**Министерство образования Республики Башкортостан**

УФИМСКИЙ КОЛЛЕДЖ РАДИОЭЛЕКТРОНИКИ, ТЕЛЕКОММУНИКАЦИЙ И БЕЗОПАСНОСТИ

|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ  Зам. директора  \_\_\_\_\_\_\_\_\_\_\_\_\_ Л.Р. Туктарова  «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2015 г. |

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛЕНИЮ ПРАКТИЧЕСКИХ (ЛАБОРАТОРНЫХ) РАБОТ И контрольнЫХ ЗАДАНИЙ**

**ДЛЯ СТУДЕНТОВ ЗАОЧНОЙ ФОРМЫ ОБУЧЕНИЯ**

**ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»**

***специальность 09.02.01 «Компьютерные системы и комплексы»***

|  |  |
| --- | --- |
|  | РАЗРАБОТЧИК  \_\_\_\_\_\_\_\_\_\_\_ М.Е.Бронштейн  РАССМОТРЕНО  на заседании кафедры «Программирование и информационные технологии»  \_\_\_\_\_\_\_\_\_\_\_\_\_ М.Е. Бронштейн  «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2015 г. |

Уфа 2015 г.

**Порядок выполнения отчета по лабораторной (практической) работе**

1. Ознакомиться с теоретическим материалом по практической работе.
2. Записать краткий конспект теоретической части.
3. Выполнить предложенное задание согласно варианту по списку группы.
4. Продемонстрировать результаты выполнения предложенных заданий преподавателю.
5. Записать код программы в отчет.
6. Ответить на контрольные вопросы.
7. Записать выводы о проделанной работе.

**Практическая работа № 1**

**«Составление блок-схем алгоритмов»**

**Цель работы:**

**-** изучать свойства алгоритмов и применять и при решении задач;

- составлять блок-схемы алгоритмов.

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям.

знать:

**-** процесс создания программ.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

***Алгоритм***рассматривается как последовательность команд управления работой исполнителя (предписание исполнителю на выполнение последовательности действий).

### Свойства алгоритма

Значение слова *алгоритм* очень схоже со значением слов *рецепт*, *инструкция*. Однако любой алгоритм в отличие от рецепта или способа обязательно обладает следующими свойствами.

1. Выполнение алгоритма разбивается на последовательность законченных действий-шагов. Только выполнив одно действие (команду), можно приступать к исполнению следующего. Это свойство алгоритма называется **дискретностью**. Произвести каждое отдельное действие исполнителю предписывает специальное указание в записи алгоритма (команда).

2. **Понятность** — алгоритм не должен содержать предписаний, смысл которых может восприниматься исполнителем неоднозначно, т.е. запись алгоритма должна быть настолько четкой и полной, чтобы у исполнителя не возникало потребности в принятии каких-либо самостоятельных решений. *Алгоритм всегда рассчитан на выполнение “не размышляющего” исполнителя*. Алгоритм составляется из команд, входящих в СКИ.

3. **Детерминированность**(*определенность и однозначность*). Каждая команда алгоритма определяет однозначное действие исполнителя, и должно быть однозначно определено, какая команда выполняется следующей. То есть если алгоритм многократно применяется к одному и тому же набору исходных данных, то на выходе он получает каждый раз один и тот же результат.

4. **Результативность** — исполнение алгоритма должно закончиться за конечное число шагов, и при этом должен быть получен результат решения задачи. В качестве одного из возможных результатов может быть и установление того факта, что задача решений не имеет.

Свойство результативности содержит в себе свойство **конечности** — завершение работы алгоритма за конечное число шагов.

5. **Массовость** — алгоритм пригоден для решения любой задачи из некоторого класса задач, т.е. алгоритм правильно работает на некотором множестве исходных данных, которое называется областью применимости алгоритма.

Свойство массовости определяет скорее качество алгоритма, а не относится к обязательным свойствам (как дискретность, понятность и пр.). Существуют алгоритмы, область применимости которых ограничивается единственным набором входных данных или даже отсутствием таковых (например, получение фиксированного числа верных цифр числа p). Правильнее говорить о том, что алгоритм должен быть применим к любым данным из своей области определения, и слово *массовость* не всегда подходит для описания такого свойства.

6. **Эффективность**. Алгоритм может быть выполнен не просто за конечное, а за разумно конечное время

7. **Компактность**. Это свойство предполагает лаконичность изложения алгоритма. Как

только компактность потеряна, алгоритм в значительной мере теряет право на существование.

**СПОСОБЫ ОПИСАНИЯ АЛГОРИТМОВ**

Можно назвать три способа написания алгоритмов:

1. на естественном языке (словесно-пошаговый);
2. на языке схем;
3. на алгоритмическом языке.

Все алгоритмы к задачам и примерам были составлены как раз на естественном язы­ке. Алгоритмическим языком, т.е. языком, ориентирован­ным на ЭВМ, мы займемся позже.

Пример. Составить алгоритм  вычисления площади круга

***а) Словесно-пошаговый.***

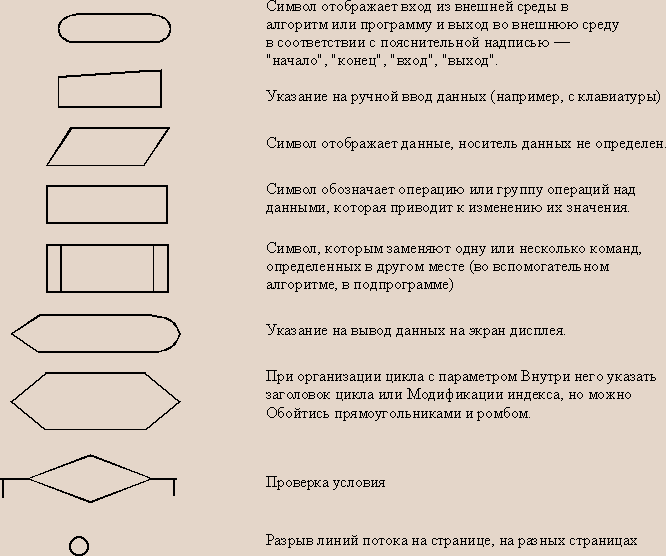
1. Ввести радиус круга R
2. Вычислить площадь круга по формуле S=PI\*R\*R
3. Вывести результат S

***б) Блок-схема***

**СХЕМОЙ** называют наглядное графическое изображение алгоритма, когда отдельные его действия (этапы) изображаются при помощи различных геометрических фигур (блоков), а связи между этапами указываются при помощи стрелок, соединяющих эти фигуры.

Рассмотрим перечень условных обозначений, наиболее часто используемые для представления алгоритмов в графической форме. Для решения задач школьникам их вполне достаточно.

Необходимо также отметить, что представленные в перечне геометрические фигуры, в соответствии с логикой решения задачи, в блок-схеме алгоритма соединяются линиями связи. На концах последних должны обязательно присутствовать стрелки, если их направление снизу вверх или справа налево, то есть отличное от стандартного направления (сверху вниз и слева направо), когда стрелки могут и отсутствовать.



**Пример:**



**Задания для практического занятия:**

1. Составить алгоритм для вычисления длины окружности и площади круга одного и того же заданного радиуса. Представить алгоритм двумя способами: блок-схемой и псевдокоде.

2. Составить алгоритм вычисления значения функции S по заданным значениям x,y. Представить алгоритм двумя способами: блок-схемой и псевдокоде.



3. Составить алгоритм, который позволяет по номеру месяца определить школьную четверть. Представить алгоритм двумя способами: блок-схемой и псевдокоде.

**Контрольные вопросы**

1.Что такое алгоритм?

2. Способы представления алгоритмов.

3. Перечислить свойства алгоритма.

**Практическая работа № 2**

**«Применение систем счисления»**

**Цель работы:** научиться работать в различных позиционных системах счисления.

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям.

знать:

**-** процесс создания программ.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

Под *системой счисления* понимается способ представления любого числа с помощью некоторого алфавита символов, называемых цифрами.

Все системы счисления делятся на *позиционные* и *непозиционные*.

*Непозиционными* системами являются такие системы счисления, в которых каждый символ сохраняет свое значение независимо от места его положения в числе.

Примером непозиционной системы счисления является римская система. К недостаткам таких систем относятся наличие большого количества знаков и сложность выполнения арифметических операций.

Система счисления называется *позиционной*, если одна и та же цифра имеет различное значение, определяющееся позицией цифры в последовательности цифр, изображающей число. Это значение меняется в однозначной зависимости от позиции, занимаемой цифрой, по некоторому закону.

Примером позиционной системы счисления является десятичная система, используемая в повседневной жизни.

Количество *p* различных цифр, употребляемых в позиционной системе определяет название системы счисления и называется *основанием* системы счисления - *"p"*.

В десятичной системе используются десять цифр: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9; эта система имеет основанием число десять.

Любое число *N* в позиционной системе счисления с основанием *p* может быть представлено в виде полинома от основания *p*:

*N = anpn+an-1pn-1+ ... +a1p+a0+a-1p-1+a-2p-2+ ...*

здесь *N* - число, *aj* - коэффициенты (цифры числа), *p* - основание системы счисления ( *p>1*).

Принято представлять числа в виде последовательности цифр:

*N = anan-1 ... a1a0 . a-1a-2 ...*

В этой последовательности точка отделяет целую часть числа от дробной (коэффициенты при положительных степенях, включая нуль, от коэффициентов при отрицательных степенях). Точка опускается, если нет отрицательных степеней (число целое).

В ЭВМ применяют позиционные системы счисления с недесятичным основанием: двоичную, восьмеричную, шестнадцатеричную.

В аппаратной основе ЭВМ лежат двухпозиционные элементы, которые могут находиться только в двух состояниях; одно из них обозначается 0, а другое - 1. Поэтому основной системой счисления применяемой в ЭВМ является двоичная система.

*Двоичная система счисления***.** Используется две цифры: 0 и 1. В двоичной системе любое число может быть представлено в виде:

*N = bnbn-1 ... b1b0 . b-1b-2 ...*

5

где *bj* либо 0, либо 1.

*Восьмеричная система счисления***.** Используется восемь цифр: 0, 1, 2, 3, 4, 5, 6, 7. Употребляется в ЭВМ как вспомогательная для записи информации в сокращенном виде. Для представления одной цифры восьмеричной системы используется три двоичных разряда (триада) (Таблица 1).

*Шестнадцатеричная система счисления***.** Для изображения чисел употребляются 16 цифр. Первые десять цифр этой системы обозначаются цифрами от 0 до 9, а старшие шесть цифр - латинскими буквами: 10-A, 11-B, 12-C, 13-D, 14-E, 15-F. Шестнадцатеричная система используется для записи информации в сокращенном виде. Для представления одной цифры шестнадцатеричной системы счисления используется четыре двоичных разряда (тетрада) (Таблица 1).

**Таблица 1. Наиболее важные системы счисления.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Двоичная (Основание 2) | Восьмеричная (Основание 8) | | Десятичная  (Основание 10) | Шестнадцатиричная (Основание 16) | |
|  | триады |  | тетрады |
| 0 1 | 0 1 2 3 4 5 6 7 | 000 001 010 011 100 101 110 111 | 0 1 2 3 4 5 6 7 8 9 | 0 1 2 3 4 5 6 7 8 9 A B C D E F | 0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 1100 1101 1110 1111 |

*Перевод чисел из одной системы счисления в другую*

*Перевод чисел в десятичную систему* осуществляется путем составления степенного ряда с основанием той системы, из которой число переводится. Затем подсчитывается значение суммы.

**Пример:**

а) Перевести 10101101.1012arrow"10" с.с.

*Здесь и в дальнейшем при одновременном использовании нескольких различных систем счисления основание системы к которой относится число будем указывать в виде нижнего индекса.*

10101101.1012 = 1point27+ 0point26+ 1point25+ 0point24+ 1point23+ 1point22+ 0point21+ 1point20+ 1point2-1+ 0point2-2+ 1point2-3 =  173.62510

б) Перевести 703.048arrow"10" с.с.

703.048 = 7point82+ 0point81+ 3point80+ 0point8-1+ 4point8-2 = 451.062510

6

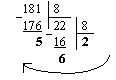
в) Перевести B2E.416arrow"10" с.с.

B2E.416 =  11point162+ 2point161+ 14point160+ 4point16-1 = 2862.2510

*Перевод целых десятичных чисел в недесятичную систему* счисления осуществляется последовательным делением десятичного числа на основание той системы, в которую оно переводится, до тех пор, пока не получится частное меньшее этого основания. Число в новой системе записывается в виде остатков деления, начиная с последнего.

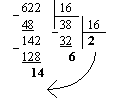
Пример.

а) Перевести 18110arrow"8" с.с.



Результат: 18110 = 2658

б) Перевести 62210arrow"16" с.с.



Результат: 62210 = 26E16

*Перевод правильных дробей из десятичной системы счисления в недесятичную.*Для перевода правильной десятичной дроби в другую систему эту дробь надо последовательно умножать на основание той системы, в которую она переводится. При этом умножаются только дробные части. Дробь в новой системе записывается в виде целых частей произведений, начиная с первого.

Пример.

Перевести 0.312510arrow"8" с.с.

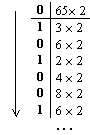
ris3

Результат: 0.312510 = 0.248

*Замечание****.*** Конечной десятичной дроби в другой системе счисления может соответствовать бесконечная (иногда периодическая) дробь. В этом случае количество знаков в представлении дроби в новой системе берется в зависимости от требуемой точности.

**Пример:**

Перевести 0.6510arrow"2" с.с. Точность 6 знаков.



Результат: 0.6510 eql1 0.10(1001)2

*Для перевода неправильной десятичной дроби в систему счисления с недесятичным*

7

*основанием* необходимо отдельно перевести целую часть и отдельно дробную.

Пример.

Перевести 23.12510arrow"2" с.с.

|  |  |
| --- | --- |
| 1) Переведем целую часть: | 2) Переведем дробную часть: |
| ris5 | ris6 |

Таким образом:  2310 = 101112; 0.12510 = 0.0012.   
Результат:  23.12510 = 10111.0012.

Необходимо отметить, что целые числа остаются целыми, а правильные дроби - дробями в любой системе счисления.

*Для перевода восьмеричного или шестнадцатеричного числа в двоичную форму*достаточно заменить каждую цифру этого числа соответствующим трехразрядным двоичным числом (триадой) (Таб. 1) или четырехразрядным двоичным числом (тетрадой) (Таб. 1), при этом отбрасывают ненужные нули в старших и младших разрядах.

Пример.

а) Перевести 305.48arrow"2" с.с.

ris7

б) Перевести 7B2.E16arrow"2" с.с.

ris8

*Для перехода от двоичной к восьмеричной (шестнадцатеричной) системе* поступают следующим образом: двигаясь от точки влево и вправо, разбивают двоичное число на группы по три (четыре) разряда, дополняя при необходимости нулями крайние левую и правую группы. Затем триаду (тетраду) заменяют соответствующей восьмеричной (шестнадцатеричной) цифрой.

**Пример:**

а) Перевести 1101111001.11012arrow"8" с.с.

ris9

б) Перевести 11111111011.1001112arrow"16" с.с.

ris10

*Перевод из восьмеричной в шестнадцатеричную систему и обратно* осуществляется через двоичную систему с помощью триад и тетрад.

Пример. Перевести 175.248arrow"16" с.с.

ris11

Результат: 175.248 = 7D.516.

**Задания для практического занятия:**

1 Перевести следующие числа в десятичную систему счисления:

а) 1101112; б) 10110111.10112; в) 563.448; г) 721.358; д) 1C4.A16; е) 9A2F.B52.

2 Перевести следующие числа из "10" с.с в "2", "8", "16" с.с.:

а) 463; б) 1209; в) 362; г) 3925; д) 11355.

3 Перевести следующие числа из "10" с.с в "2", "8", "16" с.с. (точность вычислений - 5 знаков после точки):

а) 0.0625; б) 0.345; в) 0.225; г) 0.725; д) 217.375; е) 31.2375; ж) 725.03125; з) 8846.04.

4 Перевести следующие числа в двоичную систему счисления:

а) 1725.3268; б) 341.348; в) 7BF.52A16; г) 3D2.C16.

5 Перевести следующие числа из одной системы счисления в другую:

а) 11011001.010112 arrow "8" с.с.;   
б) 1011110.11012 arrow "8" с.с.;   
в) 1101111101.01011012 arrow "16" с.с.;   
г) 110101000.1001012 arrow "16" с.с.

6 Перевести следующие числа из одной системы счисления в другую:

а) 312.78 arrow "16" с.с.;      б) 51.438 arrow "16" с.с.;  
в) 5B.F16 arrow "8" с.с.;        г) D4.1916 arrow "8" с.с.

**Контрольные вопросы**

1 Дать определение системы счисления. Назвать и охарактеризовать виды системы счисления.

2 Какие символы используются для записи чисел в двоичной системе счисления, восьмеричной, шестнадцатеричной?

3 Какое количество цифр используется для записи чисел в восьмеричной СС?

4 Назовите наибольшее однозначное число в десятичной СС? В шестнадцатеричной СС?

**Практическая работа № 3**

**«Применение алгебры логики»**

**Цель работы:** научиться применять формулы и законы алгебры логики

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям.

знать:

**-** процесс создания программ.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

Алгебра логики — раздел [математической логики](http://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BB%D0%BE%D0%B3%D0%B8%D0%BA%D0%B0), в котором изучаются логические операции над [высказываниями](http://ru.wikipedia.org/wiki/%D0%9B%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%B2%D1%8B%D1%81%D0%BA%D0%B0%D0%B7%D1%8B%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5). Высказывания могут быть истинными и ложными.

*Операции алгебры логики:*

1. основные операции

логическое умножение (конъюнкция), логическое сложение (дизъюнкция), отрицание (инверсия), следование(импликация)



1. Исключающее или

AB, A либо В, A xor B (A либо В; A либо В, но не оба)

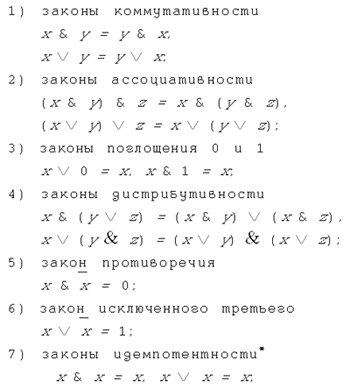


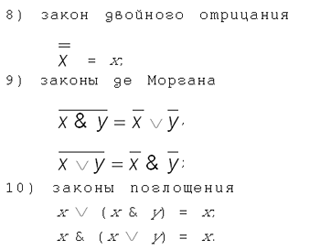
1. Эквиваленция

,A EQV B (А эквивалентно В, А необходимо и достаточно для В, А тогда и только тогда, когда В, А если и только если В)



###### Законы алгебры логики:



**

Любой из этих законов может быть легко доказан с помощью таблиц истинности, или путем логических рассуждений, или с помощью тождественных преобразований, использующих доказанные ранее законы.

Формула называется тавтологией, если она принимает только

истинные значения при любых значениях букв

### *Приоритет выполнения логических операций*

Для логических операций в одном логическом выражении установлен следующий порядок вычислений:

· отрицание — первый, наивысший приоритет;

· конъюнкция — второй приоритет;

· дизъюнкция, разделительная дизъюнкция — третий приоритет;

· импликация, эквивалентность — низший приоритет.

Изменить порядок выполнения операций можно с помощью расстановки скобок.

В алгебре логики дизъюнкция (логическое сложение) играет роль, аналогичную сложению в алгебре действительных чисел, конъюнкция (логическое умножение) — умножению, а

отрицание (инверсия значения логической формулы) — унарному минусу (инверсия знака обычной формулы). Операция эквивалентность аналогична операции отношения “=”, а операция импликация — операции отношения “LesQ”.

**Таблицы истинности для основных логических функций**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Конъюнкция](http://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%8A%D1%8E%D0%BD%D0%BA%D1%86%D0%B8%D1%8F)   |  |  |  | | --- | --- | --- | | ~a | ~b | ~a \land b | | ~0 | ~0 | ~0 | | ~0 | ~1 | ~0 | | ~1 | ~0 | ~0 | | ~1 | ~1 | ~1 | | [Дизъюнкция](http://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B7%D1%8A%D1%8E%D0%BD%D0%BA%D1%86%D0%B8%D1%8F)   |  |  |  | | --- | --- | --- | | ~a | ~b | ~a \lor b | | ~0 | ~0 | ~0 | | ~0 | ~1 | ~1 | | ~1 | ~0 | ~1 | | ~1 | ~1 | ~1 | | [Исключающее](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5_%D0%BF%D0%BE_%D0%BC%D0%BE%D0%B4%D1%83%D0%BB%D1%8E_2) или   |  |  |  | | --- | --- | --- | | ~a | ~b | ~a \oplus b | | ~0 | ~0 | ~0 | | ~0 | ~1 | ~1 | | ~1 | ~0 | ~1 | | ~1 | ~1 | ~0 | |
| [Импликация](http://ru.wikipedia.org/wiki/%D0%98%D0%BC%D0%BF%D0%BB%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F)   |  |  |  | | --- | --- | --- | | ~a | ~b | ~a \rightarrow b | | ~0 | ~0 | ~1 | | ~0 | ~1 | ~1 | | ~1 | ~0 | ~0 | | ~1 | ~1 | ~1 | | [Отрицание](http://ru.wikipedia.org/wiki/%D0%9E%D1%82%D1%80%D0%B8%D1%86%D0%B0%D0%BD%D0%B8%D0%B5)   |  |  | | --- | --- | | ~a | \neg a | | ~0 | ~1 | | ~1 | ~0 | | [Эквиваленция](http://ru.wikipedia.org/wiki/%D0%AD%D0%BA%D0%B2%D0%B8%D0%B2%D0%B0%D0%BB%D0%B5%D0%BD%D1%86%D0%B8%D1%8F)   |  |  |  | | --- | --- | --- | | ~a | ~b | ~a \leftrightarrow b | | ~0 | ~0 | ~1 | | ~0 | ~1 | ~0 | | ~1 | ~0 | ~0 | | ~1 | ~1 | ~1 | |

**Задания для практического занятия:**

1 Доказать, что формула является тавтологией, без построения таблицы истинности.

1) (*A*→ *B*)→((*A*∨ *C*)→(*B* ∨ *C*)).

2) ((*A*→ *B*)∧ (*B*→*C*))→(*A*→*C*).

3) ((*A* ∧ *B*)↔ *B*)↔(*B*→ *A*).

4) (*A*→ *B*)↔(¬*B*→¬*A*).

5) ((*A*→ *B*)→ *A*)→ *A*.

6) ¬*A*→(*A*→ *B*) .

7) (¬*A*→ *B*)∨ ¬(*A*∧ *B*).

8) (*A*→ *B*)→(¬*B*→¬*A*).

9) (*A*→*C*)→((*B*→*C*)→(*A*∨ *B*→*C*)).

10) (*A*→ *B*)→((*A* ∧ *C*)→(*B* ∧ *C*)).

2 Доказать, что формулы логически эквивалентны.

1) *A*∧ (*B* ∨ *C*) и (*A*∧ *B*)∨ (*A*∧*C*).

2) *A*→ *B* и ¬*B*→¬*A*.

3) ¬¬*A* и *A* .

4) ¬(*A*∨ *B*) и ¬*A*∧¬*B* .

5) (*A*∧ *B*)∧*C* и *A*∧ (*B* ∧*C*).

6) *A*∨ (*B* ∧*C*) и (*A*∨ *B*)∧ (*A*∨ *C*).

7) ¬(*A*∧ *B*) и ¬*A*∨ ¬*B* .

8) (*A*∨ *B*)∨ *C* и *A*∨ (*B* ∨ *C*).

9) *A*→ *B* и ¬*A*∨ *B*.

10) *A* ∨ (*A* ∧ *B*) и *A*.

3 По данной таблице истинности запишите логическую функцию

|  |  |  |  |
| --- | --- | --- | --- |
| a | b | c | F(a,b,c) |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

4 Упростить логические выражения

1. (х∨ у)∧(х∧ у)=

2. А∧В∧А∧В∧С∨С∧А =

3. (А∨В)∧(А∧В∨В∧С)=

4. 

5. 

6. 

5 В каких из логических выражений значение эквивалентно 0?



6 В каких из логических выражений значение эквивалентно 1?



7 Какие из следующих логических функций эквивалентны Y?



1. Упростите выражение, применив законы логики:



1. Составить таблицы истинности по заданным формулам



**Контрольные вопросы**

1 Что изучает алгебра логики?

2 Что такое таблица истинности?

3 Перечислить основные законы алгебры логики

**Практическая работа № 4**

**«Составление линейной программы»**

**Цель работы:** научиться составлять линейные программы с использованием операторов ввода, вывода и присваивания.

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям;

- составлять и оформлять программы на языках программирования;

- тестировать и отлаживать программы.

знать:

- общие принципы построения и использования языков программирования, их классификацию;

- современные интегрированные среды разработки программ;

- процесс создания программ;

- стандарты языков программирования.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

Во всех приведенных примерах переменные могут быть типа REAL или INTEGER.

Кроме операторов, программа содержит описания переменных:

X: REAL;

Y, Z, T: INTEGER;

Каждая переменная, используемая в программе должна быть вписана в разделе описаний

VAR X: REAL;

Z, Y: INTEGER;

Схематически программа в Паскале выглядит следующим образом:

VAR A1;…; Ak;

BEGIN – начало

P1;…; Pn;

END. – конец – операционные скобки (служебные слова)

N – имя программы (идентификатор);

A1;…; Ak – описание переменных;

P1;…; Pn – операторы; выписываются в порядке следования друг за другом.

**;** - необходимо.

**Пример:** Программа ROOT1 вычисления корней квадратного уравнения ax2 + bx + c = 0, заданного коэффициентами a, b, c (предполагая, что a ≠ 0и что корни действительные)

VAR a, b, c, x1, x2: REAL;

BEGIN READ (a, b, c);

x1: = (- b + sqrt (sqr (b) – 4 \* a \* c) / (2 \* a));

x2: = (- b – sqrt (sqr (b) – 4 \* a \* c) / (2 \* a));

WRITE (x1, x2)

END.

**Задания для практического занятия:**

**1.** Найти значение переменной d:

Var a, b, c: integer; d: real;

Begin

Read (a);

B: = a\*a;

C: = a+b;

D: = c/a;

End.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 2 | 3 | 4 | 5 | 6 | 7 |
| d |  |  |  |  |  |  |

2. Разработать программу нахождения следующего значения согласно варианту:

№1 -a3c+b-2 №2 \* -

№3 \* tg *xy*

№4 - №5 №6 x - +

№7 ln| (y - )(x - ) №8 (1 – tg x)-2 + cos(x – y) №9

№10 ( )3 + 18xy2 №11 (1 + )4 - 12x2y

№12

№13 π + 16x cos(xy) – 2 №14 x-3 – cosx + sin(2xy)

№15 2ctg(3x) -

№16 |x2 – x3| - №17 x ⋅ ln x +

№18 sin - sin;

№19 *ex* - №20

№21 2ctg(3x) -

№22 *ex* – x – 2 + (1 + x)2  №23 3 – 4x + (y - )

№24 x- 10sinx + |x4 – x5|

№25 x – 10sinx + cos(x – y) №26 + x

№ 27.Найти площадь равнобедренной трапеции с основаниями *а* и *b* и углом а при большем основании *а.*

№ 28. Вычислить корни квадратного уравнения *ах2 + bх + с = 0,* заданного коэффициентами *a, b* и *с* (предполагается, что *а ≠* 0 и что дискриминант уравнения неотрицателен).

№ 29.Дано действительное число *х.* Не пользуясь никакими други­ми арифметическими операциями, кроме умножения, сложе­ния и вычитания, вычислить за минимальное число операций

2x4 - *Зх3 +* 4x2 - 5х + 6.

№ 30.Дано *х.* Получить значения *-2х + Зх2 -* 4х3 и

1 *+ 2х + 3x2 +* 4х3. Позаботиться об экономии операций,

№ 31.Найти площадь треугольника, две стороны которого равны *а* и b*,* а угол между этими сторонами равен *g.*

№ 32.Дано число *а.* Не используя никаких функций и никаких операций, кроме умножения, получить *а8* за три операции; *а10* и а16 за четыре операции.

№ 33.Найти сумму членов арифметической прогрессии, если из­вестны ее первый член, знаменатель и число членов прогрес­сии,

№ 34.Найти все углы треугольника со сторонами *а, b, с.* Предусмот­реть в программе перевод радианной меры угла в градусы, ми­нуты и секунды.

№ 35.Три сопротивления *R1 ,R2, R3* соединены параллельно. Найди­те сопротивление соединения.

№ 36.Составить программу для вычисления пути, пройденного лод­кой, если ее скорость в стоячей воде υ км/ч, скорость течения реки и км/ч, время движения по озеру t*1 ч,* а против течения реки — t2 ч.

**Контрольные вопросы**

1. Для чего используется указание типа данных величин?

2. Как описывается тип величины в языке Паскаль?

3. Охарактеризуйте целочисленные типы данных: какие они могут принимать значения, в каких операциях могут принимать участие, сколько места занимают в памяти. Какие стандартные функции определены для целых чисел?

4. Какие типы отношений определены над данными вещественного типа?

5. Чем отличаются вещественные числа от целых?

**Практическая работа № 5**

**«Применение условного оператора»**

**Цель работы:** ознакомиться с условным оператором Паскаля и научиться составлять программы с его использованием.

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям;

- составлять и оформлять программы на языках программирования;

- тестировать и отлаживать программы.

знать:

- общие принципы построения и использования языков программирования, их классификацию;

- современные интегрированные среды разработки программ;

- процесс создания программ;

- стандарты языков программирования.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

Для того, чтобы переменной max присвоилось наибольшее из значений переменных x1 и x2, то необходимо сравнить значения x1 и x2 и в зависимости от результата сравнения выполнить либо оператор max: = x1, либо max: = x2. Действия такого рода задаются условным оператором:

if B then P1 else P2,

где В – условие, P1 и P2 – операторы, if – если, then – то, else – иначе. Если В соблюдается, то выполняется P1, иначе выполняется P2.

В качестве условий используются отношения. Отношения представляют собой записи равенств и неравенств. Примеры отношений: a = b; d ≠ 0; sqr (b) – 4 \* a \* c > 0.

В общем случае отношение – это два выражения, разделенные одним из знаков =, <, >, ≠, ≤, ≥ (<>, >=, <=)

**Пример:** Написать программу нахождения наибольшего значения из двух заданных целых чисел.

VAR x1, x2, max: INTEGER;

BEGIN

WRITE ('x1, x2');

READ (x1, x2);

IF x1>x2 THEN max: = a1

ELSE max: = x2;

WRITE (max);

END.

Оператор, расположенный после ELSE, может быть любым оператором; оператор, расположенный между THEN ELSE не может быть условным.

**Задания для практического занятия:**

**1.** Найти значение переменной c:

If a>b then c: = (a+b)/2

Else

If a<b then c: = a\*a + b\*b

Else c: = a-b;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 5 | 2 | 3 | 6 | 1 | 5 |
| b | 3 | 3 | 3 | 2 | 4 | 5 |
| c |  |  |  |  |  |  |

**2. Разработать программу, выполняющую следующие действия:**

№ 1 Даны три действительные числа. Возвести в квадрат те из них, значения которых неотрицательны, и в четвертую сте­пень — отрицательные.

№ 2 Даны две точки *A(x1 ,у1)* и B*(х2, y2).* Составить алгоритм, оп­ределяющий, которая из точек находится ближе к началу координат.

№ 3 Даны два угла треугольника (в градусах). Определить, существует ли такой треугольник. Если да, то будет ли он прямоугольным.

№ 4 Даны действительные числа *х* и *у,* не равные друг другу. Меньшее из этих двух чисел заменить половиной их суммы, а большее — их удвоенным произведением.

№ 5 На плоскости *XOY* задана своими координатами точка *А.* Указать, где она расположена: на какой оси или в каком коор­динатном углу.

№ 6 Даны целые числа *т, п.* Если числа не равны, то заменить каждое из них одним и тем же числом, равным большему из исходных, а если равны, то заменить числа нулями.

№ 7 Дано трехзначное число *N.* Проверить, будет ли сумма его цифр четным числом.

№ 8 Определить, равен ли квадрат заданного трехзначного числа кубу суммы цифр этого числа.

№ 9 Определить, является ли целое число *N* четным двузначным числом.

№ 10 Определить, является ли треугольник со сторонами *а, b, с* рав­носторонним.

№11 Определить, является ли треугольник со сторонами *а, b, с* рав­нобедренным.

№ 12 Определить, имеется ли среди чисел а, b*, с* хотя бы одна пара взаимно противоположных чисел.

№ 13 Подсчитать количество отрицательных среди чисел а, b, с.

№14 Подсчитать количество положительных среди чисел а, b, с.

№ 15 Подсчитать количество целых чисел среди чисел а, b, с.

№ 16 Определить, делителем каких чисел а, b, сявляется число k.

№ 17 Услуги телефонной сети оплачиваются по следующему правилу: за разговоры до *А* минут в месяц оплачиваются *В* р., а разговоры сверх установленной нормы оплачиваются из расчета С р. в минуту. Написать программу, вычисляющую плату за пользование телефоном для введенного времени разговоров за месяц.

№ 18 Даны три стороны одного и три стороны другого треугольни­ка. Определить, будут ли эти треугольники равновеликими, т.е. имеют ли они равные площади.

№ 19 Программа-льстец. На экране появляется вопрос «Кто ты: мальчик или девочка? Введи Д иди М». В зависимости от ответа на экране должен появиться текст «Мне нравятся девочки!» или «Мне нравятся мальчики!».

№ 20 Грузовой автомобиль выехал из одного города в другой со скоростью υ1 км/ч. Через *t* ч в этом же направлении выехал легковой автомобиль со скоростью *υ2* км/ч. Составить программу, определяющую, догонит ли легковой автомобиль грузовой через *t1* ч после своего выезда.

№ 21 Перераспределить значения переменных *х* и *у* так, чтобы в *х* оказалось большее из этих значений, а в *у* — меньшее.

№ 22 Определить правильность даты, введенной с клавиатуры (число — от 1 до 31, месяц — от 1 до 12). Если введены некор­ректные данные, то сообщить об этом.

№ 23 Составить программу, определяющую результат гадания на ромашке — «любит — не любит», взяв за исходное данное ко­личество лепестков *п.*

№ 24 Написать программу нахождения суммы большего и меньше­го из 3 чисел.

№ 25 Написать программу, распознающую по длинам сторон среди всех треугольников прямоугольные. Если таковых нет, то вы­числить величину угла С.

№ 26 Найти max{min(a, b), min(c, d)}.

№ 27 Даны три числа а, b, с. Определить, какое из них равно d. Если ни одно не равно d, то найти max(d-a, d-b, d-c).

№ 28 Даны четыре точки А1(х1,у1), А2(х2,у2), аз(х3,у3), А4(х4,,у4). Определить, будут ли они вершинами параллелограмма.

№ 29 Даны три точки А(х1 ,y,), В(х2, у2 )и С(х3, у3). Определить, будут ли они расположены на одной прямой.

№ 30 Даны действительные числа а, b, с. Удвоить эти числа, если а<b<с, и заменить их абсолютными значениями, если это не так.

№ 31 На оси ОХ расположены три точки а, b, с. Определить, какая из точек b, с расположена ближе к а.

№ 32 Даны три положительных числа а, b, с. Проверить, могут ли они быть длинами сторон треугольника. Если да, то вычислить площадь этого треугольника.

№ 33 Написать программу решения уравнения ах3 + bх = 0 для произвольных а, b.

№ 34Дан круг радиуса R. Определить, поместится ли правильный треугольник со стороной а в этом круге.

№ 35 Написать программу, которая по заданным трем числам опре­деляет, является ли сумма каких-либо двух из них положи­тельной.

№ 36 Известно, что из четырех чисел а1 а2, а3 и а4 одно отлично от трех других, равных между собой; присвоить номер этого числа переменной n.

**Контрольные вопросы:**

1. Что такое оператор? Чем отличаются простые и структурные операторы?
2. Назначение, формы записи и порядок выполнения оператора условия if?
3. Как выглядит блок-схема программы с ветвлением?

**Практическая работа № 6**

**«Применение оператора цикла с предусловием»**

**Цель работы:** ознакомиться с оператором цикла с предусловием и научиться составлять программы с его использованием.

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям;

- составлять и оформлять программы на языках программирования;

- тестировать и отлаживать программы.

знать:

- общие принципы построения и использования языков программирования, их классификацию;

- современные интегрированные среды разработки программ;

- процесс создания программ;

- стандарты языков программирования.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

## ***Многократно повторяемые действия могут быть заданы операторами цикла.***

while условие do оператор

Выполняется этот оператор цикла так: проверяется условие, и если оно удовлетворяется, то выполняется оператор, а затем вновь проверяется условие и т.д. Как только на очередном шаге окажется, что условие не удовлетворяется, то выполнение оператора цикла прекращается.

while x ≤ 0 do x :=х+ 1, если значения х положительно, то выполнение оператора цикла прекратится после первой же проверки условия x ≤ 0, и значение переменной х не измениться. Если же значение х не положительно, то к этому значению будет добавляться по единице до тех пор, пока значение не станет положительным.

**Пример:**

Пусть даны числа a, b (a > 1) и надо получить все члены бесконечной последовательности a, a2, a3, … меньше числа b.

var a, b, c : real;

begin

read (a, b); c := a;

while c < b do

begin

writeln (c); c := c \* a

end

end.

Если a ≥ b, то не будет выведено ни одного члена последовательности a, a2, a3.

**Задания для практического занятия:**

**1.** Найти значение переменной c:

While a<b do

### begin

с:= c+a\*b;

а:= a+1;

еnd;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 2 | 1 | 3 | -1 | 0 | -2 |
| b | 4 | 3 | 5 | 2 | 3 | 1 |
| c | 3 | 2 | 1 | 5 | 4 | 3 |
| c |  |  |  |  |  |  |

**2.** Разработать программу, выполняющую следующие действия:

№ 1 Дано натуральное число N. Вычислить:

S = + + … +

№ 2 Дано натуральное число N. Вычислить произведение первых N сомножителей

P = • • • … •

№ 3 Дано натуральное число N. Вычислить:

• • … •

№ 4 Дано действительное число х. Вычислить:

x - + - + - +

№ 5 Даны натуральное n, действительное х. Вычислить:

S = sinx + sinsinx + … sinsin…sinx

n раз

№ 6 Даны действительное число а, натуральное число n. Вычис­лить:

Р = а (а + 1) х … х (а + n- 1)

№7Дандействительное число а, натуральное число n. Вычис­лить:

Р = а (а - n)(а – 2n) х ... х (а - n2).

№ 8 Даны действительное число а, натуральное число n. Вычис­лить:

S = + + + … +

№ 9 Даны натуральное n действительное х. Вычислить:

sinx + sinx2 +...+ sinn.

№ 10 Дано натуральное n. Вычислить:

S - 1 • 2 +2 - 3 • 4 + ... + n •(n + 1) х ... х 2n.

№ 11 Дано натуральное число n. Вычислить:

P = (1 - )(1 - )\*…\*(1 - ), где n>2

№ 12 Дано натуральное число n. Вычислить:

P = (1 - )(1 - )(1 - )\*…\*(1 - )

№ 13 Дано натуральное число n. Вычислить:

S= 1! + 2! + 3! + ... + n! (n>1).

№ 14 Дано натуральное число n. Вычислить:

S = + + + … +

№ 15 Дано натуральное n. Вычислить: у - 1 • 3 • 5 х ... х (2n - 1).

№ 16 Дано натуральное n. Вычислить: у - 2 • 4 • 6 x ... х (2n).

№ 17 Вычислить: у = cosx + cosx2 + cosx3 + ... + cosxn.

№ 18 Вычислить: у = sinl + sinl,l + sinl,2 + ... sin2.

№ 19 Дано натуральное число N. Вычислить:

S = 1 - + - + … + (-1)n \*

№ 20 Дано натуральное число N. Вычислить:

S = + +…+

№ 21 Дано натуральное число N. Вычислить произведение первых N сомножителей

P = • • • … • ,

№ 22 Дано натуральное число N. Вычислить:

• • … •

№ 23 Дано действительное число х. Вычислить:

x - + - + - +

№ 24 Даны натуральное n, действительное х. Вычислить:

S = sinx + sinsinx + … sinsin…sinx

n раз

№ 25 Даны действительное число а, натуральное число n. Вычис­лить:

S = + + + … +

№ 26 Дано действительное х. Вычислить:

№ 27 Вычислить: (1 + sin0,l)(l + sin0,2) х ... х (1 + sinl0).

**Контрольные вопросы:**

1. Каково назначение операторов повтора (цикла)?
2. Какие требования предъявляются к выражениям, управляющим повторениями?
3. В каких случаях используется оператор цикла с предусловием?

**Практическая работа № 7**

**«Применение оператора цикла с постусловием»**

**Цель работы:** ознакомиться с оператором цикла с постусловием и научиться составлять программы с его использованием.

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям;

- составлять и оформлять программы на языках программирования;

- тестировать и отлаживать программы.

знать:

- общие принципы построения и использования языков программирования, их классификацию;

- современные интегрированные среды разработки программ;

- процесс создания программ;

- стандарты языков программирования.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

repeat оператор until условие

Выполняется этот оператор так: выполняются операторы, проверяются условие, если оно не выполняется, то вновь выполняются операторы, если условие выполняются, – цикл заканчивается. Если условие выполняется с самого начала, то операторы выполняются один раз. Если условие никогда не выполняется, то группа операторов выполняется бесконечное число раз, происходит "зацикливание".

repeat c := c + 1/ x; x := x – 1

until x = 0;

Сначала выполняется оператор c := c + 1/ x; x := x – 1, затем проверяется условие х = 0. Если х ≠ 0, то проверяется выполнение указанных операторов, если х = 0, то управление передается на оператор, следующий за строкой until x = 0.

Пример: Вычислить сумму: 1 + 1/2+ 1/3 + … + 1/50 с использованием оператора цикла repeat …until.

var S : real; N : integer;

begin

S := 0; N := 1;

repeat S := S + 1 / N; N := N + 1;

until N > 50;

write (S)

end.

Результат S = 4,499…

**Задания для практического занятия:**

1.Найти значение переменной с:

Repeat

b:= b+1;

а:= a-b;

с:= a+b;

Until a≤0;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | **В5** | B6 |
| A | 10 | 9 | 8 | 7 | 6 | 5 |
| B | 5 | 4 | 3 | 3 | 2 | 1 |
| С |  |  |  |  |  |  |

**2.** Разработать программу, выполняющую следующие действия:

№ 1 Дано натуральное число N. Вычислить:

S = + + … +

№ 2 Дано натуральное число N. Вычислить произведение первых N сомножителей

P = • • • … •

№ 3 Дано натуральное число N. Вычислить:

• • … •

№ 4 Дано действительное число х. Вычислить:

x - + - + - +

№ 5 Даны натуральное n, действительное х. Вычислить:

S = sinx + sinsinx + … sinsin…sinx

n раз

№ 6 Даны действительное число а, натуральное число n. Вычис­лить:

Р = а (а + 1) х … х (а + n- 1)

№7Дандействительное число а, натуральное число n. Вычис­лить:

Р = а (а - n)(а – 2n) х ... х (а - n2).

№ 8 Даны действительное число а, натуральное число n. Вычис­лить:

S = + + + … +

№ 9 Даны натуральное n действительное х. Вычислить:

sinx + sinx2 +...+ sinn.

№ 10 Дано натуральное n. Вычислить:

S - 1 • 2 +2 - 3 • 4 + ... + n •(n + 1) х ... х 2n.

№ 11 Дано натуральное число n. Вычислить:

P = (1 - )(1 - )\*…\*(1 - ), где n>2

№ 12 Дано натуральное число n. Вычислить:

P = (1 - )(1 - )(1 - )\*…\*(1 - )

№ 13 Дано натуральное число n. Вычислить:

S= 1! + 2! + 3! + ... + n! (n>1).

№ 14 Дано натуральное число n. Вычислить:

S = + + + … +

№ 15 Дано натуральное n. Вычислить: у - 1 • 3 • 5 х ... х (2n - 1).

№ 16 Дано натуральное n. Вычислить: у - 2 • 4 • 6 x ... х (2n).

№ 17 Вычислить: у = cosx + cosx2 + cosx3 + ... + cosxn.

№ 18 Вычислить: у = sinl + sinl,l + sinl,2 + ... sin2.

№ 19 Дано натуральное число N. Вычислить:

S = 1 - + - + … + (-1)n \*

№ 20 Дано натуральное число N. Вычислить:

S = + +…+

№ 21 Дано натуральное число N. Вычислить произведение первых N сомножителей

P = • • • … • ,

№ 22 Дано натуральное число N. Вычислить:

• • … •

№ 23 Дано действительное число х. Вычислить:

x - + - + - +

№ 24 Даны натуральное n, действительное х. Вычислить:

S = sinx + sinsinx + … sinsin…sinx

n раз

№ 25 Даны действительное число а, натуральное число n. Вычис­лить:

S = + + + … +

№ 26 Дано действительное х. Вычислить:

№ 27 Вычислить: (1 + sin0,l)(l + sin0,2) х ... х (1 + sinl0).

Контрольные вопросы:

1. Каково назначение операторов повтора (цикла)?
2. Какие требования предъявляются к выражениям, управляющим повторениями?
3. В чём отличия операторов повтора while и repeat?

**Практическая работа № 8**

**«Применение оператора цикла с параметром»**

**Цель работы:** ознакомиться с оператором цикла с параметром и научиться составлять программы с его использованием.

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям;

- составлять и оформлять программы на языках программирования;

- тестировать и отлаживать программы.

знать:

- общие принципы построения и использования языков программирования, их классификацию;

- современные интегрированные среды разработки программ;

- процесс создания программ;

- стандарты языков программирования.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

for i := число 1 to число 2 do оператор

i – переменная типа integer, которая называется параметром цикла. Такой оператор цикла

заставляет i принимать значения число 1, число 1 + 1, число 1 + 2, …, число 2. Для каждого из

значения выполняется оператор. Если число 1 > число 2, оператор не будет выполнен не разу.

S = 0;

for i := 1 to n do S := S + i \* i \* i

После выполнения S = 13 + 23 + … + n3.

**Пример:** Программа вычисления n!.

Применяется, когда число шагов легко определить по исходным данным.

var n, i, p : integer;

begin read (n); p := 1;

for i := 1 to n do p := p \* i;

write (n'! = ', p)

end.

**Задания для практического занятия:**

**1.** Найти значение переменной с:

For i:=1 to n do

с:= c+a\*b;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| а | 2 | 3 | 2 | 5 | 2 | 1 |
| b | 4 | 2 | 3 | 2 | 5 | 2 |
| C | 3 | 4 | 4 | 1 | 1 | 5 |
| n | 3 | 4 | 3 | 4 | 3 | 4 |
| с |  |  |  |  |  |  |

**2.** Разработать программу, выполняющую следующие действия:

№ 1 Дано натуральное число N. Вычислить:

S = 1 - + - + … + (-1)n \*

№ 2 Дано натуральное число N. Вычислить:

S = + +…+

№ 3 Дано натуральное число N. Вычислить произведение первых N сомножителей

P = • • • … • ,

№ 4 Дано натуральное число N. Вычислить:

• • … •

№ 5 Дано действительное число х. Вычислить:

x - + - + - +

№ 6 Даны натуральное n, действительное х. Вычислить:

S = sinx + sinsinx + … sinsin…sinx

n раз

№ 7 Даны действительное число а, натуральное число n. Вычис­лить:

S = + + + … +

№ 8 Дано действительное х. Вычислить:

№ 9 Вычислить: (1 + sin0,l)(l + sin0,2) х ... х (1 + sinl0).

№ 10 Дано натуральное число N. Вычислить:

S = + + … +

№ 11 Дано натуральное число N. Вычислить произведение первых N сомножителей

P = • • • … •

№ 12 Дано натуральное число N. Вычислить:

• • … •

№ 13 Дано действительное число х. Вычислить:

x - + - + - +

№ 14 Даны натуральное n, действительное х. Вычислить:

S = sinx + sinsinx + … sinsin…sinx

n раз

№ 15 Даны действительное число а, натуральное число n. Вычис­лить:

Р = а (а + 1) х … х (а + n- 1)

№ 16 Даны действительное число а, натуральное число n. Вычис­лить:

Р = а (а - n)(а – 2n) х ... х (а - n2).

№ 17 Даны действительное число а, натуральное число n. Вычис­лить:

S = + + + … +

№ 18 Даны натуральное n действительное х. Вычислить:

sinx + sinx2 +...+ sinn.

№ 19 Дано натуральное n. Вычислить:

S - 1 • 2 +2 - 3 • 4 + ... + n •(n + 1) х ... х 2n.

№ 20 Дано натуральное число n. Вычислить:

P = (1 - )(1 - )\*…\*(1 - ), где n>2

№ 21 Дано натуральное число n. Вычислить:

P = (1 - )(1 - )(1 - )\*…\*(1 - )

№ 22 Дано натуральное число n. Вычислить:

S= 1! + 2! + 3! + ... + n! (n>1).

№ 23 Дано натуральное число n. Вычислить:

S = + + + … +

№ 24 Дано натуральное n. Вычислить: у - 1 • 3 • 5 х ... х (2n - 1).

№ 25 Дано натуральное n. Вычислить: у - 2 • 4 • 6 x ... х (2n).

№ 26 Вычислить: у = cosx + cosx2 + cosx3 + ... + cosxn.

№ 27 Вычислить: у = sinl + sinl,l + sinl,2 + ... sin2.

**Контрольные вопросы:**

1. В каких случаях предпочтительнее использовать для организации циклов оператор повтора for? Что записывается в заголовке этого оператора?
2. Каким образом в операторе цикла for описывается направление изменения значения параметра цикла?
3. Какие ограничения налагаются на использование управляющей переменной (параметра цикла) в цикле for?

**Практическая работа № 9**

**«Программирование массивов»**

**Цель работы:** ознакомиться с основными приемами действий над массивами.

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям;

- составлять и оформлять программы на языках программирования;

- тестировать и отлаживать программы.

знать:

- общие принципы построения и использования языков программирования, их классификацию;

- современные интегрированные среды разработки программ;

- процесс создания программ;

- стандарты языков программирования.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

В Турбо Паскале могут использо­ваться объекты, содержащие множество однотипных элементов. Это **массивы** *-*формальное объединение нескольких однотипных объектов (чисел, символов, строк и т.п.), рассматриваемое как единое целое. К необходимости применения массивов мы приходим всякий раз, когда требуется связать и использовать целый ряд родственных величин. Например, результаты многократных замеров температуры воздуха в течение года удобно рассматривать как совокупность вещественных чисел, объединенных в один сложный объект - массив измерений.

При описании массива необходимо указать общее число входящих в массив эле­ментов и тип этих элементов. Например:

**Var** А : array [1..10] of Real; b : array [0..50] of Char; с : array [-3..4] of Boolean;

При описании массива используются зарезервированные слова ARRAY и OF (массив,из).За словом ARRAY в квадратных скобках указывается тип-диапазон, с помощью которого компилятор определяет общее число элементов массива. Тип-диапазон задается левой и правой границами изменения индекса массива, так что массив Асостоит из 10 элементов, массив В *-* из 51, а массив С *-* из 8 элементов. За словом ОF указывается тип элементов, образующих массив.

**Пример.** В массиве хранятся данные о выпавших осадках за последние 20 лет. Найти среднее значение и отклонения от среднего для каждого года.

**Var** А : array [1..20] of Real; I: integer; S: real;

Begin

{‘Ввод массива”}

for I:=1 to 20 do read(A[I]);

{‘Нахождение среднего значения”}

for I:=1 to 20 do S:=S+A[I];

S:=S/20;

{‘Нахождение отклонения от среднего ”}

for I:=1 to 20 do write(abs(A[I]-S));

end.

**Задания для практического занятия:**

**1.** Найти значение переменной S:

For i:=1 to n do

Begin

A [i]:=i;

S:=S+A [i];

End;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| n | 5 | 3 | 4 | 5 | 8 | 6 |
| i | 0 | 5 | 10 | 5 | 0 | 1 |
| S |  |  |  |  |  |  |

**2**. Разработать программу, выполняющую следующие действия:

№1. Дан массив из 100 целых чисел. Написать программу нахождения суммы положительных элементов массива.

№2. Дан массив из 100 целых чисел. Написать программу нахождения произведения положительных элементов массива.

№3. Дан массив из 100 целых чисел. Написать программу нахождения суммы четных элементов массива.

№4. Дан массив из 100 целых чисел. Написать программу нахождения произведения четных элементов массива.

№5. Дан массив из 100 целых чисел. Написать программу нахождения абсолютного значения суммы отрицательных элементов массива.

№6. Дан массив из 100 целых чисел. Написать программу нахождения произведения отрицательных элементов массива

№7. Дан массив из 100 целых чисел. Написать программу нахождения количества положительных элементов массива.

№8. Дан массив из 100 целых чисел. Написать программу нахождения количества отрицательных элементов массива.

№9. Дан массив из 100 целых чисел. Написать программу нахождения количества четных элементов массива.

№10. Дан массив из 100 целых чисел. Написать программу замены всех отрицательных элементов средним значением массива.

№11. Дан массив из 100 целых чисел. Написать программу нахождения суммы нечетных элементов массива.

№12. Дан массив из 100 целых чисел. Написать программу нахождения произведения нечетных элементов массива.

№13. Дан массив из 100 целых чисел. Поменять местами наибольший и наименьший элементы массива.

№14. Дан массив из 100 целых чисел. Написать программу замены всех положительных элементов средним значением массива.

№15. Дан массив из 100 целых чисел. Написать программу нахождения суммы элементов массива с четными индексами.

№16. Дан массив из 100 целых чисел. Написать программу нахождения произведения элементов массива с четными индексами.

№17. Дан массив из 100 целых чисел. Написать программу нахождения суммы элементов массива с нечетными индексами.

№18. Дан массив из 100 целых чисел. Написать программу нахождения произведения элементов массива с нечетными индексами.

№19. Дан массив из 100 целых чисел. Написать программу нахождения среднего значения среди положительных элементов массива

№20. Дан массив из 100 целых чисел. Написать программу замены всех отрицательных элементов значением их произведения.

№21. Дан массив из 100 целых чисел. Заменить первые К элементов на противоположные по знаку.

№22. Дан массив из 100 целых чисел. Заменить максимальный по модулю отрицательный элемент нулем.

№23. Дан массив из 100 целых чисел. Заменить минимальный по модулю положительный элемент нулем.

№24. Дан массив из 100 целых чисел. Умножить все элементы, кратные трем, на третий элемент массива.

№25. Дан массив из 100 целых чисел. Заменить элементы между минимальным и максимальным нулем.

**Контрольные вопросы:**

1. Что такое массив?
2. Как определить местоположение элемента в массиве?
3. Что такое индекс? Каким требованиям он должен удовлетворять?
4. Особенности расположения элементов массива в памяти ЭВМ. Особенности расположения в памяти элементов многомерных массивов.
5. Каким образом задаётся описание массива, что в нём указывается?

**Практическая работа № 10**

**«Программирование матриц»**

**Цель работы:** ознакомиться с основными приемами действий над двумерными массивами (матрицами).

**Образовательные результаты, заявленные во ФГОС третьего поколения:**

Студент должен

уметь:

- формализовать поставленную задачу;

- применять полученные знания к различным предметным областям;

- составлять и оформлять программы на языках программирования;

- тестировать и отлаживать программы.

знать:

- общие принципы построения и использования языков программирования, их классификацию;

- современные интегрированные среды разработки программ;

- процесс создания программ;

- стандарты языков программирования.

**Краткие теоретические и учебно-методические материалы по теме практической работы**

Объектами обработки некоторых алгоритмов нередко являются прямоугольные таблицы данных, называемые иначе **двумерными массивами.** Примером такого массива может быть таблица Пифагора – квадратная таблица А из 10 строк и 10 столбцов, каждый элемент которой определяется формулой аij = i\*j (i,j=1,2,…,10). Положение элемента аij в таблице задаётся двойным индексом: первый (i) обозначает номер строки элемента, а второй (j) – номер его столбца, на пересечении которых стоит элемент (нумерация строк производится сверху вниз, а столбцов –

слева на право). К двумерным массивам относятся также таблицы соревнований, коэффициенты систем уравнений, шахматные доски, таблицы распределения температур некоторого тела и т.д. Реже на практике встречаются массивы, содержащие три и более измерений.

Объявление двумерных массивов пишется аналогично одномерным, но для прямоугольной таблицы надо указать границы изменения индексов как по горизонтали, так и по вертикали. Например, объявим таблицу А, содержащую целые числа и состоящую из 9 строк и 10 столбцов. Вот это объявление:

Var А: array [1..9, 1..10] of integer;

**Пример:** Дана квадратная матрица размерностью 10х10, состоящая из целых чисел. Найти произведение элементов главной диагонали матрицы.

**Var A:** array[1..10,1..10] of integer; i,j,P: integer;

Begin

{‘Ввод матрицы’}

for i:=1 to 10 do

for j:=1 to 10 do read(A[i,j]);

{‘ Нахождение произведения элементов главной диагонали матрицы’ }

P:=1;

for i:=1 to 10 do

for j:=1 to 10 do

if i=j then P:=P\*A[i,j];

write(P);

end.

**Задания для практического занятия:**

Разработать программу, выполняющую следующие действия:

№1. В целочисленной матрице размерностью 10\*10 найти наибольший элемент и номер строки, в которой он находится.

№2. В целочисленной матрице размерностью 10\*10 найти наименьший элемент и номер столбца, в которой он находится.

№3. В целочисленной матрице размерностью 10\*10 найти номер столбца содержащего максимальный элемент матрицы.

№4. В целочисленной матрице размерностью 10\*10 найти номер строки, содержащей минимальный элемент матрицы.

№5. В целочисленной матрице размерностью 10\*10 найти номер строки, содержащей наибольшее среднее значение.

№6. В целочисленной матрице размерностью 10\*10 найти номер столбца, содержащего наибольшее среднее значение.

№7. В целочисленной матрице размерностью 10\*10 найти номер строки, содержащей наименьшее среднее значение.

№8. В целочисленной матрице размерностью 10\*10 найти номер столбца, содержащего наименьшее среднее значение.

№9. В целочисленной матрице размерностью 10\*10 найти разность между наибольшим и наименьшим элементами матрицы.

№10. В целочисленной матрице размерностью 10\*10 поменять местами наибольшие и наименьшие элементы матрицы.

№11. В целочисленной матрице размерностью 10\*10 найти номер строки, содержащей наибольшее произведение её элементов.

№12. В целочисленной матрице размерностью 10\*10 найти номер строки, содержащей наименьшее произведение её элементов.

№13. В целочисленной матрице размерностью 10\*10 найти номер столбца, содержащего наибольшее произведение элементов.

№14. В целочисленной матрице размерностью 10\*10 найти номер столбца, содержащего наибольшее произведение его элементов.

№15. В целочисленной матрице размерностью 10\*10 найти сумму элементов четных строк.

№16. В целочисленной матрице размерностью 10\*10 найти произведение элементов четных строк.

№17. В целочисленной матрице размерностью 10\*10 найти сумму элементов нечетных строк.

№18. В целочисленной матрице размерностью 10\*10 найти произведение элементов нечетных строк.

№19. В целочисленной матрице размерностью 10\*10 найти произведение элементов нечетных столбцов.

№20. В целочисленной матрице размерностью 10\*10 найти сумму элементов нечетных столбцов.

№21. В целочисленной матрице размерностью 10\*10 найти сумму элементов четных столбцов.

№22. В целочисленной матрице размерностью 10\*10 найти произведение элементов четных столбцов.

№23. В целочисленной матрице размерностью 10\*10 найти сумму положительных элементов четных строк.

№24. В целочисленной матрице размерностью 10\*10 найти сумму положительных элементов нечетных строк.

№25. В целочисленной матрице размерностью 10\*10 найти сумму отрицательных элементов четных строк.

№26. В целочисленной матрице размерностью 10\*10 найти сумму отрицательных элементов нечетных строк.

№27. В целочисленной матрице размерностью 10\*10 найти произведение положительных элементов четных столбцов.

№28. В целочисленной матрице размерностью 10\*10 найти произведение отрицательных элементов нечетных столбцов.

№29. В целочисленной матрице размерностью 10\*10 определить, каких элементов: положительных или отрицательных больше.

№30. В целочисленной матрице размерностью 10\*10 определить, каких элементов: четных или нечетных больше.

**Контрольные вопросы:**

1.Что такое матрица?

2.Как определить местоположение элемента в матрице?

3.Что такое индекс? Каким требованиям он должен удовлетворять?

4.Особенности расположения в памяти элементов многомерных массивов.

5.Каким образом задаётся описание матрицы, что в нём указывается?

**Уфимский государственный колледж радиоэлектроники**

**КОНТРОЛЬНАЯ РАБОТА**

по дисциплине «Основы алгоритмизации и программирования»

Вариант\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Выполнил студент (ка) гр \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Ф.И.О. студента)

«\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_ г.

(дата сдачи)

Проверил: Бронштейн М.Е.

(Ф.И.О. преподавателя)

«\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_ г.

(дата проверки)

#### Контрольные задания

Исходные данные для расчета взять из таблиц, согласно варианту.

**Задание 1.** Найти значение переменной d:

var a, b, c: integer; d: real;

# Begin

Read (a);

b: = a\*a;

c: = a+b;

d: = c/a;

End.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 2 | 3 | 4 | 5 | 6 | 7 |
| d |  |  |  |  |  |  |

**Задание 2.** Найти значение переменной c:

if a>b then c: = (a+b) / 2

else

if a<b then c: = a\*a + b\*b

else c: = a-b;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 5 | 2 | 3 | 6 | 1 | 5 |
| b | 3 | 3 | 3 | 2 | 4 | 5 |
| c |  |  |  |  |  |  |

**Задание 3.** Найти значение переменной c:

if (a<b) and (b<c) then d:= c-a+b

else

if (a>b) and (b>c) then d:= a-b+c

else

if (a>b) and (b<c) then d:= a-c+b

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 2 | 5 | 5 | 4 | 6 | 2 |
| b | 3 | 3 | 2 | 3 | 3 | 5 |
| c | 4 | 1 | 4 | 5 | 1 | 7 |
| d |  |  |  |  |  |  |

**Задание 4.** Найти значение переменной c:

while a<b do

### begin

c:= c + a\*b;

a:= a+1;

end;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 2 | 1 | 3 | -1 | 0 | -2 |
| b | 4 | 3 | 5 | 2 | 3 | 1 |
| c | 3 | 2 | 1 | 5 | 4 | 3 |
| c |  |  |  |  |  |  |

Задание 5. Найти значение переменной с:

repeat

b:= b+1;

a:= a-b;

c:= a+b;

until a≤0;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 10 | 9 | 8 | 7 | 6 | 5 |
| b | 5 | 4 | 3 | 3 | 2 | 1 |
| c |  |  |  |  |  |  |

**Задание 6.** Найти значение переменной с:

For i:=1 to n do

C:= c + a\*b;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| a | 2 | 3 | 2 | 5 | 2 | 1 |
| b | 4 | 2 | 3 | 2 | 5 | 2 |
| c | 3 | 4 | 4 | 1 | 1 | 5 |
| n | 3 | 4 | 3 | 4 | 3 | 4 |
| c |  |  |  |  |  |  |

**Задание 7.** Найти значение переменной S:

for i:=1 to n do

begin

A [i]: = i;

S:= S + A [i];

end;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | B1 | B2 | B3 | B4 | B5 | B6 |
| n | 5 | 3 | 4 | 5 | 8 | 6 |
| i | 0 | 5 | 10 | 5 | 0 | 1 |
| S |  |  |  |  |  |  |

**Задание 8:**

A-множество букв имени студента

B-множество букв фамилии студента

C-множество букв отчества студента

var d:char, a,b,c: set of char;

for d:= 'a' to 'z' do

If in a + b then write (d);

Какая последовательность символов будет выведена на экран?

**Задание 9:**

Вариант 1.

Создать комбинированный тип LIBRARY (библиотека), полями которого являются: фамилия писателя, его инициалы, название книги, год ее издания.

Вариант 2.

Создать комбинированный тип SESSION (сессия), полями которого являются: фамилия студента, его инициалы, название дисциплин, оценки за экзамены.

Вариант 3.

Создать комбинированный тип VIDEO, полями которого являются: название фильма, год его выхода, фамилия режиссера.

Вариант 4.

Создать комбинированный тип PERSONAL, полями которого являются: фамилия

сотрудника, его инициалы, название должности, номер отдела, размер оклада.

Вариант 5.

Создать комбинированный тип HOSPITAL (больница), полями которого являются:

фамилия больного, его инициалы, его возраст, диагноз, фамилия лечащего врача.

Вариант 6.

Создать комбинированный тип SUPERMARKET, полями которого являются: наименование товара, дата его поставки, цена, наименование поставщика.

**Литература**

Основные источники:

1. Попов В.Б. Turbo Pascal для школьников – М.: Финансы и статистика, 2011

2. Фаронов В.В. Турбо Паскаль 7.0. Учебный курс: учебное пособие. – М.: КНОРУС, 2012

3. Немнюгин С.А. Turbo Pascal. Программирование на языке высокого уровня – М., СПб: Питер, 2011

4. Попов И.И., Партыка Т.Л. Языки программирования - ООО Издательство «Форум», 2013

Дополнительные источники:

1. Камаев В.А., Костерин В.В. Технологии программирования - М.: Высшая школа, 2012

2. Семакин И.Г., Шестаков А.П. Основы программирования - ОИЦ «Академия», 2011

Интернет ресурсы:

1. Система федеральных образовательных порталов Информационно-коммуникационные технологии в образовании. [Электронный ресурс] – режим доступа: <http://www.ict.edu.ru> (2003-2015)