

Тема 3. Технология разработки программ

Лекция 1

Подготовка прикладных задач к решению на ЭВМ

- Процесс подготовки** включает следующие этапы:
1. постановку (формулировку) задачи;
 2. выбор метода решения задачи;
 3. разработку алгоритма решения задачи;
 4. составление программы;
 5. отладку программы;
 6. автоматическое решение задачи.

Основы алгоритмизации

Алгоритмом называется понятное и точное предписание (указание) исполнителю совершить определенную последовательность действий для достижения поставленной цели.

Основные свойства алгоритмов следующие:

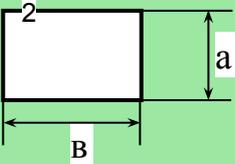
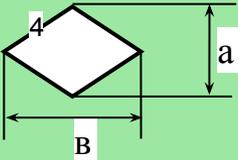
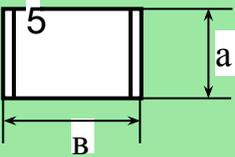
Дискретность (прерывность) – т. е. алгоритм должен представлять процесс решения задачи как последовательное выполнение простых шагов;

Определённость – каждое действие (команда) алгоритма должно быть чётким, однозначным и не оставлять повода для иного произвольного толкования;

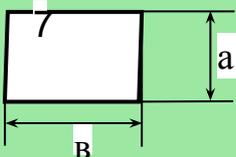
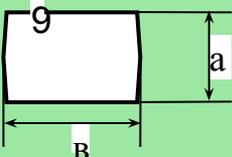
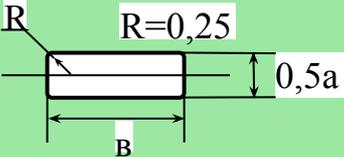
Результативность (конечность) – алгоритм должен приводить к решению задачи за конечное число шагов;

Массовость – алгоритм разрабатывается в общем виде, т. е. он должен быть применим для некоторого класса задач, различающихся лишь исходными данными.

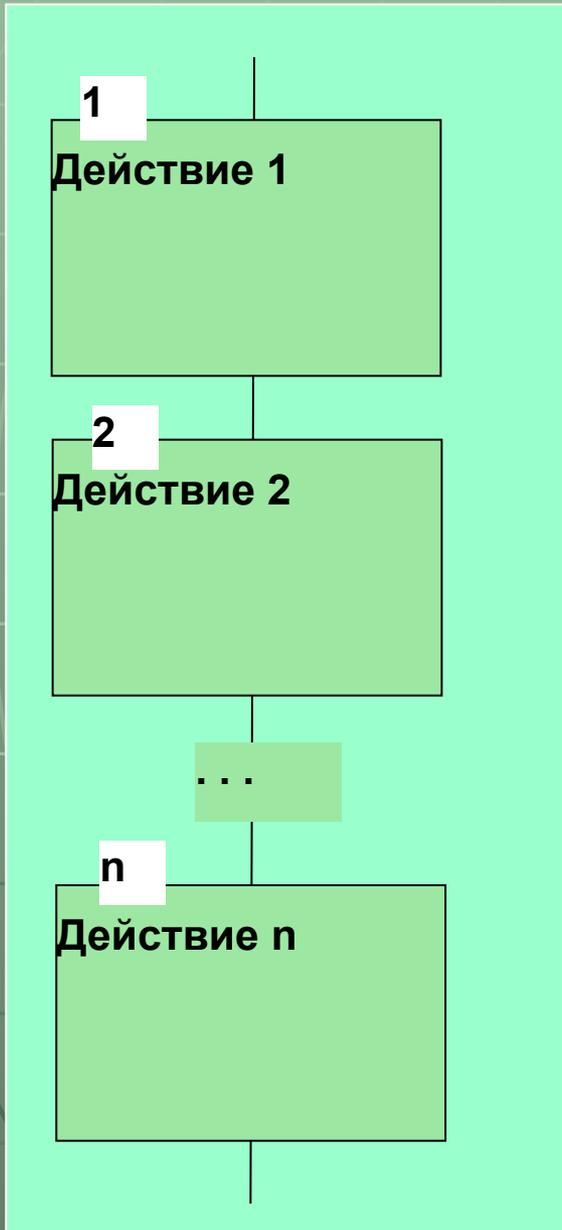
Блочные символы для схем алгоритмов

№	Наименование символа	Графическое обозначение символа	Правила применения символа и обозначений
1	Процесс		Используется для описания выполнения операции или группы операций
2	Решение		Используется для описания выбора направления выполнения алгоритма в зависимости от истинности или ложности условий.
3	Предопределенный процесс		Применяется для описания использования ранее созданных и отдельно написанных алгоритмов или программ

Блочные символы для схем алгоритмов

№	Наименование символа	Графическое обозначение символа	Правила применения символа и обозначений
4	Ввод-вывод		Используется для описания преобразования данных в форму, пригодную для обработки (ввод) или отображения ее результатов (вывод)
5	Модификация		Используется для описания и организации циклических конструкций. Определяет начало выполнения цикла
6	Пуск-останов		Используется для описания начала, конца, прерывания процесса обработки данных или выполнения программы

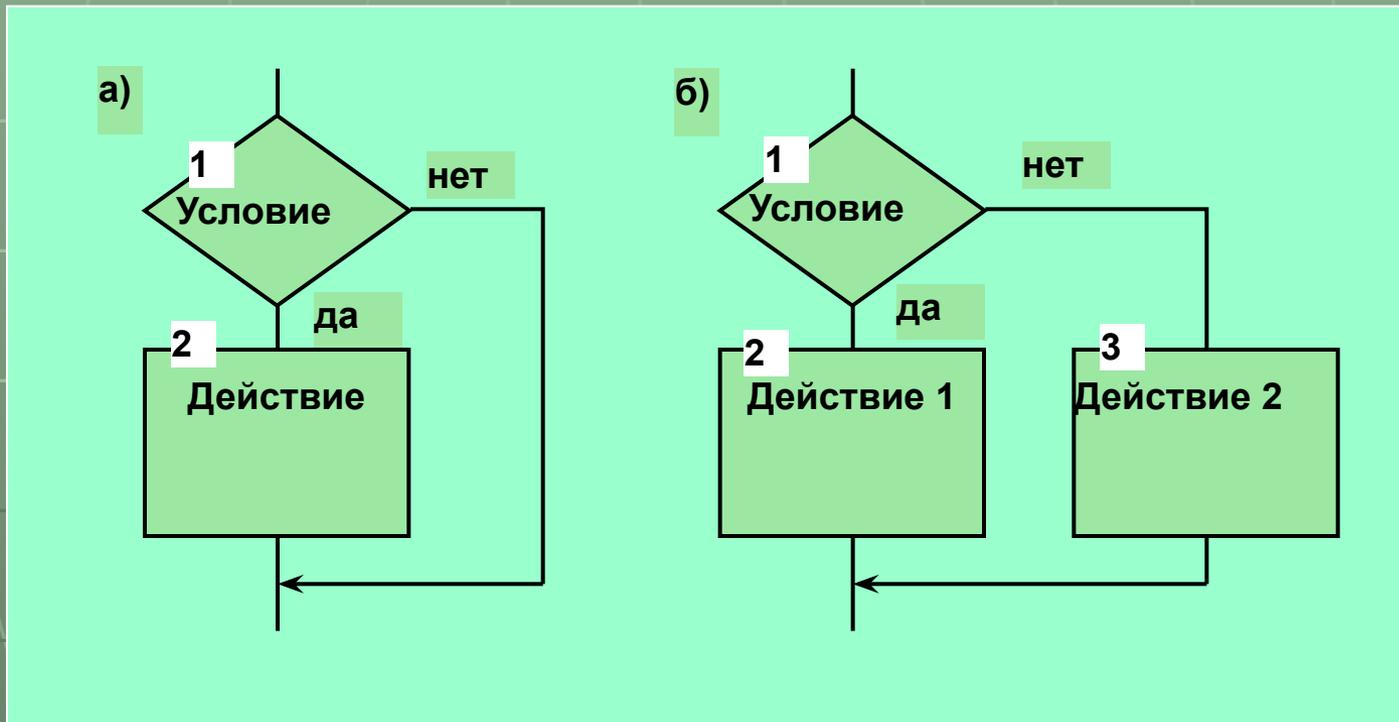
Базовые алгоритмические структуры



Базовая структура *следование*.
Образуется из последовательности однократно выполняемых действий, следующих одно за другим

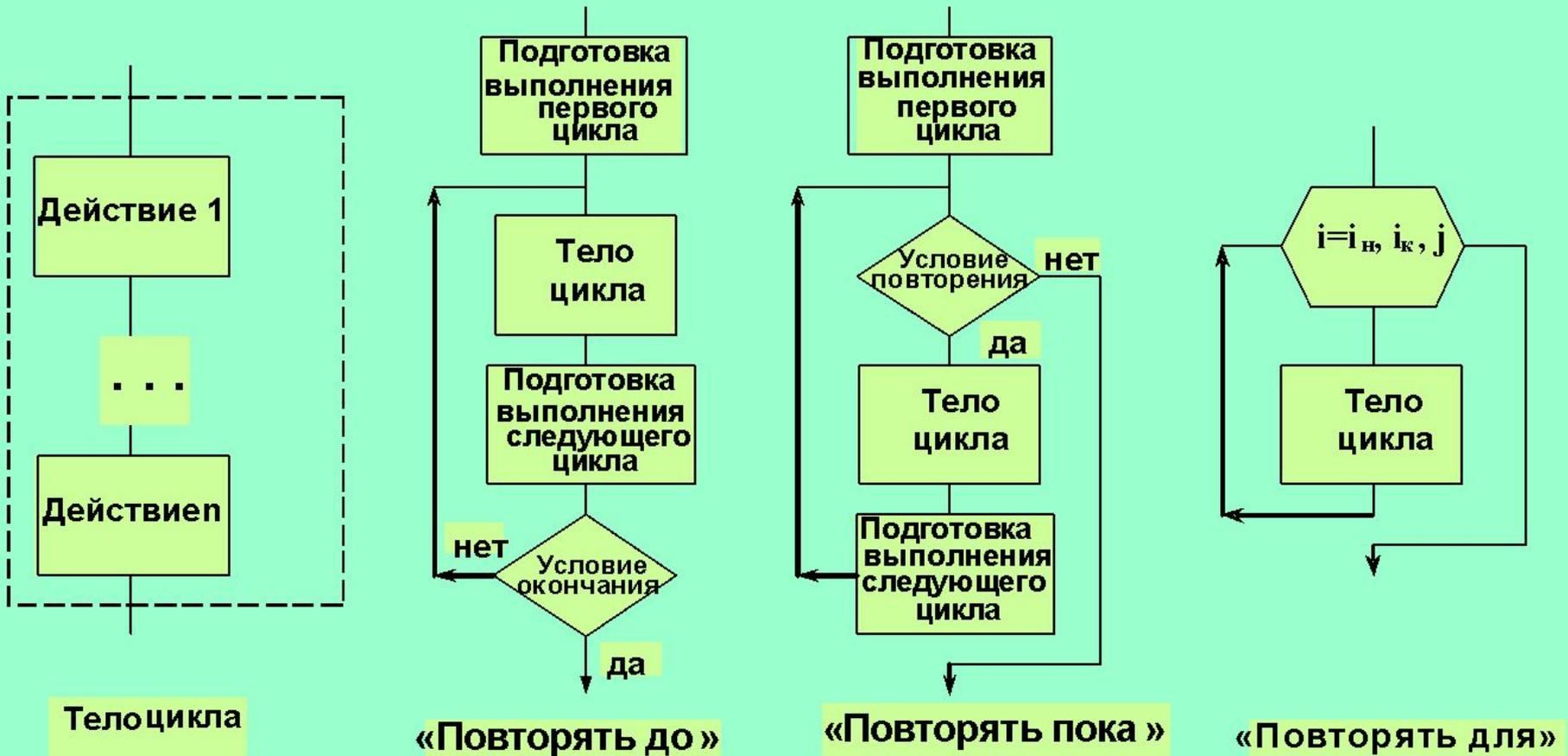
Базовые алгоритмические структуры

Базовая структура *ветвление*. Обеспечивает в зависимости от результата проверки условия (да или нет) выбор одного из альтернативных путей работы алгоритма. Каждый из путей ведет к общему выводу.

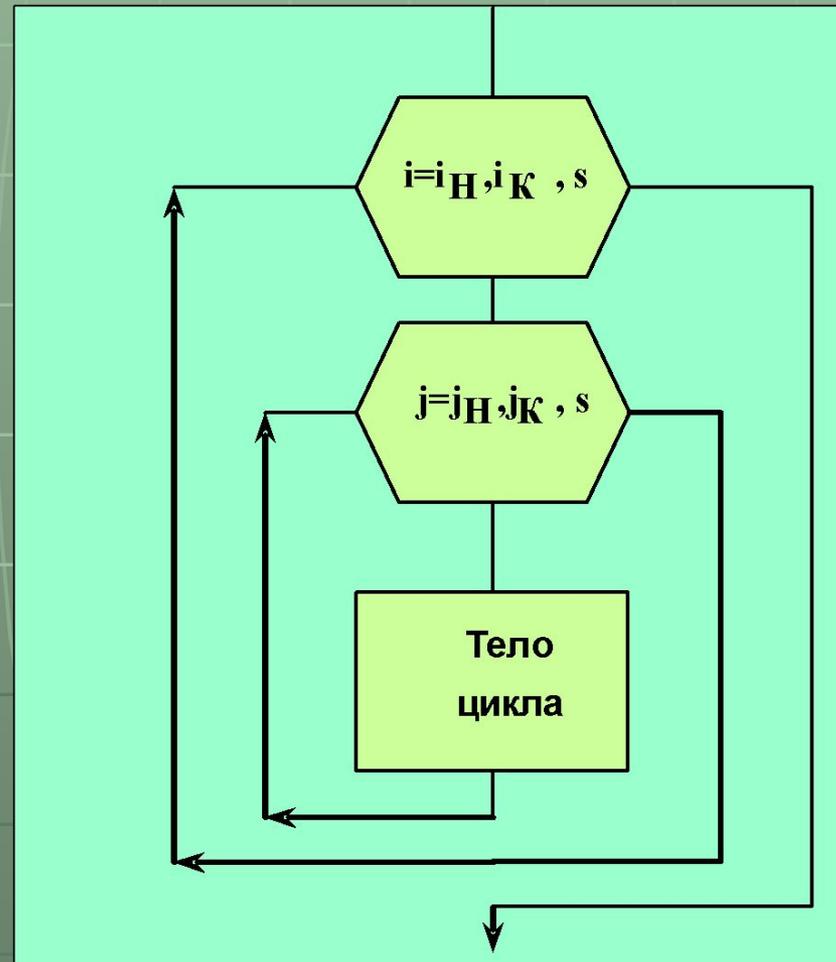


Базовые алгоритмические структуры

Базовая структура *цикл*. Обеспечивает многократное выполнение некоторой совокупности действий, которое называется ***телом цикла***.



Вычислительные процессы, содержащие два и более включенных друг в друга циклов, называются **сложными циклическими процессами**.



КОНТРОЛЬ

Тестовое задание №1

Одной из составных частей алгоритмического языка является ...

- a) фонетика
- b) алфавит
- c) грамматика
- d) стилистика

Тестовое задание № 2

Выбор одного из альтернативных путей обеспечивает базовая структура алгоритма...

- a) Следования.
- б) Повторения.
- с) Перехода.
- d) Ветвления.

Тестовое задание № 3

1.Выбор одного из альтернативных путей обеспечивает базовая структура алгоритма...

a) Следования.

b) Повторения.

c) Перехода.

d) Ветвления.

Тестовое задание № 4

Базовой алгоритмической структурой, обеспечивающей в зависимости от результата проверки условия выбор одного из путей работы алгоритма, является...

А) Ветвление;

Б) Следование;

В) Цикл;

Г) Размножение.

Тестовое задание № 5

В алгоритме для обозначения выполнения действий или последовательности действий, изменяющих значение, форму или представление данных применяется символ...

б) ромб

а) шестиугольник

в) прямоугольник

г) Овал

Спасибо за внимание!

Тема 3. Технология разработки программ

Лекция 2

Программирование на алгоритмическом языке бейсик

Алгоритмический язык образуют три его составные части: алфавит, синтаксис и семантика.

Алфавит – это фиксированный для данного языка набор основных символов, из которых состоит любой текст на этом языке.

Синтаксис – это набор правил, устанавливающих, какие комбинации символов являются осмысленными фразами и предложениями на этом языке.

Семантика – определяет смысловое содержание предложений языка. Семантика системой правил устанавливает, какие последовательности действий описываются теми или иными фразами языка.

Имена (идентификаторы) – применяются для обозначения различных объектов программы: переменных, констант, массивов, меток и др. Имена состояются из символов алфавита.

Метки – предназначены для отметки любого оператора программы.

Данные – это величины, обрабатываемые программой.

Константы – это данные, которые зафиксированы в тексте программы и не изменяются в процессе её выполнения.

Операции – определяют действия, которые выполняются над операндами (данными). Различают следующие операции:

арифметические операции $+$, $-$, $*$, $/$, $^$ и др;

логические операции AND, OR, NOT и др;

операции отношения $<$, $>$, $=$, \leq , $=>$, $<>$.

Выражения – предназначены для выполнения необходимых вычислений. Выражения состоят из констант, переменных, указателей функций, объединенных знаками операций.

Различают следующие типы выражений:

арифметические – $1+a/(b*\text{Sin}(x))$;

логические – $(f<d) \text{ OR } (a=b)$.

Функции и процедуры – это именованные подпрограммы, состоящие из логически законченных операторов (действий), которые можно вызывать для выполнения любое количество раз из любой части программы.

Примеры стандартных функций: $\text{Sin}(x)$, $\text{Abs}(y)$, $\text{log}(x)$, $\text{tan}(x)$.

Операторы – это наиболее содержательное понятие языка. Каждый оператор представляет собой законченную фразу языка и определяет некоторый завершённый этап обработки данных.

Неисполняемые операторы предназначены для описания данных и структуры программы.

Исполняемые операторы предназначены для выполнения различных действий.

Простой оператор – это оператор, не содержащий другие операторы.

Составной оператор – это любая совокупность простых операторов.

Основные составные операторы языка:

Оператор цикла *FOR ... NEXT* – управляющий оператор, повторяющий выполнение нескольких операторов указанное число раз.

Условный оператор *IF ... THEN ... ELSE* – управляющий оператор, осуществляющий условное ветвление по условию заданного логического выражения.

Комментарии – выполняют чисто информационную функцию и служат для описания назначения отдельных частей программы. Они не влияют на работу программы. Назначаются оператором REM или одиночным апострофом (').

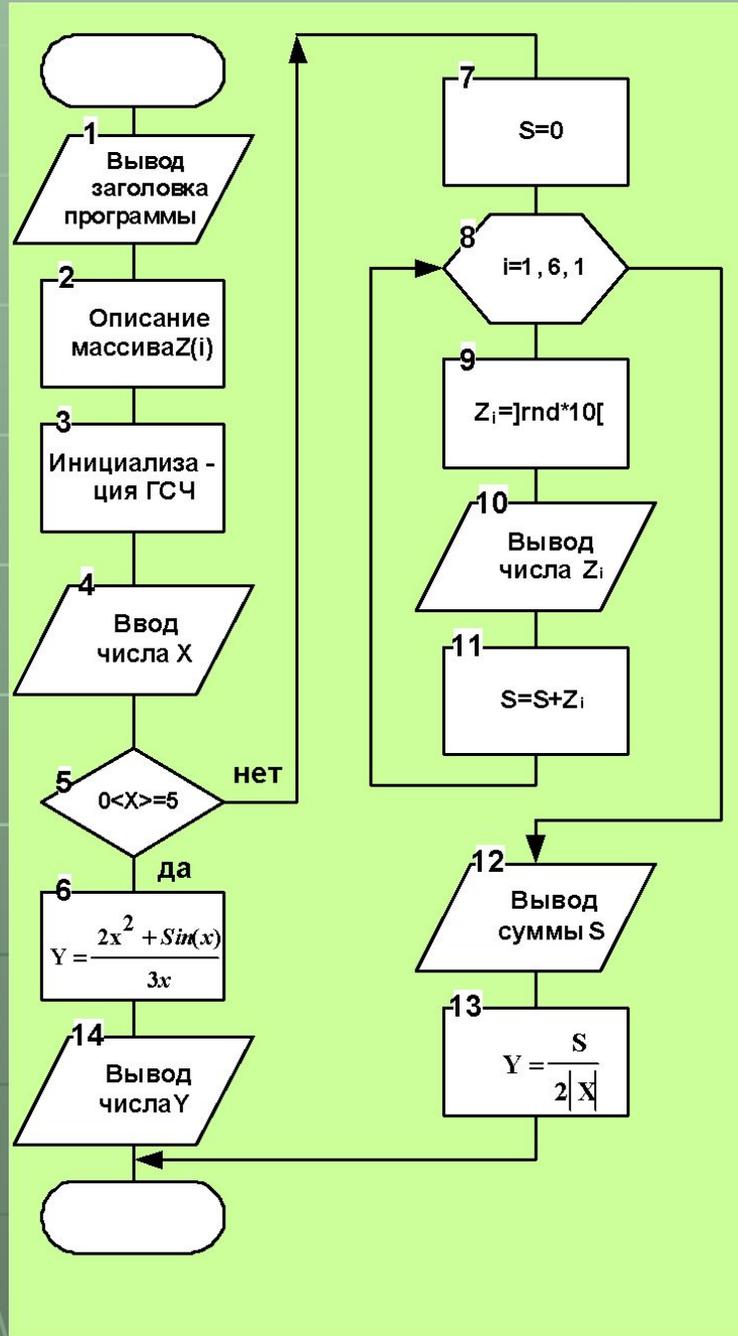
Разработка программы на алгоритмическом языке бейсик

Задача. Вычислить значение переменной Y по следующему выражению:

$$Y = \begin{cases} \frac{2 \cdot x^2 + \text{Sin}(x)}{3 \cdot x}, & \text{если } 0 < x \leq 5; \\ (\sum_{i=1}^6 Z_i) / (2 \cdot |x|), & \text{в противном случае.} \end{cases}$$

Переменные Z_i ($i=1, \dots, 6$) являются случайными числами с равномерным законом распределения в диапазоне от 0 до 10.

Схема разработанного алгоритма программы



Листинг программы:

```
PRINT "Пример программы "  
DIM Z(6)  
RANDOMIZE TIMER  
PRINT "Введите число X "  
INPUT  
IF (X > 0) AND (X <= 5)  
Y = (2 * X ^ 2 + SIN(X)) / (3 * X)  
ELSE  
S = 0  
  FOR i = 1 TO 6 STEP 1  
    Z(i) = INT(RND * 10)  
    PRINT "Z("; i; ")="; Z(i)  
    S = S + Z(i)  
  NEXT i  
PRINT "S="; S  
Y = S / (2 * ABS(X))  
END IF  
PRINT "Результат Y="; Y  
END
```

КОНТРОЛЬ

Тестовое задание № 1

**Именем функции языка
Бейсик не является ...**

a) BST

b) RND

c) EXP

d) ABS

Тестовое задание № 2

Имя функции языка Бейсик, которая возвращает возведенное в указанную степень число “e”...

- a) END
- b) EXP
- c) RND
- d) XOR

Тестовое задание № 3

Имя функции языка Бейсик, которая возвращает случайное число в диапазоне от 0 до 1 ...

a) INT

b) ABS

c) AND

d) RND

Тестовое задание № 4

Одной из основных конструкций алгоритмического языка является ...

- a) выражение
- b) переменная
- c) массив
- d) константа

Тестовое задание № 5

Имя функции языка Бейсик, которая возвращает квадратный корень числового выражения ...

a) ABS

b) SIN

c) SQRT

d) FIX

Спасибо за внимание!