**Вопрос №2**

**1. Что такое алгоритм?**

*Алгоритм* – это четкое описание последовательности действий, которые необходимо выполнить для получения результата.

**2. Перечислите базовые структуры алгоритма.**

*Базовые структуры алгоритма* - это структуры, с помощью которых создается алгоритм для решения определенной задачи.

***Алгоритмические структуры***

1) Линейная 2) Разветвленная 3) Циклическая

*Линейный алгоритм* - обеспечивает получение результата путем однократного выполнения последовательности действий, независимо от входных данных и промежуточных результатов.

*Разветвленный алгоритм* - выбор действий в случае выполнения или невыполнения заданного условия.

*Циклический алгоритм* - повторения некоторой серии команд.

**3. Назовите основные свойства алгоритма.**

*К основным свойствам алгоритма относят:*

1) *Дискретность* – алгоритм должен представлять процесс решения задачи как последовательное исполнение простых действий.

2) Определенность — каждое правило алгоритма должно быть четким, однозначным. Благодаря этому свойству выполнение алгоритма носит механический характер и не требует никаких дополнительных указаний или сведений о решаемой задаче.

3)Результативность (конечность) — алгоритм должен приводить к решению задачи за конечное число шагов.

4) Массовость — алгоритм решения задачи должен быть применим для некоторого класса задач, различающихся только исходными данными. При этом исходные данные могут выбираться из некоторой области, которая называется областью применимости алгоритма.

**4. Что такое идентификатор применительно к языкам программирования?**

***Идентификаторы***

1) Стандартные 2) Пользовательские

*Стандартные идентификаторы* - используются для обозначения определенных разработчиками языка функций, констант. Например: sqrt, abs и др.

*Пользовательские идентификаторы* - применяют для обозначения имён программ, переменных, констант и других объектов, определенных в программе. Имя может состоять из любого количества букв или цифр (максимальная длина – 127 символов), но должно начинаться с буквы или символа подчёркивание.

**5. Что такое переменная применительно к языкам программирования?**

*Переменная* - это именованная область памяти для хранения данных, которые могут изменяться в процессе исполнения программы.

*Переменная характеризуется:*

1) Именем («обозначением ячейки памяти»)

2) Значением (данными, содержащимися в переменной в конкретный момент времени)

3) Типом определяющим:

а) какие значения может принимать переменная;

б) какие операции можно производить с этими значениями;

в) как данные представлены в памяти компьютера.

**6. Что такое константа применительно к языкам программирования?**

*Константа* – именованный элемент программы, который не меняет свое значение. Перед употреблением должна быть объявлена в разделе констант const.

**7. Что такое массив применительно к языкам программирования?**

*Массив* — структура данных в виде набора однотипных элементов, расположенных в памяти непосредственно друг за другом.

**8. Что такое подпрограмма?**

*Подпрограмма* — поименованная часть компьютерной программы, содержащая описание определённого набора действий.

Подпрограмма может быть многократно вызвана из разных частей программы.

**9. Как обозначаются в языке Паскаль операции «И», «ИЛИ», «НЕ», «ИСКЛЮЧАЮЩЕЕ ИЛИ»?**

and - «И»

or - «ИЛИ»

not - «НЕ»

xor - «ИСКЛЮЧАЮЩЕЕ ИЛИ»

**10. Как обозначаются в языке Паскаль операции целочисленного деления и взятие остатка от деления?**

div – целочисленное деление

mod – взятие остатка от деления

**11. Как описать в языке паскаль массив A из 100 вещественных элементов?**

var

A: array [1..100] of Real;

**или**

const

N = 5;

type

mas = array [1…N] of Real;

var

A: mas;

**12. Как описать в языке Паскаль двумерный массив A размером 10 на 20 логических элементов?**

Const

N = 10;

M = 20;

Type

vector = array [1..N] of Boolean;

matrix= array [1..M] of vector;

Var

A: matrix;

**13. Как описать в языке паскаль строку из 100 символов?**

Const

N = 10;

Var

s: string[N];

N - максимально возможная длина строки - целое число в диапазоне 1..255. Если этот параметр опущен, то по умолчанию он принимается равным 255.

**14. Синтаксис оператора IF в языке Паскаль?**

**IF <условие> THEN <оператор 1>**

**ELSE <оператор 2>,**

где <условие> - выражение логического типа; <оператор 1>, <оператор 2> - любые операторы.

**15. Синтаксис оператора FOR в языке Паскаль?**

**FOR <счётчик цикла>:=<нач. зн.> TO <кон. зн.> DO <оператор>**

где **<счётчик цикла>** - переменная порядкового типа, чаще всего **INTEGER**.

**FOR <счётчик цикла>:=<нач. зн.> DOWNTO <кон. зн.> DO <оператор>**

Здесь цикл будет повторяться, пока **<счётчик цикла> ≥ <кон. зн.>.**

**16. Синтаксис оператора WHILE…DO в языке Паскаль?**

Это цикл с предварительной проверкой условия повторения цикла. Его структура:

**WHILE <условие> DO <оператор>**

Пока проверяемое условие справедливо (TRUE) то выполняется оператор, стоящий после слова DO. Если проверяемое условие равно FALSE, то оператор WHILE заканчивает свою работу.

**17. Синтаксис оператора REPEAT…UNTIL в языке Паскаль?**

Это цикл с постусловием. Он имеет следующую структуру:

**REPEAT**

**<Тело цикла>**

**UNTIL <условие>**

Все операторы, входящие в тело цикла, повторяются, пока не выполнится условие выхода из цикла, стоящее после слова UNTIL. После этого цикл заканчивает свою работу. В тело цикла можно включать любое количество операторов, не оформляя их, как составной оператор.

**18. Синтаксис оператора CASE в языке Паскаль?**

Оператор выбора позволяет выбрать одно из нескольких возможных продолжений программы. Его структура:

**CASE <ключ выбора> OF**

**<список выбора>**

**ELSE <оператор>**

**END**

где **<ключ выбора>** - выражение порядкового типа (обычно целого или символьного); **<список выбора>** - одна или более конструкций следующего вида: **<константа выбора> : <оператор>**, где **<константа выбора>** - константа того же типа, что и выражение **<ключ выбора>**.

**19. Как описать запись в языке Паскаль?**

Объявление структуры записи производится в разделе описания типов TYPE или переменных VAR.

**VAR**

< имя\_записи > : **RECORD**

< список полей >

**END**

В описании указывают имя каждого поля и его тип:

STUD : **RECORD**

FAM,IM,OTCH:STRING[20];

DAY,MONTH,YEAR:WORD;

**END**;

**20. Для чего используется раздел VAR в языке Паскаль?**

*Var* - обозначает начало блока определения переменных. Всё что будет идти за этим словом, будет рассматриваться компилятором, как объявление переменных.

*Формат:*

var <идентификатор > : <тип>;

**21. Для чего используется раздел CONST в языке Паскаль?**

*Раздел описания CONST позволяет ввести в программу*[*идентификаторы*](http://physics.herzen.spb.ru/library/03/01/pp/TPHelp/alphabet.htm#Идентификаторы (имена))*, являющиеся представителями определенных значений.*

CONST

ИмяКонстанты1 = Значение1;

ИмяКонстанты2 = Значение2;

**или**

ИмяКонстанты = ЗначениеВыражения;

*Примеры описания констант:*

CONST

Min = 0; { константа - целое число }

Max = 100;{ константа - целое число }

e = 2.7; { константа - вещественное число }

SpecChar = '\'; {константа - символ }

HelpStr = 'Нажмите клавишу F1'; {константа - строка }

OK = True; { логическая константа "истина" }

**22. Для чего используется раздел TYPE в языке Паскаль?**

*Для создания пользовательских типов в программе, объявленныхется в разделе TYPE по формату:*

TYPE

[имя] = [тип]

Программист может создавать свои типы произвольной структурой сложности на основе стандартных типов, либо уже определенных пользователем типов. Количество создаваемых типов неограниченно.

**Пример:**

TYPE

Matr = аггау[1..10] of integer;

**23. Для чего используется раздел LABEL в языке Паскаль?**

*В разделе Label объявляются метки, которые помечают операторы в соответствующей операторной части.*

*Синтаксис:*

**label идентификатор, ..., идентификатор**;

Каждая метка должна помечать только один оператор.

В дополнение к идентификаторам, в качестве метки также можно использовать последовательность цифр в диапазоне от 0 до 9999.

Передать управление к метке можно с помощью оператора goto.

*Пример*

label 1, 2;

begin

goto 1;

.

.

1 : WriteLn('Ненормальное завершение');

2 : WriteLn('Нормальное завершение');

end.

**24. С помощью какого оператора языка Паскаль можно ввести значение переменной с клавиатуры?**

*Общий вид:*

**Read(переменная, переменная...)**

**ReadLn(переменная, переменная...)**

При выполнении процедуры Read ожидается ввод перечисленных в скобках значений. Вводимые данные нужно отделить друг от друга пробелами. Присваивание значений идет по очереди.

Часто процедуру ReadLn без параметров применяют в конце программы для задержки: до нажатия на клавишу <Enter> результат выполнения программы остается на экране.

**25. С помощью какого оператора языка Паскаль можно вывести значение переменной с клавиатуры?**

*Общий вид:*  
  **Write (список выражений)  
  WriteLn (список выражений)**

Процедуры Write и WriteLn используются не только для вывода результата, но и для вывода различных сообщений или запросов.

Оператор WriteLn можно применить и без параметров. В этом случае курсор просто будет переведен в начало следующей строки.