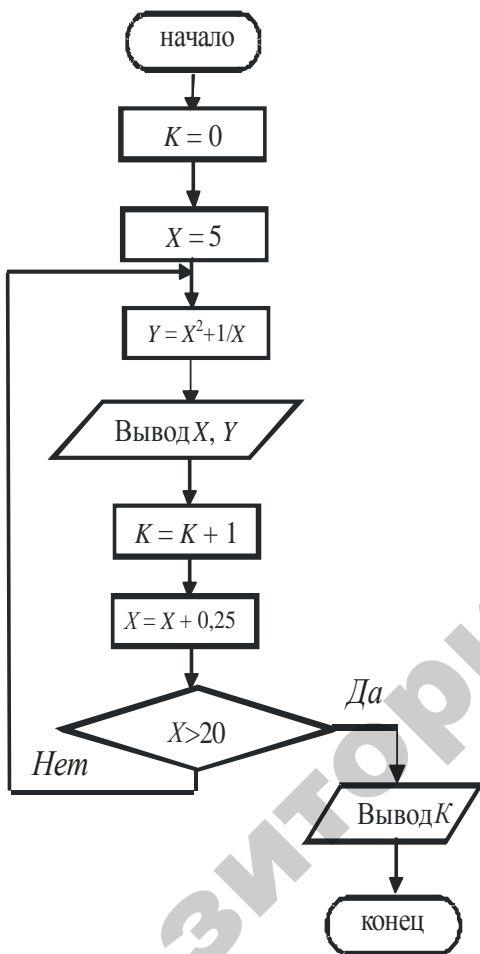


Рисунок 5 – Схема оператора REPEAT

Схема, реализующая оператор цикла **REPEAT**, является схемой итерационного цикла с постусловием и имеет вид, представленный на рисунке 5.

Оператор **REPEAT** действует следующим образом: выполняются все операторы циклической части, записанные между операторами **REPEAT** и **UNTIL**. Затем проверяется логическое выражение и если оно ложно (ветвь “Нет”), то вновь выполняются все операторы цикла до тех пор, когда логическое выражение станет истинным (ветвь “Да”). Если логическое выражение истинно с самого начала, то операторы циклической части выполняются один раз.

Используем оператор цикла **REPEAT** для решения примера 2. Тогда схема алгоритма и программа будут иметь следующий вид:



```

PROGRAM PRIMER2;
VAR K: INTEGER;
X,Y: REAL;
BEGIN
  K:=0;
  X:=5.;
  REPEAT
    Y:=X*X+1/X;
  WRITELN (X:6:2, Y:12:6);
  K:=K+1;
  X:=X+0.25;
  UNTIL X>20
  WRITELN('КОЛИЧЕСТВО=',K:3);
END.

```

Рисунок 6 – Схема алгоритма решения примера 2

Лекция 2

Программирование алгоритмов с использованием одномерных массивов данных

План лекции:

1. Массив данных. Объявление массива.
2. Программирование алгоритмов с использованием одномерных массивов данных.

Если переменная в каждый момент времени способна принимать лишь одно фиксированное значение, то ее называют *простой (скалярной)*. Переменные, состоящие из нескольких объединенных по какому-то признаку простых компонентов одного типа и имеющие общее имя, называют *сложными (структурированными)*. Примером структурированной переменной является **массив**.

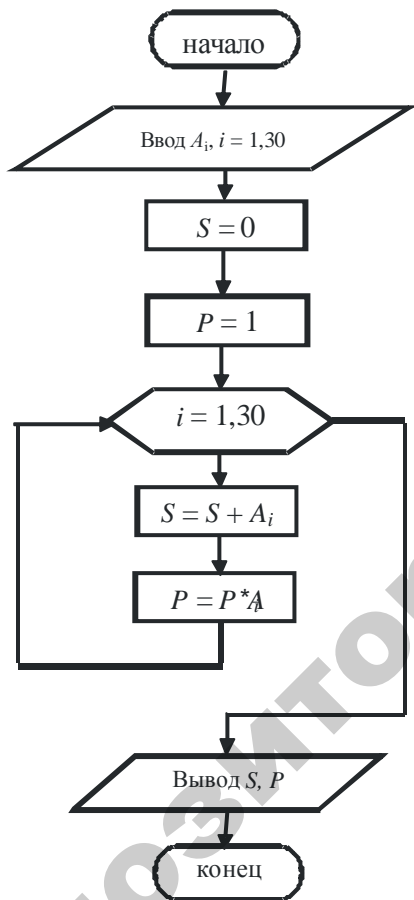
Массив (индексированные переменные) – подмножество однотипных переменных, объединенных по какому-либо признаку и имеющих общее имя. Отдельные величины, составляющие массив, называются его *элементами*. Имена массивов образуются по тем же правилам, что и имена простых переменных. Доступ к конкретному элементу массива осуществляется с помощью *индексов*, которые указывают в квадратных скобках после имени. Например: **Beta[5]** – пятый элемент массива **Beta**; **Delta[J]** – элемент массива **Delta** с порядковым номером **J**. Значение индекса есть целочисленная переменная (константа, выражение) или отдельные символы (тип **CHAR**). Присутствующие в программе элементы массивов объявляются в разделе описания переменных **VAR**, например:

```
VAR MAS1 : ARRAY[1..10] OF INTEGER;  
      MAS2 : ARRAY[1.. 8] OF REAL;
```

При объявлении массива вначале указывают его имя (например, **MAS1**), а затем, после двоеточия и слова **ARRAY**, интервал изменения индексов в квадратных скобках (в частности, для **MAS1** – от 1 до 10), ключевое слово **OF** и указатель типа элементов в массиве (в нашем примере **INTEGER** – **целый**).

Количество приводимых в квадратных скобках индексов, необходимых для установления месторасположения элементов массива, определяет его размерность. Для ввода и вывода элементов массива будем использовать оператор цикла **FOR**, в котором параметром цикла будет индекс (порядковый номер) элемента массива.

Пример 3 В заданном массиве A_1, A_2, \dots, A_{30} найти и вывести на печать сумму и произведение всех элементов массива.

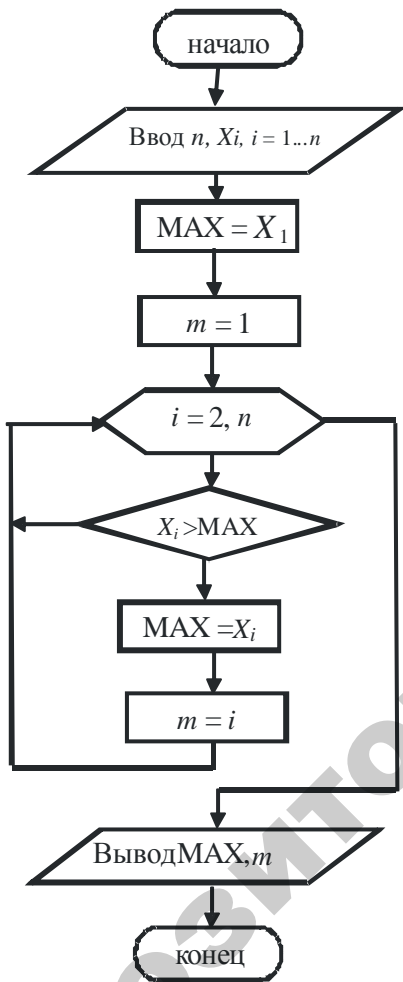


```

PROGRAM PRIMER3;
VAR S,P,I: INTEGER;
A: ARRAY [1..30] OF INTEGER;
BEGIN
FOR I:=1 TO 30 DO READLN (A[I]);
S:=0; P:=1;
FOR I:=1 TO 30 DO BEGIN
S:=S+A[I];
P:=P*A[I];
END;
WRITELN ('СУММА=',S:6,
'ПРОИЗВЕДЕНИЕ=',P:10);
END.
  
```

Рисунок 7 – Схема алгоритма примера 3

Пример 4 В массиве из n элементов (n задано): X_1, X_2, \dots, X_n не содержащем одинаковых элементов, найти максимальный элемент и его порядковый номер.



```

PROGRAM PRIMER4;
CONST N=20;
VAR I,MAX,M: INTEGER;
    X: ARRAY [1..N] OF INTEGER;
BEGIN
FOR I:=1 TO N DO READLN (X[I]);
MAX:=X[1];
M:=1;
FOR I:=2 TO N DO IF X[I]>MAX
THEN
BEGIN
MAX:=X[I];
M:=I;
END;
WRITELN('МАКСИМ.ЭЛЕМЕНТ=',
MAX:5,'ЕГО НОМЕР - ',M:2);
END.
  
```

Рисунок 8 – Схема алгоритма решения примера 4

Вопросы для самоконтроля

1. Что такое цикл, циклический вычислительный процесс?
2. Как классифицируются циклы?
3. Что такое параметр цикла?
4. Что такое массив данных?
5. Что такое индекс элемента массива и что он характеризует?
6. Какой цикл называется детерминированным?
7. В чем отличие операторов цикла WHILE и REPEAT?

МАТЕРИАЛЫ К ЛАБОРАТОРНОЙ РАБОТЕ

Тема: Алгоритмы циклической структуры.

Цель:

- изучить приемы составления схем алгоритмов циклической структуры, операторы организации циклов и условных переходов;
- уметь составлять программы реализации алгоритмов циклической структуры и проводить по ним расчет на компьютере.

При составлении алгоритмов решения задач возникает необходимость неоднократного повторения одних и тех же действий. Участок алгоритма, где многократно повторяются вычисления при различных значениях используемых в нем величин, называют циклом, а сам алгоритм, содержащий циклы - циклическим. Циклические алгоритмы позволяют существенно сократить объем программы за счет многократного выполнения группы повторяющихся вычислений. Специально изменяемый по заданному закону параметр, входящий в тело цикла, называется *переменной цикла*. Переменная цикла используется для подготовки очередного повторения цикла. В качестве переменной цикла могут использоваться индексы массивов, аргументы вычисляемых функций и другие величины. Во время выполнения тела цикла параметры переменной цикла изменяются с заданным шагом. Следовательно, при организации циклических вычислений необходимо предусмотреть задание начального значения переменной цикла, закона ее изменения и проверку на окончание цикла, при выполнении которой произойдет завершение цикла. Циклы, в теле которых нет разветвлений и других встроенных в них циклов, называют *простыми*. В противном случае их относят к *сложным*.

Циклические алгоритмы разделяют на *детерминированные* и *итерационные*.

Циклы, в которых число повторений заранее известно из исходных данных или определено в ходе решения задачи, называют *детерминированными*. Для организации детерминированных циклов наиболее целесообразно использовать блок модификации, внутри которого указывается переменная цикла, ее начальное и конечное значения, а также шаг ее изменения (если шаг изменения равен 1, то его допускается не указывать). Организовать подобный цикл возможно и при использовании блока проверки условия вместо блока модификации, однако при этом несколько усложняется алгоритм и теряется его рациональность.

Студенту следует для трех задач своего варианта, условия которых приведены ниже, выполнить следующие задания:

- составить схему алгоритма задачи;
- записать пояснения к схеме алгоритма;
- составить программу на языке Turbo-Pascal;
- проверить выполнение составленного алгоритма на контрольном примере, приняв упрощенные значения исходных данных по своему усмотрению.

ПРИМЕРЫ РАЗНОУРОВНЕВЫХ ЗАДАНИЙ ДЛЯ КОНТРОЛЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО МОДУЛЮ

I УРОВЕНЬ

Вариант 1

1. Вычислить сумму ряда:

$$S = 1 + 2 + 3 + 4 + \dots + 20.$$

2. Для массива $a_1, a_2, a_3, \dots, a_{10}$ получить среднее арифметическое его положительных элементов.

3. Напечатать таблицу перевода температуры из градусов по шкале Цельсия (C) в градусы по шкале Фаренгейта (F) для значений T от 15°C до 30°C с шагом 1°C . (Перевод осуществляется по формуле $F = 1,8^\circ\text{C} \times 32^\circ\text{C}$).

Вариант 2

1. Вычислить сумму ряда:

$$S = 1^2 + 2^2 + 3^2 + 4^2 + \dots + 10^2.$$

2. Для массива $x_1, x_2, x_3, \dots, x_{10}$ получить сумму и произведение всех отрицательных элементов.

3. За сезон уборки каждым из 10 комбайнов убрано соответственно p_1, p_2, \dots, p_{10} гектаров поля. Определить, сколько комбайнов достигли плановой наработки A гектаров. Искомое количество напечатать.

Вариант 3

1. Вычислить сумму ряда:

$$S = \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots + \frac{1}{10}.$$

2. Для массива $a_1, a_2, a_3, \dots, a_{15}$ получить среднее арифметическое его отрицательных элементов.

3. В массиве $a_1, a_2, a_3, \dots, a_{10}$ определить число элементов меньших 5, и число элементов, больших среднего арифметического элементов заданного массива.

Вариант 4

1. Вычислить сумму ряда:

$$S = \frac{1}{\cos 1} + \frac{1}{\cos 2} + \frac{1}{\cos 3} + \dots + \frac{1}{\cos 8}.$$

2. Для массива $a_1, a_2, a_3, \dots, a_{15}$ определить количество его отрицательных элементов и сумму положительных элементов.

3. Заданы координаты 8 точек $(x_1, y_1), (x_2, y_2), \dots, (x_8, y_8)$. Вывести на печать координаты точек, лежащих на оси абсцисс и номера этих точек.

Вариант 5

1. Вычислить произведение:

$$S = 1 \times 2 \times 3 \times 4 \times \dots \times 9.$$

2. Для массива $a_1, a_2, a_3, \dots, a_{10}$ получить сумму его отрицательных элементов и произведение положительных.

3. Имеется 7 точек с координатами $(x_1, y_1), (x_2, y_2), \dots, (x_7, y_7)$. Определить количество точек, лежащих во 2-ой четверти.

II УРОВЕНЬ

Вариант 6

1. Вычислить сумму ряда:

$$S = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \dots + \frac{1}{n^2} \quad (n \text{ задано}).$$

2. В массиве $a_1, a_2, a_3, \dots, a_{10}$ найти сумму положительных элементов, произведение отрицательных элементов и количество элементов, равных нулю.

3. Имеется две группы комбайнов: 8 комбайнов ККУ-2А и 6 комбайнов – КПК-3. Комбайнами I группы выкопано соответственно p_1, p_2, \dots, p_8 тонн картофеля; II – r_1, r_2, \dots, r_6 тонн. Напечатать и сравнить средние производительности комбайнов, сделать вывод, какой тип комбайнов лучше.

Вариант 7

1. Вычислить сумму ряда:

$$S = \sum_{N=1}^{20} \frac{N^3}{N+1}.$$

2. Для массива $a_1, a_2, a_3, \dots, a_{15}$ получить количество его положительных элементов и вывести на печать номера отрицательных элементов.

3. За сезон уборки каждым из 8 комбайнов убрано соответственно s_1, s_2, \dots, s_8 га поля. Определить, достигает ли средняя наработка комбайнов плановой величины P га и выдать соответствующее сообщение.

Вариант 8

1. Вычислить:

$$S = \frac{\sum_{i=1}^{10} Y_i^2}{\sum_{i=1}^8 (X_i^2 + 2)}.$$

2. Для массива $a_1, a_2, a_3, \dots, a_{15}$ получить сумму его положительных элементов и количество отрицательных элементов.

3. Имеется 10 клубней I сорта весом p_1, p_2, \dots, p_{10} и 8 клубней II сорта весом r_1, r_2, \dots, r_8 . Определить, клубни какого сорта в среднем тяжелее.

Вариант 9

1. Вычислить:

$$S = \sum_{i=1}^5 (A_i + B_i)^2.$$

2. Получить произведение тех элементов массива c_1, c_2, \dots, c_{10} , которые превышают заданное число d и сумму тех элементов, которые меньше или равны d .

3. Даны номера выигрышных билетов p_1, p_2, \dots, p_{10} . Определить, является ли билет с номером r выигрышным, и напечатать соответствующее сообщение.

Вариант 10

1. Вычислить:

$$S = \sum_{X=1}^{25} \frac{X^2}{X^3 + 1}.$$

2. Вычислить сумму квадратов первых n натуральных чисел (n задано).

3. За сезон уборки каждым из 10 комбайнов убрано соответственно p_1, p_2, \dots, p_{10} гектаров поля. Определить, сколько комбайнов не достигли плановой наработки A га, и вывести на печать номера этих комбайнов.

Вариант 11

1. Вычислить сумму:

$$S = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \dots + \frac{1}{8^2}.$$

2. Даны действительные числа a_1, a_2, \dots, a_{10} . Все отрицательные элементы массива заменить их квадратами и подсчитать сумму всех положительных элементов полученного массива. Преобразованный массив вывести на печать.

3. Имеются сведения о выработке электроэнергии одной из турбин электростанции по месяцам с января по декабрь R_1, R_2, \dots, R_{12} . Определить среднемесячную выработку электроэнергии турбиной и номер месяца, в который выработали наибольшее количество электроэнергии.

Вариант 12

1. Вычислить:

$$S = \sum_{i=1}^7 (D_i^2) - \sum_{i=1}^9 (Z_i^3).$$

2. Вычислить произведение первых 15 натуральных чисел.

3. Даны координаты 7 точек $(x_1, y_1), (x_2, y_2) \dots (x_7, y_7)$. Вывести на экран координаты точки, расположенной ближе всех к началу координат.

Вариант 13

1. Вычислить:

$$S = \frac{\sum_{i=1}^5 Y_i^3}{\sum_{i=1}^8 (X_i^2 + 4)}.$$

2. В массиве $x_1, x_2, x_3 \dots x_{10}$ найти число элементов, превышающих заданную величину V , и произведение всех элементов массива.

3. Среди точек с координатами $(x_1, y_1), (x_2, y_2) \dots (x_{10}, y_{10})$ найти количество и номера точек, лежащих в 3-ей четверти.

Вариант 14

1. Вычислить сумму ряда (n – задано):

$$S = \frac{1}{1 \times 2} + \frac{1}{2 \times 3} + \frac{1}{3 \times 4} + \dots + \frac{1}{n(n+1)}.$$

2. В массиве $a_1, a_2, a_3 \dots a_{10}$ вместо отрицательных элементов записать их модули и вывести на экран полученный массив. Найти сумму положительных элементов массива.

3. Имеется 10 результатов по прыжкам в длину c_1, c_2, \dots, c_{10} . Вывести на печать номера спортсменов, чей результат превышает L метров.

Вариант 15

1. Вычислить:

$$S = \sum_{i=1}^{10} (X_i) + \frac{1}{\sum_{i=1}^{10} (X_i + 4)} - C.$$

2. Среди точек с координатами $(x_1, y_1), (x_2, y_2) \dots (x_{10}, y_{10})$ найти количество точек, лежащих в 1-ой четверти и вывести на печать их номера.

3. Для массива $a_1, a_2, a_3 \dots a_{15}$ найти разность между средним арифметическим элементов массива и наименьшим из них.

Вариант 16

1. Вычислить:

$$A = \sum_{i=1}^7 \frac{X_i^2 + \cos(Y_i)}{i^2 + 1}.$$

2. В массиве $a_1, a_2, a_3, \dots, a_{10}$ найти минимальный элемент и его порядковый номер (в массиве нет совпадающих элементов) и поменять его местами с первым элементом массива. Преобразованный массив вывести на печать.

3. Даны координаты 10 точек $(x_1, y_1), (x_2, y_2) \dots (x_{10}, y_{10})$. Определить количество точек, лежащих в 4-ой четверти, если таких точек нет – выдать соответствующее сообщение.

Вариант 17

1. Вычислить:

$$S = \sum_{i=1}^{10} \frac{(\sin(X_i) - 1)^2}{i^2}.$$

2. В массиве $a_1, a_2, a_3 \dots a_{10}$ заменить на 5 все элементы, равные и большие среднего арифметического, и на 0 – меньшие среднего

арифметического всех элементов массива. Вывести на экран полученный массив.

3. Имеется 10 клубней шарообразной формы с диаметрами $a_1, a_2, a_3, \dots, a_{10}$. Определить, сколько из них пройдет сквозь круглое отверстие сепарирующего решета площадью S .

Вариант 18

1. Вычислить:

$$S = \frac{\sum_{i=1}^{10} X_i^2 - \sum_{i=1}^{10} (X_i Y_i)}{4}.$$

2. В массиве $a_1, a_2, a_3 \dots a_{10}$ найти максимальный элемент и его порядковый номер (массив не содержит совпадающих элементов) и поменять его местами с последним элементом массива. Преобразованный массив вывести на печать.

3. В районе 10 колхозов. В каждый из них должно быть поставлено $a_1, a_2, a_3 \dots a_{10}$ тракторов, а фактически поставлено $b_1, b_2, b_3 \dots b_{10}$ тракторов. Определить, сколько колхозов недополучили трактора. Вывести на печать номера колхозов, получивших трактора сверх плана.

Вариант 19

1. Вычислить:

$$S = \sum_{i=1}^{10} (X_i) + \sum_{i=1}^{12} (Z_i^3) + c^2, \quad (c - \text{задано}).$$

2. Определить разность между наибольшим и наименьшим элементами массива $a_1, a_2 \dots a_{10}$.

3. Дан список 5 студентов и отметки каждого из них за выполнение двух контрольных работ соответственно $a_1, a_2 \dots a_5$ и $b_1, b_2, \dots b_5$. Подсчитать число студентов, выполнивших обе работы на 9, если таких нет – вывести сообщение об этом.

Вариант 20

1. Вычислить:

$$S = \sum_{i=1}^{10} X_i^2 - \sum_{i=1}^7 Z_i.$$

2. Дан массив x_1, x_2, \dots, x_{15} . Все отрицательные элементы массива заменить на 1, а положительные элементы массива заменить на 5. Вывести на печать полученный массив.

3. Имеется список 15 членов бригады с указанием их возраста. Определить средний возраст членов бригады и порядковые номера работников, чей возраст больше 50 лет.

Вариант 21

1. Вычислить:

$$S = \sum_{i=1}^{10} X_i \sum_{i=1}^{10} Y_i - 10 \sum_{i=1}^{10} (X_i Y_i).$$

2. В массиве $a_1, a_2, a_3 \dots a_{15}$ подсчитать количество элементов, больших среднего арифметического, и количество меньших и равных среднему арифметическому элементов массива.

3. Заданы координаты 12 точек $(x_1, y_1), (x_2, y_2), \dots, (x_{12}, y_{12})$. Вывести на печать номера точек, лежащих в I или II четвертях.

Вариант 22

1. Вычислить:

$$S = \frac{\sum_{i=1}^{12} (1 - \cos X_i)^2}{i^2 - 1} + R, \quad (R - \text{задано}).$$

2. Задан массив x_1, x_2, \dots, x_{15} . Определить количество и сумму элементов массива, принадлежащих отрезку $[a, b]$. Величины a, b заданы, причем $a < b$. Если таких элементов нет – вывести на печать соответствующее сообщение.

3. Имеется список 10 членов колхоза с указанием возраста: a_1, a_2, \dots, a_{10} . Определить средний и максимальный возраст колхозников.

Вариант 23

1. Вычислить:

$$S = \sum_{i=1}^{10} (X_i^2) + \frac{1}{\sum_{i=1}^6 (Y_i + 4)} + d, \quad (d - \text{задано}).$$

2. Даны координаты 10 точек: $(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots (x_{10}, y_{10})$. Определить, сколько точек попадает в кольцо с внутренним радиусом r_1 и внешним r_2 , если центр кольца находится в начале координат.

3. Имеется список 12 членов бригады с указанием их возраста. Определить средний возраст и вывести порядковые номера членов бригады, возраст которых превышает средний.

Вариант 24

1. Вычислить:

$$S = \sum_{i=1}^7 \frac{(\cos(Y_i) - f)^2}{i^2}, f - \text{задано.}$$

2. В массиве $a_1, a_2, \dots a_{10}$ найти среднее арифметическое отрицательных элементов массива и количество положительных элементов массива.

3. Имеются сведения о количестве тракторов, которые должны быть поставлены по плану каждому из 10 колхозов $a_1, a_2, \dots a_{10}$. Также имеются сведения о фактической поставке тракторов этим колхозам $b_1, b_2, \dots b_{10}$. Определить скольким колхозам осуществили неполный объем поставки тракторов и напечатать номера этих колхозов.

III УРОВЕНЬ

Вариант 25

1. Вычислить:

$$S = \frac{\sum_{i=1}^8 X_i^2 - \sum_{i=1}^8 (X_i Y_i)}{4 \sum_{i=1}^8 X_i}.$$

2. Дан массив $a_1, a_2, \dots a_8$. Получить и вывести на печать число неотрицательных элементов массива и номера отрицательных.

3. Даны координаты 10 точек $(x_1, y_1), (x_2, y_2), \dots (x_{10}, y_{10})$. Вывести на печать координаты точки, расположенной дальше всех от начала координат.

Вариант 26

1. Вычислить:

$$S = \sum_{i=1}^5 X_i \sum_{i=1}^{10} Y_i^2 + K \sum_{i=1}^5 (X_i), (K - \text{задано}).$$

2. Заданы координаты 10 точек $(x_1, y_1), (x_2, y_2) \dots (x_{10}, y_{10})$. Вывести на печать координаты точек, лежащих на оси ординат и количество точек на оси абсцисс.

3. Пассажирский самолет может поднять груз общим весом R кг. Составить программу определения веса почтового груза, который можно поместить в самолет после посадки N пассажиров (условный вес одного человека 70 кг) и загрузки их багажа, составляющего P_1, P_2, \dots, P_N кг.

Вариант 27

1. Вычислить:

$$S = \sum_{i=1}^{10} (X_i) + \sum_{i=1}^7 (Z_i) - \sum_{i=1}^{10} D_i .$$

2. Дан массива a_1, a_2, \dots, a_{10} . Все неотрицательные элементы массива заменить на 1 и получить число неотрицательных элементов массива, а также вывести на печать полученный массив.

3. Даны два массива C_1, C_2, \dots, C_7 и D_1, D_2, \dots, D_7 . Все элементы массивов отличны от нуля. Подсчитать сумму положительных элементов массива R_1, R_2, \dots, R_7 , каждый элемент которого: $R_i = C_i / D_i, i = 1, 2, \dots, 7$.

Вариант 28

1. Вычислить:

$$S = \frac{\sum_{i=1}^5 X_i^2 - 10}{\sum_{i=1}^7 Y_i} .$$

2. Для массива $a_1, a_2, a_3, \dots, a_{10}$ получить среднее арифметическое его отрицательных элементов и произведение положительных элементов.

3. Получить сумму и номера тех элементов массива C_1, C_2, \dots, C_{10} , которые превышают заданное число D , если таких элементов нет – вывести на печать соответствующее сообщение.

Вариант 29

1. Вычислить:

$$S = \sum_{i=1}^7 (X_i^2) + \sum_{i=1}^{10} (Z_i).$$

2. Дан массив a_1, a_2, \dots, a_{12} и величины c, d ($c < d$). Все элементы массива, принадлежащие отрезку $[c, d]$, распечатать и заменить нулями. Полученный массив вывести на печать. Если указанных элементов нет, напечатать соответствующее сообщение.

3. В массиве b_1, b_2, \dots, b_{12} поменять местами первый и последний элементы массива. Полученный массив вывести на печать.

Вариант 30

1. Вычислить

$$S = \sum_{i=1}^{10} X_i \sum_{i=1}^{15} Y_i^2 - K \sum_{i=1}^{10} (X_i), \quad (K - \text{задано}).$$

2. Для массива $x_1, x_2, x_3, \dots, x_{15}$ найти сумму положительных элементов, количество элементов равных 0, а также вывести на экран номера отрицательных элементов данного массива.

3. Заданы координаты 10 точек $(x_1, y_1), (x_2, y_2), \dots, (x_{10}, y_{10})$. Вывести на печать: координаты точек, которые попали в круг радиуса R с центром в начале координат, и номера точек, не попавших в круг.

ЗАДАНИЯ ДЛЯ УПРАВЛЯЕМОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Самостоятельно изучить и выполнить раздел курса «Программирование алгоритмов циклической структуры с использованием операторов **WHILE** и **REPEAT**» по следующему плану.

1. Составление схем алгоритмов решения задачи (используя графический способ записи) для:
- оператора цикла **REPEAT**,
 - оператора цикла **WHILE**.

2. Составление программ на языке Turbo-Pascal.
3. Проверка выполнения составленных программ на контрольном примере, используя значения исходных данных из таблицы № 1.

Вычислить и вывести на печать значения функции f на отрезке $[a, b]$ с постоянным шагом h , при разбиении отрезка на n равных частей. Исходные данные приведены в таблице 1.

Таблица 1 – Варианты заданий

№ варианта	Функция f	Границы отрезка $[a, b]$	Кол-во частей разбиения, n
1	$\sqrt[3]{\cos x} + (1+x)^2$	[1; 5]	7
2	$2\sin x + \frac{1}{x}$	[1; 5]	5
3	$\sqrt{\frac{1}{x}} + \cos x$	[2; 4]	6
4	$x^3 + \sin x$	[0; 4]	5
5	$x^2 + 2^x$	[0; 3]	4
6	$\sin x^2 + \frac{x}{2}$	$[\pi/2; 2\pi]$	5
7	$\cos x^3 + (x+1)^2$	$[0; \pi]$	6
8	$\cos x + 2^x$	[1; 4]	4
9	$\operatorname{tg}(x+1)$	[1; 3]	5
10	$\sin\left(\frac{x}{2}\right)$	[0; 1]	6
11	$\sin^2 x + 1$	[2; 5]	4
12	$\cos x - 10x$	$[0; \pi/2]$	5
13	$\sin x - \cos x$	[1; 3,6]	6
14	$\sin x^2 + \cos^2 x$	[1; 5]	7

Окончание таблицы 1

№ варианта	Функция f	Границы отрезка [a, b]	Кол-во частей разбиения, n
15	$1 - \sin x$	[2; 4]	5
16	$x - \cos x$	[1; 3]	6
17	$3 + \cos x$	$[\pi/2; \pi]$	5
18	$x^3 + \frac{1}{x}$	[1; 4]	7
19	$x \sin x$	[1; 4]	5
20	$x - \cos x$	[0,5; 2,7]	6
21	$\sin^2 x - 5$	[1; 10]	6
22	$x^2 + \sin x^2$	[0, π]	7
23	$\frac{1}{x} - x^3$	[1; 6]	5
24	$5 - \cos x$	$[0; \pi/2]$	8
25	$2x + \cos x$	[0; 10]	6
26	$\cos^2 \frac{x}{2} + 3,5$	[1, 8]	6
27	$\sin x + 2x$	[0, 10]	7
28	$\cos^2 x - 4,2x$	[1, 8]	6
29	$x \cos x$	[1, 5]	6
30	$2,5x + e^x$	[1, 6]	5

Форма контроля – представить преподавателю отчет в письменном виде по изученному разделу.

МОДУЛЬ № 3

«Работа с двумерными массивами (матрицами)»

В результате изучения модуля студент должен:

- знать типовые структуры алгоритмов для решения задач, содержащих двумерные массивы данных;
- уметь составлять схемы алгоритмов и программы на языке Turbo-Pascal для решения практических задач с использованием матриц;
- анализировать возможность применения определенной типовой структуры алгоритма к решению конкретной научно-ориентированной задачи.

НАУЧНО-ТЕОРЕТИЧЕСКОЕ СОДЕРЖАНИЕ МОДУЛЯ

СЛОВАРЬ ОСНОВНЫХ ПОНЯТИЙ

Новые понятия

Новое понятие	Определение
Двумерный массив (матрица)	Представляет собой таблицу, заполненную элементами с двумя индексами. Первый индекс указывает номер строки, второй индекс – номер столбца, на пересечении которых находится элемент
Вложенные циклы	Циклы, которые имеют внешний и внутренний циклы
Квадратная матрица	Матрица называется квадратной, если количество строк соответствует количеству столбцов

Понятия для повторения

Понятие для повторения	Определение
Массив данных	Совокупность однотипных данных, имеющих общее имя и разные порядковые номера
Индекс (порядковый номер)	Указывает место (положение) элемента в массиве
Сложные циклы	Циклы, в теле которых имеются разветвления и другие встроенные в них циклы

ОСНОВНОЙ ТЕКСТ

Лекция 1

Программирование алгоритмов с использованием двумерных массивов данных (матриц)

План лекции:

1. Составление алгоритмов с использованием двумерных массивов данных (матриц) и вложенных циклов.
2. Программирование алгоритмов с использованием двумерных массивов данных (матриц).
3. Квадратные матрицы.

В двумерном массиве (матрице) положение каждого элемента определяется двумя индексами, разделенными запятой, например: $A[i,j]$, $B[3,7]$, $C[k,l]$.

Первый из индексов всегда обозначает номер строки, а второй - номер столбца матрицы, на пересечении которых находится данный элемент.

Все элементы матриц, встречающихся в программе, должны быть объявлены в разделе **VAR**, например:

VAR

I, J: INTEGER;

A: ARRAY [1..10, 1..15] OF INTEGER;

X: ARRAY [1..8, 1..3] OF REAL;

Ввод и вывод элементов матриц отличается более сложной структурой, так как требует последовательного изменения двух переменных циклов: номера строки и номера столбца. Для организации ввода-вывода матриц используют два вложенных друг в друга цикла **FOR**, какой из них будет внешним, а какой внутренним (вложенным) – зависит от условия решаемой задачи.

Пусть задана матрица $a(3,4)$, где $a[i,j]$ – элементы матрицы, стоящие на пересечении строки с номером i и столбца с номером j .

$a_{1,1}$ $a_{1,2}$ $a_{1,3}$ $a_{1,4}$

$a_{2,1}$ $a_{2,2}$ $a_{2,3}$ $a_{2,4}$

$a_{3,1}$ $a_{3,2}$ $a_{3,3}$ $a_{3,4}$

Ввод элементов матрицы по строкам:

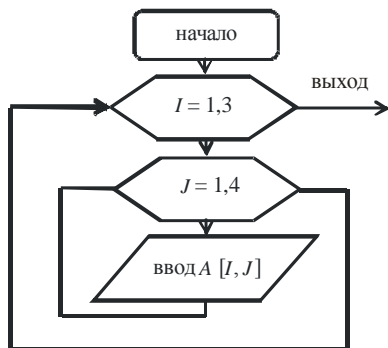


Рисунок 1 – Ввод элементов матрицы по строкам

Такие циклы называются вложенными. Причем в данном случае цикл по переменной i – называется внешним, цикл по j – внутренним. Пока переменная i изменяется на 1 шаг, переменная j проходит все значения от начального до конечного.

Ввод элементов матрицы по столбцам:

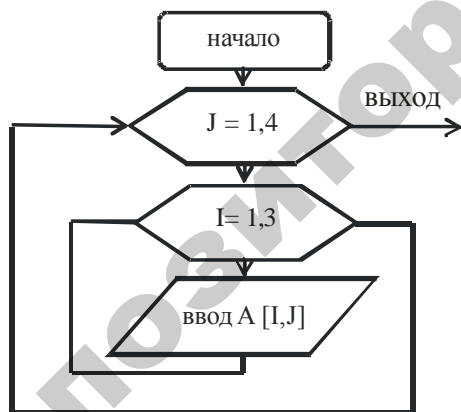


Рисунок 2 – Ввод элементов матрицы по столбцам

```

PROGRAM VVOD;
VAR I,J: INTEGER;
A: ARRAY [1..3,1..4] OF NTEGER;
BEGIN
FOR I:= 1 TO 3 DO
FOR J:= 1 TO 4 DO READLN (A[I,J] );
  
```

```

PROGRAM VVOD;
VAR I,J: INTEGER;
A: ARRAY [1..3,1..4] OF
NTEGER;
BEGIN
FOR J:= 1 TO 4 DO
FOR I:= 1 TO 3 DO
READLN(A[I,J]);
  
```

В этом случае внешним является цикл по переменной j , а внутренним – по переменной i .

Пример 1 Задана матрица из семи строк и пяти столбцов. Найти среднее арифметическое отрицательных элементов матрицы **a(7,5)**:

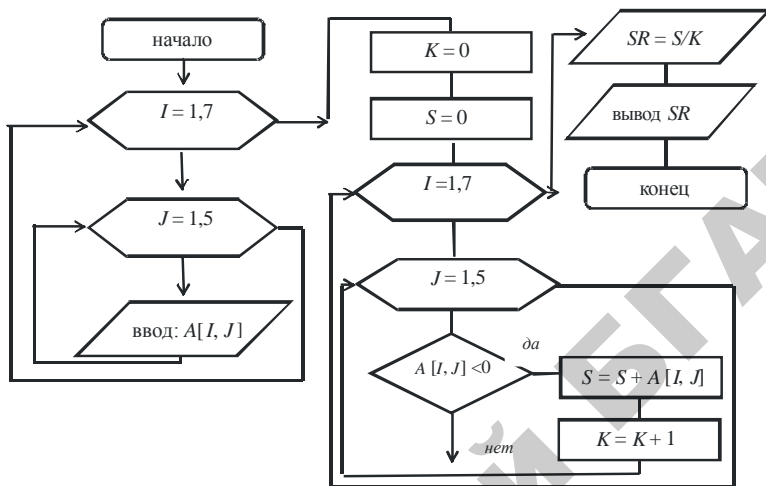


Рисунок 3 – Схема алгоритма к решению примера 1

Программа на языке Turbo-Pascal к примеру 1.

PROGRAM PRIM1;

VAR

I,J,K: INTEGER;

SR,S: REAL;

A: ARRAY [1..7,1..5] OF REAL;

BEGIN

FOR J:= 1 TO 7 DO

FOR I:= 1 TO 5 DO READLN(A[I,J]);

K:=0;

S:=0;

FOR I:= 1 TO 7 DO

FOR J:= 1 TO 5 DO IF A[I,J]<0 THEN BEGIN

S:= S+A[I,J];

K:= K+1;

END;

SR = S/K;

WRITELN ('среднее арифметическое', SR:10:3);

END.

Квадратные матрицы

Матрица называется квадратной, если количество строк соответствует количеству столбцов. Элементы, стоящие на главной диагонали матрицы, имеют одинаковые номера строки и столбца.

Пример 2 Задана квадратная матрица a (5,5). Найти сумму элементов матрицы, стоящих на главной диагонали.

$a_{1,1}$ $a_{1,2}$ $a_{1,3}$ $a_{1,4}$ $a_{1,5}$
 $a_{2,1}$ $a_{2,2}$ $a_{2,3}$ $a_{2,4}$ $a_{2,5}$
 $a_{3,1}$ $a_{3,2}$ $a_{3,3}$ $a_{3,4}$ $a_{3,5}$
 $a_{4,1}$ $a_{4,2}$ $a_{4,3}$ $a_{4,4}$ $a_{4,5}$
 $a_{5,1}$ $a_{5,2}$ $a_{5,3}$ $a_{5,4}$ $a_{5,5}$.

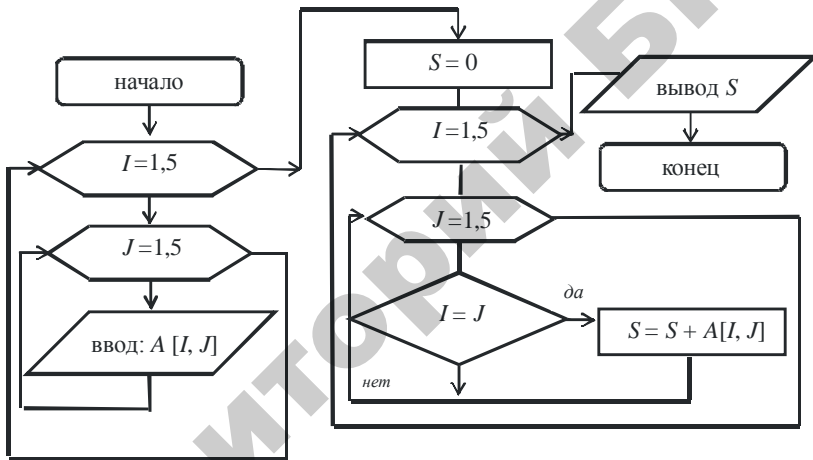


Рисунок 4 – Схема алгоритма к решению примера 2

Программа на языке Turbo-Pascal к примеру 2.

```
PROGRAM PRIM2;
VAR
  I,J: INTEGER;
  S: REAL;
  A: ARRAY [1..5,1..5] OF REAL;
BEGIN
  FOR J:= 1 TO 5 DO
  FOR I:= 1 TO 5 DO READLN(A[I,J]);
  S:=0;
```

```

FOR I:= 1 TO 5 DO
FOR J:= 1 TO 5 DO IF I=J THEN S:=S+A[I,J];
WRITELN ('СУММА', S:10:3);
END.

```

Заметим, что у всех элементов, стоящих над главной диагональю, $i < j$ (верхний треугольник), под главной диагональю $i > j$ (нижний треугольник).

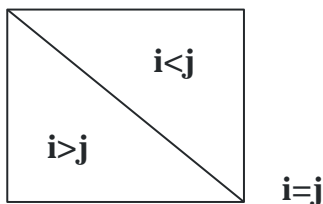


Рисунок 5 – Соотношение номера строки i и номера столбца j в квадратной матрице

Вопросы для самоконтроля

1. Какие циклы называются вложенными?
2. Какие массивы называются двумерными (матрицами)?
3. Ввод элементов матрицы по столбцам.
4. Ввод элементов матрицы по строкам.
5. Вывод элементов матрицы по строкам и столбцам.
6. Нахождение максимального элемента матрицы (номер строки и номер столбца, на пересечении которых он находится).
7. Что означают индексы i и j в обозначении элемента матрицы $a[i,j]$?
8. Пояснить понятия «главная диагональ», «верхний и нижний треугольники» в квадратной матрице.

МАТЕРИАЛЫ К ЛАБОРАТОРНОЙ РАБОТЕ

Тема: Программирование алгоритмов с использованием двумерных массивов данных (матриц).

Цель:

- приобрести навыки разработки алгоритмов циклических вычислительных процессов сложной структуры на примере матриц.
- уметь составлять программы с вложенными циклами на алгоритмическом языке Turbo-Pascal и проводить по ним расчет на компьютере.

Понятие вложенного цикла тесно связано с понятием внешнего и внутреннего цикла. Схема алгоритма с одним вложенным циклом приведена на рисунке 6.

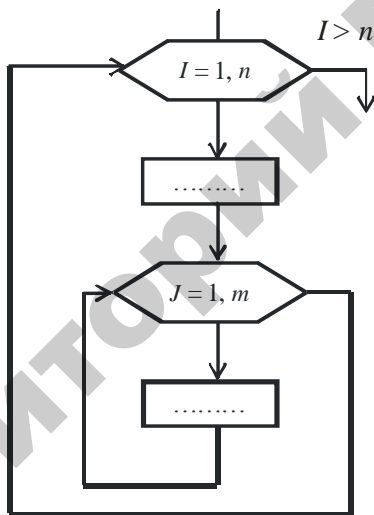


Рисунок 6 – Схема алгоритма вложенного цикла

Цикл по J вложен в цикл по I и по отношению к нему является внутренним, а цикл по I – внешним. Из схемы алгоритма следует, что при $I = 1$ цикл по J повторяется m раз, т. е. на 1 внешний цикл по I приходится m внутренних по J , поэтому общее число циклов в данном алгоритме будет $n \times m$. В алгоритмах может быть любое число вложений в цикл.

Циклы с одним вложением обычно используются при обработке таблиц, состоящих из строк и столбцов, и имеющих вид двумерных

$X_{1,1} X_{1,2} X_{1,3} X_{1,4} X_{1,5}$

$X_{2,1} X_{2,2} X_{2,3} \dots X_{2,5}$

.....

$X_{8,1} X_{8,2} X_{8,3} X_{8,4} X_{8,5}$

Элемент матрицы $X [I, J]$ – оценка за экзамен, где первый индекс I указывает номер строки (номер студента), а второй индекс J – номер столбца (номер экзамена). Схема алгоритма решения задачи приведена на рисунке 9.

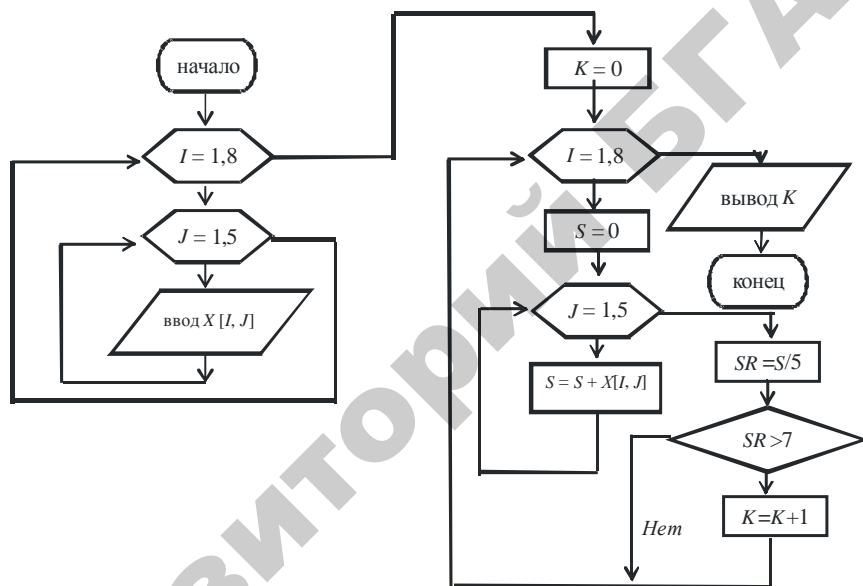


Рисунок 9 – Схема алгоритма решения к примеру 3

```
PROGRAM PRIM3;
VAR
  S,I,J,K:INTEGER;
  SR:REAL;
X:ARRAY [1..8,1..5] OF INTEGER;
BEGIN
  FOR J:= 1 TO 8 DO
  FOR I:= 1 TO 5 DO READLN(X[I,J]);
  K:=0;
```

```

FOR I:= 1 TO 8 DO BEGIN
    S:=0;
    FOR J:= 1 TO 5 DO S:=S+X[I,J];
    SR:= S/5;
    IF SR>7 THEN K:=K+1;
    END;
WRITELN ('количество студентов',K:3);
END.

```

В данном алгоритме ввод элементов матрицы проведен по строкам, а затем обнуляется счетчик количества студентов ($K = 0$). Далее при $I = 1$ внутренний цикл по $J = 1,5$ осуществляет суммирование баллов по каждому из экзаменов ($S = S + X [I, J]$), предварительно принимая сумму баллов для каждого студента, равной нулю ($S = 0$). По выходе из внутреннего цикла вычисляется средний балл каждого студента SR и количество студентов увеличивается на единицу ($K = K + 1$) при $SR > 7$, а при невыполнении этого условия K остается прежним. После этого осуществляется выход во внешний цикл, и процесс повторяется для следующего студента ($I = 2$) и т. д., пока I не превысит 8.

ПРИМЕРЫ РАЗНОУРОВНЕВЫХ ЗАДАНИЙ ДЛЯ КОНТРОЛЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО МОДУЛЮ

I УРОВЕНЬ

Вариант 1

1. Для матрицы **a(3,4)** определить среднее арифметическое каждого столбца матрицы.
2. В матрице **a(4,2)** заменить на 1 все положительные элементы, на 0 все отрицательные элементы и вывести полученную новую матрицу на экран.

Вариант 2

1. Подсчитать количество положительных элементов в каждом столбце матрицы **b (3,5)**.
2. В матрице **c(4,4)** найти среднее арифметическое в каждой строке.

Вариант 3

1. Найти среднее арифметическое значение неотрицательных элементов матрицы $\mathbf{a}(3,4)$.
2. В матрице $\mathbf{b}(5,5)$ найти суммы всех элементов каждой строки, кроме элемента, стоящего на главной диагонали.

Вариант 4

1. В матрице $\mathbf{b}(2,5)$ найти среднее арифметическое значение отрицательных элементов каждого столбца.
2. В матрице $\mathbf{z}(3,4)$ каждого столбца найти суммы элементов каждой строки.

Вариант 5

1. Из матрицы $\mathbf{a}(3,3)$ получить новую $\mathbf{b}(3,3)$, умножив элементы каждого столбца на элемент главной диагонали, находящейся в данном столбце.
2. Дана действительная матрица $\mathbf{x}(5,3)$. Определить количество отрицательных элементов в каждой строке матрицы.

Вариант 6

1. Из матрицы $\mathbf{b}(4,4)$ получить новую матрицу $\mathbf{c}(4,4)$, умножив элементы каждой строки на элемент главной диагонали, находящийся в данной строке.
2. Для матрицы $\mathbf{a}(3,5)$ напечатать номера тех столбцов, сумма элементов которых положительна.

Вариант 7

1. Для матрицы $\mathbf{x}(5,5)$ вывести на печать суммы элементов каждого из ее столбцов, у которых элемент главной диагонали отрицательный.
2. В каждом столбце матрицы $\mathbf{b}(2,4)$ подсчитать количество элементов, равных нулю.

Вариант 8

1. Для матрицы $\mathbf{a}(5,3)$ найти среднее арифметическое значение элементов в каждой строке матрицы.
2. Дана действительная квадратная матрица $\mathbf{z}(5,5)$. Заменить нулями все ее элементы, расположенные на главной диагонали и выше нее. Полученную матрицу вывести на экран.

Вариант 9

1. В матрице $\mathbf{a}(3,5)$ найти и вывести на печать номера тех столбцов, у которых первый элемент больше второго.
2. В матрице $\mathbf{x}(4,4)$ найти среднее арифметическое значение элементов каждой строки и вывести их на экран.

Вариант 10

1. В матрице $\mathbf{x}(3,4)$ найти среднее арифметическое значение элементов в каждом столбце.
2. В матрице $\mathbf{b}(5,4)$ найти и вывести на печать номера строк матрицы, сумма элементов которых больше 20.

Вариант 11

1. Задана матрица $\mathbf{a}(3,6)$. Найти минимальный элемент матрицы и сумму всех элементов столбца, в котором он находится.
2. В матрице $\mathbf{x}(3,4)$ найти произведение элементов каждой строки.

Вариант 12

1. В матрице $\mathbf{z}(3,4)$ найти максимальный и минимальный элементы.
2. Задана матрица $\mathbf{a}(3,6)$. В каждом столбце матрицы найти количество положительных и отрицательных элементов.

Вариант 13

1. Определить, каких элементов в матрице $\mathbf{a}(3,4)$ больше: положительных, отрицательных или равных нулю.
2. Задана матрица $\mathbf{b}(3,6)$. В каждом столбце матрицы найти произведение положительных элементов.

Вариант 14

1. Найти суммы положительных и отрицательных элементов в каждой строке матрицы $\mathbf{a}(6,3)$.
2. Задана матрица $\mathbf{z}(3,5)$. Найти и вывести на печать все элементы матрицы, удовлетворяющих условию $0 < a[i,j] < 15$, считая, что такие элементы есть.

II УРОВЕНЬ

Вариант 15

1. В матрице \mathbf{a} (3,5) подсчитать количество столбцов, сумма элементов которых положительна.

2. В матрице \mathbf{x} (3,4) найти максимальный элемент и вывести на печать сумму элементов строки, в которой он находится.

Вариант 16

1. Для матрицы \mathbf{a} (4,3) напечатать номера тех строк, суммы элементов которых отрицательны.

2. Для каждой строки матрицы \mathbf{b} (5,5) подсчитать количество элементов, совпадающих с элементом, стоящим на главной диагонали соответствующей строки.

Вариант 17

1. Для каждой строки матрицы \mathbf{a} (5,3) определить сумму и произведение элементов и вывести их на экран.

2. В матрице \mathbf{b} (3,4) найти минимальный элемент и вывести на печать элементы столбца, в котором он находится.

Вариант 18

1. В матрице \mathbf{a} (3,4) определить количество положительных и количество отрицательных элементов в каждом столбце.

2. Дана действительная матрица размера \mathbf{x} (3,4), в которой не все элементы равны нулю. Получить новую матрицу путем деления всех элементов данной матрицы на ее наибольший по модулю элемент. Вывести полученную матрицу на экран.

Вариант 19

1. В матрице \mathbf{x} (3,4) найти сумму положительных элементов и количество отрицательных элементов.

2. Задана матрица \mathbf{a} (5,3). В каждом столбце матрицы найти произведение положительных элементов и вывести их на экран.

Вариант 20

1. Задана матрица \mathbf{a} (3,6). Найти максимальный элемент матрицы и вывести на печать все элементы, расположенные в строке, где он находится.

2. В матрице \mathbf{x} (3,4) найти среднее арифметическое значение элементов каждого столбца и вывести их на экран.

Вариант 21

1. Найти среднее арифметическое значение отрицательных элементов в каждом столбце матрицы **b(5,4)**.
2. Задана матрица **a(3,5)**. Найти и вывести на печать все отрицательные элементы матрицы.

Вариант 22

1. Дана действительная матрица **a(5,3)**. Определить числа b_1, \dots, b_5 , равные произведениям элементов строк.
2. Задана матрица **a(3,4)**. В каждом столбце матрицы найти суммы отрицательных элементов.

Вариант 23

1. Задана матрица **a(3,6)**. Найти минимальный элемент матрицы и вывести на печать все элементы столбца, в котором он расположен.
2. Задана матрица **z(3,4)**. Определить, каких элементов в матрице больше: положительных или отрицательных.

Вариант 24

1. Задана матрица **z(3,5)**. Найти и вывести на печать все элементы матрицы, удовлетворяющие условию $1 < a_{[i,j]} < 10$, считая, что такие элементы есть.
2. Найти суммы элементов первой и последней строк матрицы **b(3,4)**.

III УРОВЕНЬ

Вариант 25

1. Дана действительная матрица **z(5,3)**. Найти сумму наибольших значений элементов ее строк.
2. Задана матрица **a(3,6)**. Определить, каких элементов в матрице больше: отрицательных или элементов ≥ 20 ?

Вариант 26

1. Дана действительная матрица размера **n x m**. Определить числа $b_1 \dots b_n$, равные наименьшим значениям элементов строк.
2. В каждом столбце матрицы **z(3,6)** найти суммы положительных и отрицательных элементов.

Вариант 27

1. Дана действительная матрица $\mathbf{b}(5,4)$. Найти среднее арифметическое наибольшего и наименьшего значений ее элементов.
2. Задана матрица $\mathbf{a}(3,4)$. Определить номер строки, сумма элементов которой максимальная.

Вариант 28

1. Найти и вывести на печать количество элементов в матрице $\mathbf{a}(5,4)$: отрицательных, положительных, равных нулю.
2. Задана матрица $\mathbf{z}(3,5)$. Определить номер столбца с наибольшим количеством отрицательных элементов.

Вариант 29

1. Определить общую сумму наибольших элементов каждой из строк матрицы $\mathbf{a}(4,3)$.
2. Для матрицы $\mathbf{x}(3,5)$ определить номер столбца, в котором наибольшее число положительных элементов.

Вариант 30

1. Найти общую сумму элементов тех столбцов матрицы $\mathbf{a}(2,6)$, сумма элементов каждого, из которых положительна.
2. Элементы каждой строки матрицы $\mathbf{x}(5,3)$ умножить на наибольший элемент соответствующей строки. Полученную матрицу вывести на экран.

МОДУЛЬ № 4 «Численные методы»

В результате изучения модуля студент должен:

- знать сущность математической формализации численных методов решения задач;
- уметь решать прикладные задачи с использованием численных методов на алгоритмическом языке Turbo-Pascal;
- приобрести навыки системного подхода к освоению нового учебного материала.

НАУЧНО-ТЕОРЕТИЧЕСКОЕ СОДЕРЖАНИЕ МОДУЛЯ

СЛОВАРЬ ОСНОВНЫХ ПОНЯТИЙ

Новые понятия

Новое понятие	Определение
Итерационные (численные) методы	Методы последовательного приближения к корню уравнения с заданной точностью E
Итерация	Каждое повторное вычисление корня
Нелинейные уравнения	Уравнение вида $f(x) = 0$, левая часть которого представляет собой многочлен от x в степени больше единицы или содержит тригонометрические, логарифмические и другие элементарные функции

Понятия для повторения

Понятие для повторения	Определение
Массив данных	Совокупность однотипных данных, имеющих общее имя и разные порядковые номера
Индекс (порядковый номер)	Указывает место (положение) элемента в массиве
Сложные циклы	Циклы, в теле которых имеются разветвления и другие встроенные в них циклы

ОСНОВНОЙ ТЕКСТ

Лекция 1 Вычисление определенных интегралов численными методами

План лекции:

1. Вычисление определенных интегралов методом левых, правых и средних прямоугольников.
2. Вычисление определенных интегралов методом трапеций.

Определенный интеграл от непрерывной функции $f(x) \geq 0$ в пределах от a до b представляет площадь криволинейной трапеции S , ограниченной кривой $f(x)$, осью абсцисс и прямыми $x = a$, $x = b$ (рисунок 1). Из курса высшей математики известно, что

$$S = \int_a^b f(x) dx = F(x) \Big|_a^b = F(b) - F(a),$$

где $F(x)$ – первообразная для $f(x)$ на отрезке $[a, b]$, т. е. $F'(x) = f(x)$ на отрезке $[a, b]$. Если $f(x) < 0$ на отрезке $[a, b]$, то в формуле $S < 0$, но $|S|$ равно площади криволинейной трапеции, находящейся под осью абсцисс.

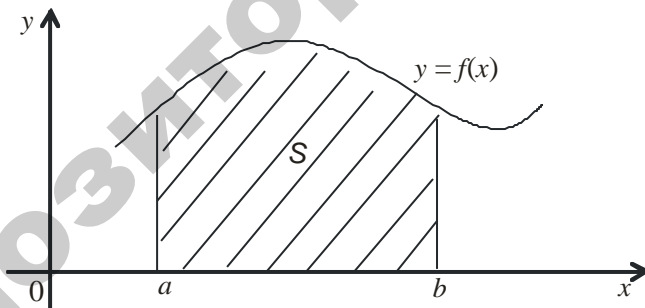


Рисунок 1 – Определенный интеграл – площадь криволинейной трапеции

Однако на практике приведенной формулой часто нельзя воспользоваться по двум основным причинам: 1) вид функции $f(x)$ не допускает непосредственного интегрирования, т. е. первообразную нельзя выразить в элементарных функциях; 2) значения функции $f(x)$ заданы только на фиксированном конечном множестве точек x_i , т. е. функция задана в виде таблицы. В этих случаях используются

методы численного интегрирования. Они основаны на аппроксимации подынтегральной функции некоторыми более простыми выражениями, например многочленами нулевой ($y = c$), первой ($y = cx + d$) или второй ($y = cx^2 + dx + k$) степени, а численные методы вычисления определенного интеграла, основанные на подобной аппроксимации, называются соответственно методами прямоугольников, трапеций и Симпсона (парабол).

Пусть требуется приближенно вычислить значение интеграла

$$S = \int_a^b f(x) dx. \text{ В методе прямоугольников криволинейная трапеция}$$

разбивается на n частей, каждая из которых представляет собой прямоугольник, основание которого равно шагу интегрирования

$$h = \frac{b - a}{n}, \text{ а длины сторон соответственно } Y_0 = f(x_0), Y_1 = f(x_1), Y_n =$$

$f(x_n)$, где $x_0 = a, x_1, \dots, x_{n-1}, x_n = b$ – точки деления отрезка $[a, b]$ на n равных частей.

Различают методы правых, левых и средних прямоугольников, в зависимости от месторасположения начальной точки x_0 при вычислении площади элементарного прямоугольника. Если за высоту каждого прямоугольника принимается левая ордината ($y_0, y_1, y_2 \dots$), то вычисление интеграла будет производиться по методу левых прямоугольников; если правая ордината ($y_1, y_2, y_3 \dots$), то по методу правых прямоугольников; если за высоту принимается середина интервала длиной h , то будет применяться метод средних прямоугольников. Основанием всех прямоугольников будет являться величина шага интегрирования h .

Тогда при методе левых прямоугольников:

$$\int_a^b f(x) dx = S_1 + S_2 + \dots + S_n = y_0 h + y_1 h + \dots + y_{n-1} h,$$

при методе правых:

$$\int_a^b f(x) dx = S_1 + S_2 + \dots + S_n = y_1 h + y_2 h + \dots + y_n h,$$

при методе средних:

$$\int_a^b f(x)dx = S_1 + S_2 + \dots + S_n = y'_1 h + y'_2 h + \dots + y'_n h..$$

Таким образом, первоначальное значение x при методе левых прямоугольников $x = a$, правых – $x = a + h$, средних – $x = a + h/2$. Последующие значения x будут получаться через операцию присваивание $x = x + h$, а элементарные площади $S_1, S_2 \dots S_n$ будут вычисляться по формуле $St = f(x)h$. Сумма этих площадей дает значение интеграла. Изложенное выше реализует алгоритм (рисунок 2), где St – значения элементарных площадей, а их сумма S – значение интеграла.

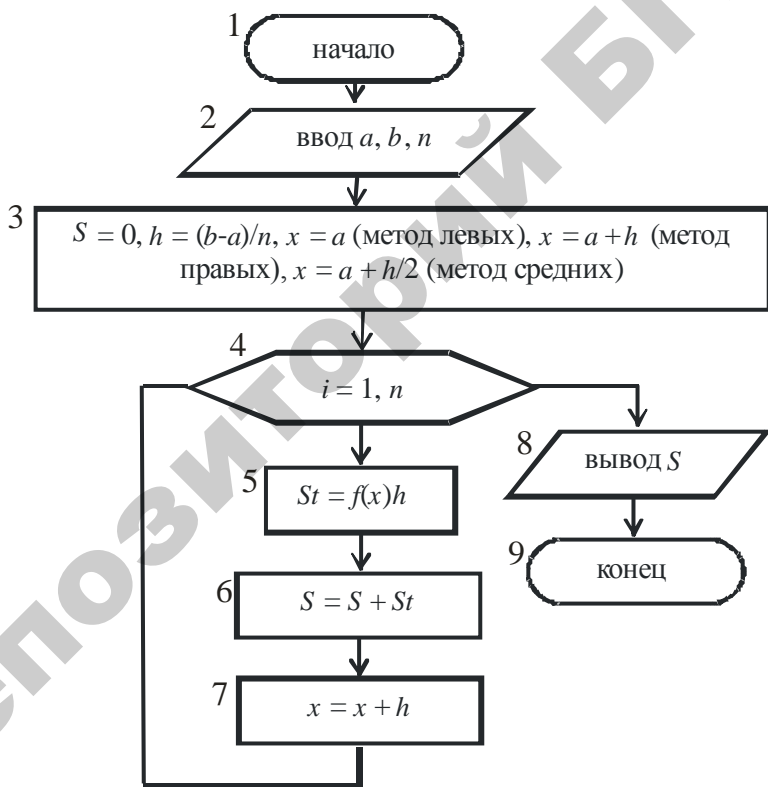


Рисунок 2 – Схема алгоритма вычисления интеграла методом прямоугольников

Более точное значение интеграла получается при вычислении его методом трапеций, когда ординаты ($y_0, y_1, y_2 \dots y_n$) подынтегральной функции соединяют отрезками прямых и искомую площадь заменяют суммой площадей трапеций, высотой которых является шаг h , а основаниями y_0 и y_1 для S_1 , y_1 и y_2 для S_2 (рисунок 3).

Тогда

$$\int_a^b f(x)dx = S_1 + S_2 + \dots + S_n = h \frac{y_0 + y_1}{2} + h \frac{y_1 + y_2}{2} + \dots + h \frac{y_{n-1} + y_n}{2},$$

где $h = (b - a) / n$, а $y_0, y_1, y_2 \dots y_n$ равны значениям функции $f(x)$ при соответствующих значениях аргумента x .

Поскольку $y_0 = f(a)$, $y_1 = f(a + h)$, $y_2 = f(a + 2h)$ и т. д., то схема алгоритма примет вид, приведенный на рисунке 3. В приведенном алгоритме блок 5 вычисляет значение элементарных площадей S_1, S_2, \dots, S_n , в блоке 6 осуществляется их суммирование и блоком 7 изменяется x на величину шага h .

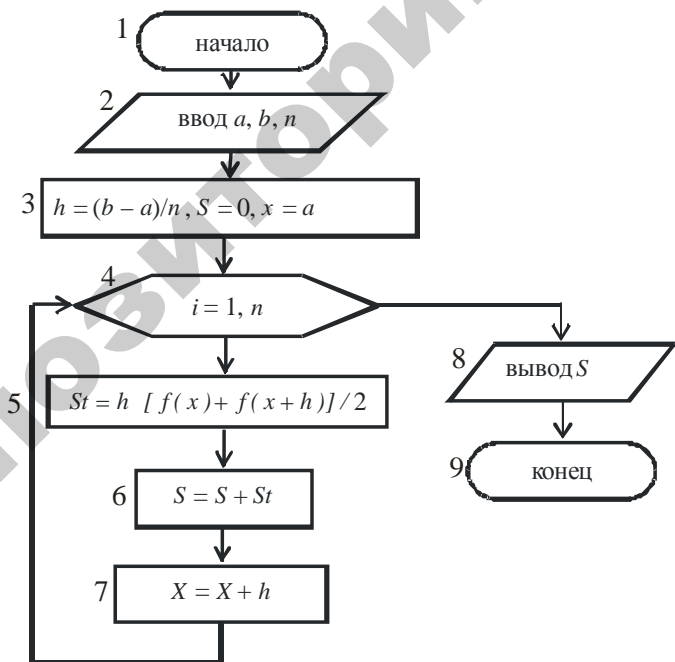


Рисунок 3 – Схема алгоритма вычисления интеграла методом трапеций

Лекция 2 Решение нелинейных уравнений численными методами

При решении любое нелинейное уравнение, например $x^3 - 3x = 1$, предварительно приводится к виду $f(x) = 0$, т. е. $x^3 - 3x - 1 = 0$, при этом функции $f(x)$ могут быть алгебраическими, у которых над аргументом x выполняются лишь арифметические операции, и трансцендентными, которые содержат, кроме того, показательные, логарифмические, тригонометрические функции аргумента x . Такие уравнения далеко не всегда можно решить точно, а для практических расчетов точное решение не всегда является обязательным. Поэтому важное значение с развитием вычислительной техники приобрели способы приближенного вычисления корней и оценки их точности.

Данная процедура состоит из двух этапов: 1) отделение корней, т. е. выделение таких отрезков $[a, b]$, в каждом из которых находится только один корень уравнения или нахождение первоначальных приближений корней; 2) уточнение корней, т. е. нахождение их на найденных отрезках с заданной степенью точности.

Простейшим способом отделения корней является графический. На графике функции $y = f(x)$ определяются отрезки $[a, b]$, в пределах которых лежат точки пересечения функции с осью OX , являющимися приближенными значениями корней. Например, при решении уравнения $x^3 - 3x - 1 = 0$ сначала вычисляем значения функции $y = x^3 - 3x - 1$, по заданным значениям аргумента x (таблица 1) и затем по полученным данным строим график (рисунок 4).

Таблица 1 – Результаты табулирования функции $y = x^3 - 3x - 1$

x	-3	-2	-1	0	1	2	3	4
y	-19	-3	1	-1	-3	1	17	51

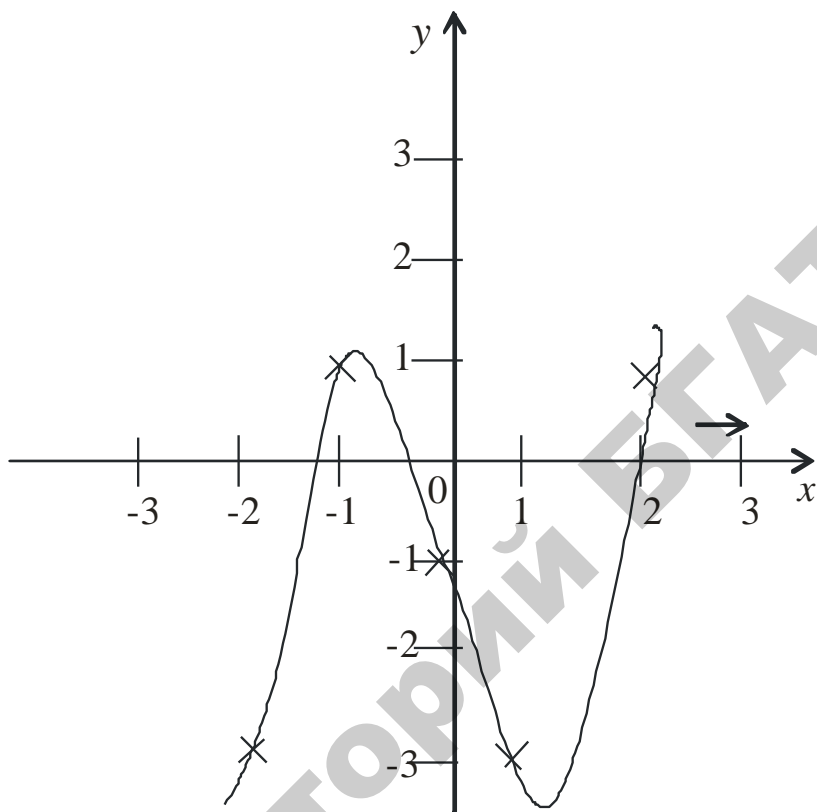


Рисунок 4 – График функции $y = x^3 - 3x - 1$

Как видно из графика, уравнение имеет 3 корня на отрезках $[-2, -1]$, $[-1, 0]$ и $[1, 2]$.

Отделить корни можно также программным способом. Для составления алгоритма решения данной задачи рассмотрим поведение функции $y = f(x)$ на интервале от $x_{\text{нач}}$ до $x_{\text{кон}}$ с шагом h (рисунок 5).

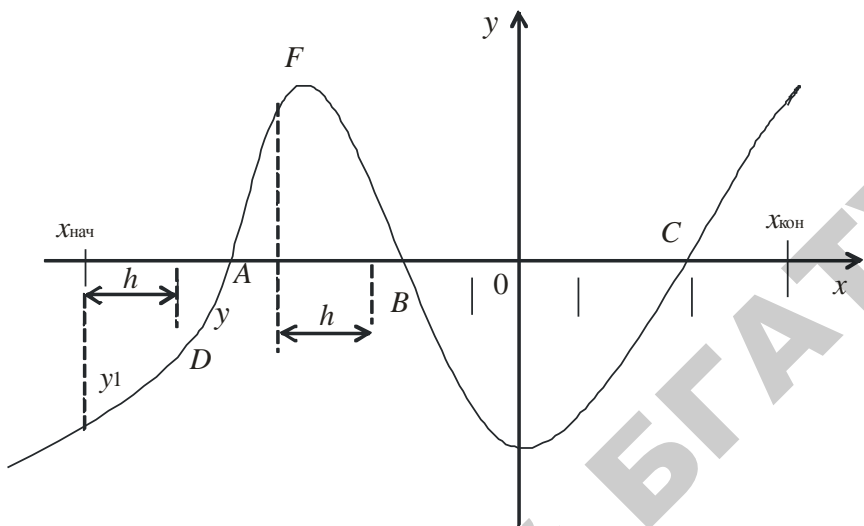


Рисунок 5 – График функции $y = f(x)$ при наличии корней

Как видно из рисунка, корни уравнения находятся в точках A , B и C , где $f(x) = 0$. Данные точки можно обнаружить, рассчитывая значения функции $y = f(x)$ от $x_{\text{нач}}$ до $x_{\text{кон}}$ с шагом h и наблюдая за изменением ее знака. Так, в точке A значение функции изменяется с «-» на «+», а в точке B с «+» на «-» и т. д. Обнаружить изменение знака функции можно путем перемножения двух ее значений, вычисленных от аргументов x , находящихся в соседних точках на расстоянии h одно от другого. Для точки A , например, это будут значения функции D и F .

Отделение корней по программному методу реализует алгоритм, представленный на рисунке 6.

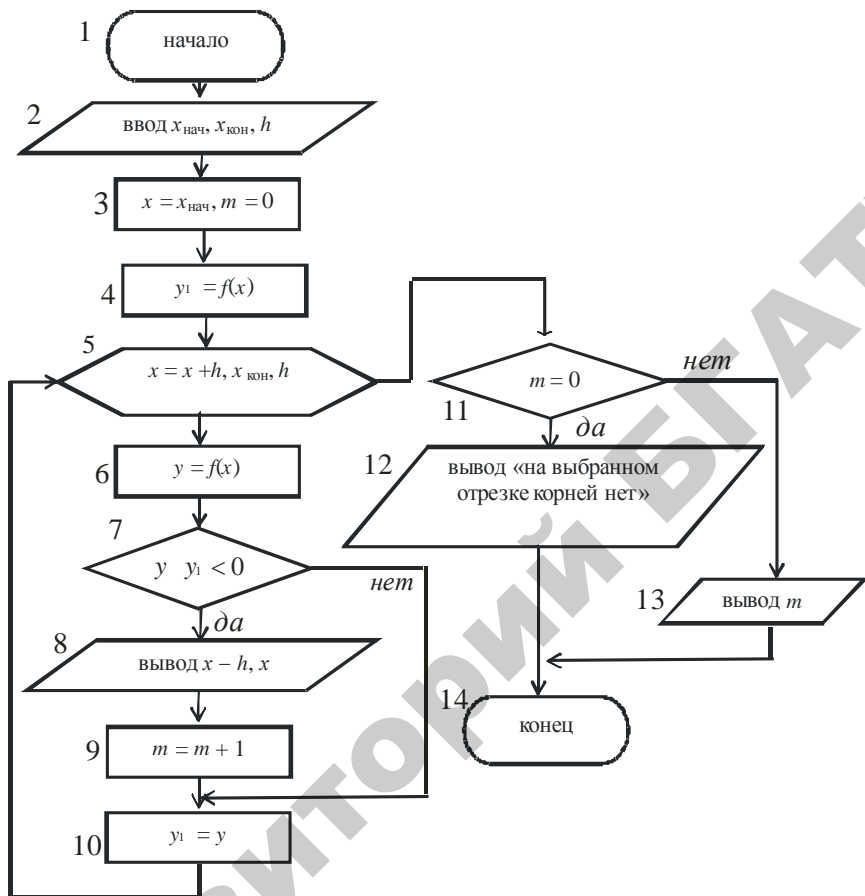


Рисунок 6 – Схема алгоритма отделения корней функции $f(x)$ на отрезке $[x_{нач}, x_{кон}]$

В блоке 2 вводятся начальное значение аргумента функции $x_{нач}$, конечное $x_{кон}$ и шаг его изменения h .

В блоке 3 принимается текущее значение x за $x_{нач}$ и количество корней $m = 0$, а в блоке 4 вычисляется первоначальное значение функции y_1 при $x = x_{нач}$.

Блок 5 организует цикл по изменению x от $x + h$ до $x_{кон}$ с шагом h . В блоке 6 вычисляется очередное значение функции в точке $x = x + h$ ($y = f(x)$). Затем в блоке 7 проверяется, не пересекла ли функция $y = f(x)$ ось OX . В случае отсутствия пересечения $y_1 > 0$ полученное значение y в блоке 10 запоминается как y_1 , а в

блоке 5 значение x опять изменяется на величину шага h . Затем в блоке 6 снова вычисляется y и в блоке 7 проверяется функция на изменение знака. Так как в этом случае $yy_1 < 0$ (точки D и F на рисунке 5), то в блоке 8 выводятся значения аргумента $x - h$ и x , в пределах которых находится корень уравнения, а в блоке 9 фиксируется увеличение количества найденных корней на единицу.

Циклический процесс повторяется до принятого конечного значения аргумента $x_{\text{кон}}$. В результате будут найдены отрезки, на которых находятся корни нелинейного уравнения (около точек A, B, C на рисунке 5.).

Вопросы для самоконтроля

1. В чем сущность численных методов вычисления определенного интеграла?
2. От чего зависит точность вычисления интеграла?
3. В чем отличие метода трапеций от метода прямоугольников?
4. Чем объясняется более высокая точность при вычислении интеграла методом трапеций?
5. Из каких этапов состоит процесс решения нелинейных уравнений?
6. Чем характерна область, где находится корень уравнения?
7. Как в алгоритме и программе определяется область нахождения корня?
8. В чем сущность нахождения корня с необходимой точностью в заданной области?

МАТЕРИАЛЫ К ЛАБОРАТОРНОЙ РАБОТЕ

Тема: нахождение корней нелинейного уравнения.

Цель: уметь отделить корни нелинейного уравнения на определенных отрезках и уточнить их с заданной точностью одним из численных методов.

Уточнение корня, найденного на отрезке $[a, b]$, осуществляется одним из следующих методов: деления отрезка пополам, хорд (секущих), касательных (Ньютона), итераций. Рассмотрим некоторые из них, например метод деления отрезка пополам (рисунок 7). Интервал $[a, b]$ делится пополам и в найденной точке $(c = (a + b) / 2)$ вычисляется значение функции $y = f(c)$. Если $|y| \leq \epsilon$, где ϵ – заданная точность, то c является корнем уравнения, т. к. при полученном c функция $y = f(c)$ пересекает ось OX . В противном случае выбираем один из отрезков или $[a, (a + b) / 2]$ (рисунок 7а) или $[(a + b) / 2, b]$

(рисунок 7б), на концах которого $f(x)$ имеет противоположные знаки. Выбранный интервал снова делим пополам ($c = (a + b) / 2$) и вычисляем значение функции $y = f(c)$. Процесс повторяется до тех пор, пока не будет получено значение $|y| \leq e$.

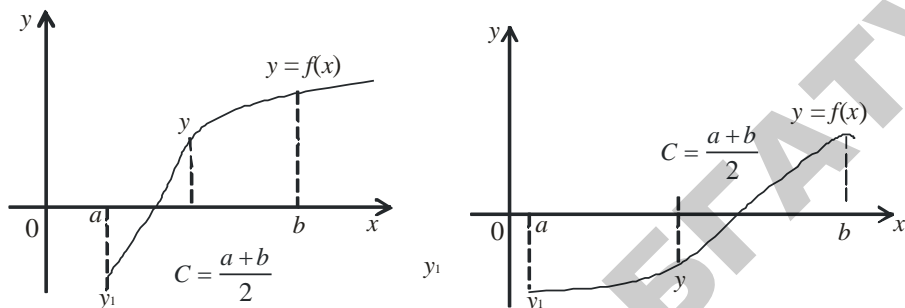


Рисунок 7 – Уточнение корня методом деления отрезка пополам

Сказанное выше реализуется следующим алгоритмом (рисунок 8), где в блоке 2 вводятся полученные при отделении корней границы интервала $[a, b]$ и точность вычисления корня e , а в блоке 3 вычисляется значения функции y_1 при $x = a$. Затем в блоке 4 вычисляется середина интервала $[a, b]$, а в блоке 5 – значение функции в середине данного интервала при $c = (a + b) / 2$. Если при проверке в блоке 6 оказывается $|y| \leq e$, то c – корень уравнения, который выводится в блоке 10. Если же условие $|y| \leq e$ не выполняется, то в блоке 7 определяется: какую половину отрезка $[a, b]$ оставить для дальнейшего нахождения корня. Если $y y_1 < 0$, то левую половину присвоением $b = c$ (блок 9), а если же нет, то правую присвоением $a = c$ (блок 8) и затем в блоке 4 опять определяется середина нового суженного интервала и процесс повторяется до тех пор, пока значение y станет меньше заданной точности e .

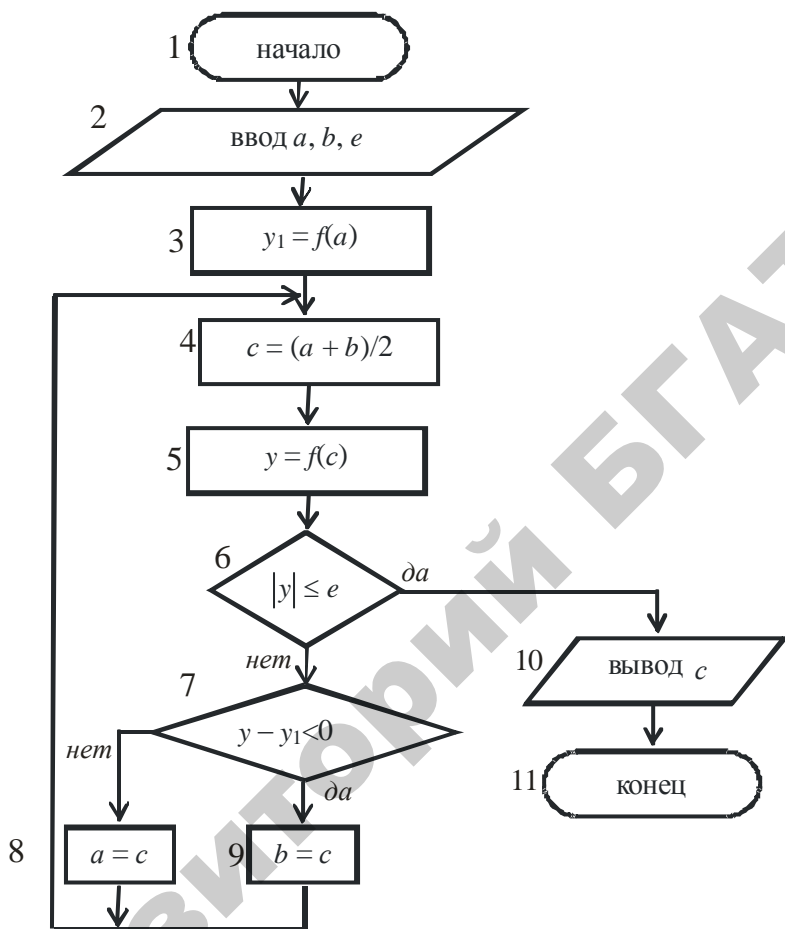


Рисунок 8 – Схема алгоритма уточнения корня методом деления отрезка пополам

При уточнении корня методом итераций в уравнении неизвестное выражают через самого себя, т. е. уравнение приводится к виду $x = f(x)$.

Тогда рассмотренное выше уравнение $x^3 - 3x = 1$ преобразуем к виду $x = (x^3 - 1)/3$.

Выберем произвольную точку x внутри отрезка $[a, b]$, на котором находится корень уравнения, и подставим это значение в правую

часть преобразованного уравнения, получив соответственно $x_n = (x^3 - 1)/3$. Затем, приняв x равным полученному x_n ($x = x_n$), опять проведем вычисления нового x_n .

Этот процесс последовательного вычисления значений x_n по формуле будет продолжаться до тех пор, пока разность между вычисленным x_n и предыдущим x по модулю не станет меньше заданной точности e ($|x_n - x| \leq e$). Рассмотренное выше нахождение корня реализуется следующим алгоритмом (рисунок 9).

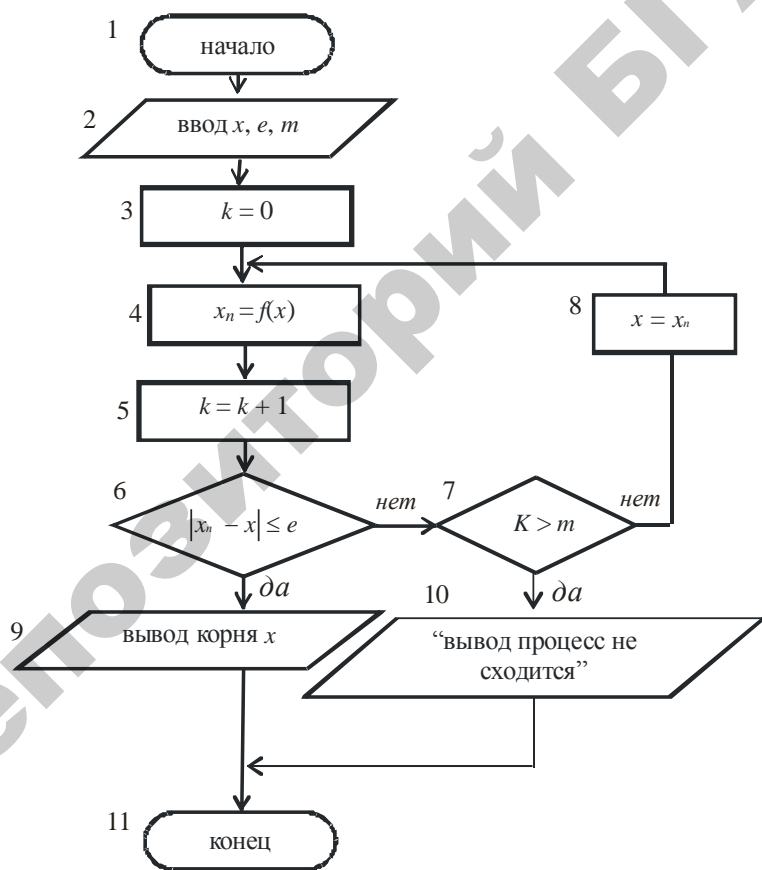


Рисунок 9 – Схема алгоритма уточнения корня методом итераций

Метод итераций применим только в том случае, если вычислительный процесс сходится, т. е. от итерации к итерации абсолютная разность $|x_n - x|$ будет уменьшаться. Для этого необходимо провести преобразования исходного уравнения к виду $x = f(x)$ так, чтобы выполнялось условие $|f'(x)| < 1$ для любого значения x , принадлежащего отрезку $[a, b]$.

Для предотвращения заклинивания в случае расходящегося процесса в схему алгоритма блоком 2 вводится параметр m , обеспечивающий ограничение на максимальное число итераций. Количество итераций подсчитывается в блоке 5 и при превышении заданного числа m блок 7 прерывает процесс поиска корня.

ПРИМЕРЫ РАЗНОУРОВНЕВЫХ ЗАДАНИЙ ДЛЯ КОНТРОЛЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО МОДУЛЮ

Для нелинейного (алгебраического или трансцендентного) уравнения, приведенного в таблице 4, произвести отделение его корней и уточнить их с помощью разработанных алгоритмов и программ.

Аргументы тригонометрических функций принять в радианах, а уравнение выбрать в соответствии с таблицей 4.

Таблица 4 – Варианты нелинейных уравнений

№ п/п	Нелинейное уравнение
1	$x - \cos x = 0$
2	$2 - x = \ln x$
3	$x^3 - 9x + 3 = 0$
4	$x^3 - 8x + 2 = 0$
5	$x^3 - 15x + 10 = 0$
6	$e^x + x = 0$
7	$x - \sin x = 0,25$
8	$x^3 + 10x + 3 = 0$
9	$x^3 + 3x - 1 = 0$
10	$x - 2 + e^x = 0$
11	$2x - \cos x = 0$

Окончание таблицы 4

№ п/п	Нелинейное уравнение
12	$x^3 - 10x + 5 = 0$
13	$x^3 - 20x - 10 = 0$
14	$x - 3 + e^x = 0$
15	$2x - \cos x = 0$
16	$x^3 - 2x - 1 = 0$
17	$x - \sin x = 1$
18	$\sin x - 5x + 1,5 = 0$
19	$5x^3 = 2 - x$
20	$2^x = 4x + 3$
21	$3x - \cos x = 1$
22	$4x - 6\sin x = 0$
23	$x^3 - 15x + 8 = 0$
24	$x + \ln x = 0,5$
25	$2x - \sin x - 0,9 = 0$
26	$x^3 + 4x - 3 = 0$
27	$e^x + e^{-3x} = 4$
28	$3x - \cos x - 1 = 0$
29	$2x + x^3 - 1,2 = 0$
30	$x^3 + 3x = 2$

ЗАДАНИЯ ДЛЯ УПРАВЛЯЕМОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Тема: вычисление определенного интеграла.

Цель работы: уяснить сущность метода численного решения задачи и овладеть первичными навыками составления, ввода, трансляции, отладки, исполнения и оформления программного модуля.

Для заданного варианта интегрируемой функции (номер варианта соответствует порядковому номеру в списке группы) составить схемы алгоритмов и программы на алгоритмическом языке Turbo-Pascal.

Методы вычисления определенного интеграла

Численное решение рассматриваемой задачи в режиме программирования может быть выполнено следующими методами: метод прямоугольников, метод трапеций и метод Симпсона (метод парабол) и т. д. Эти методы с точки зрения программной реализации отличаются только способом вычисления площади фигуры, ограниченной отрезками функции в точках нижнего и верхнего пределов интегрирования $[a, b]$, линией графика функции и отрезком оси X в границах $[a, b]$. Методы могут отличаться точностью результатов вычислений, если задачу решать без достижения требуемой точности, а также скоростью решения. Однако, если задачу решать, добиваясь требуемой точности результата, то указанные различия могут быть только в скорости решения. Различия в скорости решения можно определить по величине числа N , которое зависит от количества повторений (итераций) внешнего цикла в алгоритме задачи. Поэтому остановимся только на методе прямоугольников как наиболее простом.

Пример вычисления определенного интеграла $\int_a^b \sin x dx$:

методом левых прямоугольников:

```
PROGRAM PRIMLP;  
VAR A,B,N: INTEGER;  
      H, S, St : REL;  
BEGIN  
  WRITELN ('Введите A, B, N');  
  READLN (A, B, N);  
  H: = (B-A)/N;  
  S: = 0;  
  X: = A;  
  FOR I: = 1 TO N DO BEGIN  
      St: =SIN(X)*H  
      S: = S + St;  
      X: = X + H;  
      END;  
  WRITELN ('Значение интеграла =', S:10:5);  
  END.
```

Просуммировав все найденные площади St , мы получим значение искомого интервала с некоторой погрешностью, которую можно уменьшить, увеличив число разбиений n .

Таблица 2 – Варианты заданий к УСРС

№ п/п	Функция $f(x)$	a	b		№ п/п	Функция $f(x)$	a	b
1	$7x^2$	0	2		16	$5\cos(x)$	0,5	2
2	$2\cos(x)$	0	$3\pi/2$		17	$2\sin(x)$	$-\pi/2$	$\pi/2$
3	$3\sin(x)$	0	$\pi/2$		18	$3x^3$	0	2
4	$\frac{1}{x^2+1}$	0,5	1,5		19	$\frac{1}{\cos(x)}$	0	$\pi/3$
5	$2\sin(x)+1$	0	3		20	$3x^2$	0	2
6	$6x^2$	0	2		21	$3\sin(x)$	0,5	1,5
7	$(1+x)\sin(x)$	0	1		22	$6x$	0,5	1,5
8	$6\cos(x)$	1	2		23	$5\cos(x)$	0	2
9	$x^2 + \frac{1}{x}$	0,5	2		24	$2\sin(x)$	1	3
10	$\cos(1+x)$	0	1,5		25	$x^2+\cos(x)$	0	1
11	$\sin^2(x)$	0	1		26	$8x^2$	0	1
12	$9x^2$	0	2		27	$6\cos(x)$	1	3
13	$4\cos(x)$	0	2		28	$\frac{\sin(x)}{x}$	1	2
14	$\frac{1}{\sin(x)}$	1	2		29	$4x^3$	0	1
15	$8x^2$	1	3		30	$\sin(x)+x$	1	2

Форма контроля: представить преподавателю отчет в письменном виде по изученному разделу.

В отчет по работе следует включить цель работы, алгоритмы и программы вычисления определенного интеграла заданными чис-

ленными методами. Все результаты вычисления интеграла свести в таблицу 3.

Таблица 3 – Результаты вычисления определенного интеграла

Число итераций n	Значение определенного интеграла вычисленного методами:			
	левых прямоугольников	правых прямоугольников	средних прямоугольников	трапеций
10				
50				
100				

ЛИТЕРАТУРА

1. Аляев, Ю.А. Практикум по алгоритмизации и программированию на языке Паскаль: учеб. пособие / Ю.А. Аляев, В.П. Гладков, О.А. Козлов. – Москва : Финансы и статистика, 2004.
2. Вычислительная техника и информатика : метод. указания / сост. М.А. Прищепов, Н.В. Исаеня, Е.В. Севернева, Н.М. Жалобкевич. – Минск : БГАТУ, 2007.
3. Интегрированная среда программирования Turbo-Pascal : метод. указания / сост. М.А. Прищепов, Е.В. Севернева, А.И. Шакирин, Н.М.Жалобкевич. – Минск : БГАТУ, 2002.
4. Основы алгоритмизации и программирования на языке Turbo-Pascal : метод. указания / сост. М.А. Прищепов, Е.В. Севернева, А.И. Шакирин, Н.М.Жалобкевич. – Минск : БГАТУ, 2004.
5. Прищепов, М.А. Программирование на языках Basic, Pascal и Object Pascal в среде Delphi : учеб. пособие / М.А. Прищепов, Е.В. Севернева, А.И. Шакирин. – Минск : Тетра Системс, 2006. – 318 с.
6. Прищепов, М.А. Экзамен по информатике. Основы алгоритмизации и программирования : учеб. пособие / М.А. Прищепов, Е.В. Севернева, В.П. Степанцов. – Минск : Тетра Системс, 2001. – 312с.
7. Турчак, Л.И. Основы численных методов : учеб. пособие / Л.И. Турчак, П.В. Плотников. – Москва : ФизМат-лит.

СОДЕРЖАНИЕ

Введение	3
МОДУЛЬ № 1 «Основы программирования. Математическая формализация текстовых задач. Составление алгоритмов их решения. Алго- ритмический язык Turbo-Pascal. Алгоритмы линейной и раз- ветвляющейся структуры и их реализация на языке Turbo- Pascal»	4
МОДУЛЬ № 2 «Алгоритмы циклической структуры и их реализация на языке Turbo-Pascal. Операторы FOR, WHILE, REPEAT»	53
МОДУЛЬ № 3 «Работа с двумерными массивами (матрицами)»	78
МОДУЛЬ № 4 «Численные методы»	93
Литература	111

Учебное издание

**Севернева Елена Васильевна,
Жалобкевич Нина Михайловна**

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Учебно-методическое пособие

Ответственный за выпуск ?

Редактор, корректор *М.А. Макрецкая*
Технический редактор *М.А.Макрецкая*
Компьютерная верстка *М.А. Макрецкая*

Подписано в печать ... г. Формат 60×84¹/₁₆.
Бумага офсетная. Гарнитура Times New Roman. Ризография. Усл. печ. л. 5,3.
Уч.-изд. л. 4,2. Тираж экз. Заказ

Издатель и полиграфическое исполнение
Белорусский государственный аграрный технический университет
ЛИ № 02330/0131734 от 10.02.2006. ЛП № 02330/0131656 от 02.02.2006.
220023, г. Минск, пр. Независимости, 99, к. 2

ДЛЯ ЗАМЕТОК

РЕПОЗИТОРИЙ БГАТУ