

ЛЕКЦИЯ. АЛГОРИТМИЗАЦИЯ И ОСНОВЫ ПРОГРАММИРОВАНИЯ

План:

1. Система программирования.
2. Понятие алгоритма. Свойства и способы описания алгоритмов.
3. Графический способ описания (блок-схема).
4. Использование выражений и функций в программировании

1. Система программирования

Система программирования – это комплекс средств, предназначенных для создания и эксплуатации программы на каком-либо языке программирования и на ЭВМ определенного типа. В состав системы программирования входят: язык программирования; среда программирования; транслятор.

Язык программирования – это формализованный язык, предназначенный для описания алгоритмов решения задачи на компьютере.

При программировании для ПК используют 2 уровня языков: **низкий и высокий.**

К низкому уровню относится машинно-ориентированный язык Assembler.

Assembler – создает наиболее быстрые и эффективные программы, т.к. в нем существует прямая связь между кодом программы и конечным машинным кодом. Современные варианты языка позволяют использовать объектно-ориентированные конструкции.

Обзор языков программирования высокого уровня:

FORTRAN – один из первых языков высокого уровня, огромное количество программ, статистика, управление спутниками, параллельные суперкомпьютеры...

COBOL – экологические области и решение бизнес – задач. Много операторов, работает с большими массивами данных , используется в США.

ALGOL - сложная структура, призван заменить FORTRAN

PASCAL – создан Николаусом Виртом, основоположником многих идей

современного программирования. Можно применять при создании крупных проектов.

BASIC – первое место в мире по популярности. Учебный язык, прост в использовании

C – создан лабораторией ВЪ, как замена Ассемблера (язык машинных кодов) . похож на Паскаль, но добавлена работа с памятью, написано множество программ и операционных систем (UNIX)

C++ - объектно – ориентированное программирование , расширение языка C, создан в 1980 Бьярном Страуструпом. Требует высокого уровня профессиональной подготовки.

JAVA – создан компанией Sun в начале 90- х годов на основе языка C++. Призван упростить разработку приложений C++, по популярности занимает 2 место в мире.

Языки программирования баз данных:

SQL – структурированный язык запросов, первый для баз данных.

Сейчас появилась большая группа языков программирования для создания баз данных и управления ими : FOXPRO, CLIPPER, VISUAL DBASE, ORACLE....

Языки программирования для Интернет:

HTML – язык разметки гипертекста. Очень прост, на нем написаны все Web – страницы.

PERL – по мощности превосходит языки типа Си, создан в 80 – х годах Ларри Уоллом.

TCL/TR- не зависит от типа системы и позволяет создавать графический интерфейс, создан в конце 80- х Джоном Аустиравутом.

URML- создан в 1994 году для виртуальных трехмерных эффектов, вращений, света,...

Существуют и другие языки программирования: PROLOG, LISP, ADA, SMALLTALR....

В последнее время появились среды быстрого проектирования – визуального программирования, предназначенные для Windows:

- BASIC – Microsoft Visual Basic
- PASCAL – Borland Delphi
- C++ - Borland C++ Bulider
- IAVA – Symantec Café

Алгоритмическое (модульное) программирование :

Основная идея – программа разбивается на последовательность модулей, каждый из которых выполняет одно или несколько действий.

Требование к модулю – выполнение начинается с первой команды и заканчивается последней, без переходов.

Алгоритм на выбранном языке программирования записывается с помощью команд описания данных, вычисления значений и управление последовательностью выполнения программы.

Основные понятия такого программирования: переменные и константы, различные виды числовых данных, арифметические операции и выражения, логические выражения, строчные или тестовые переменные, блоки операторов – присваивание, условный оператор, операторы цикла, операторы ввода и вывода данных.

Структурное программирование.

Используется при создании средних по размеру приложений.

Идея заключается в том, что структура программы должна отражать структуру решаемой задачи, чтобы алгоритм решения был ясен из исходного текста. С этой целью введено понятие подпрограммы – набора операторов, выполняющих нужное действие и зависящих от других частей исходного кода. Программа разбивается на множество мелких подпрограмм (до 50 операторов). Комбинируя эти подпрограммы, формируют итоговый алгоритм уже не из простых операторов, а из законченных блоков, причем обращаться к блокам

можно по названиям. Получается, что подпрограммы – это новые операторы или операции языка. Подпрограммы можно использовать повторно.

Объектно – ориентированное программирование.

В середине 80 – х годов в программировании возникло новое направление, основанное на понятии объекта.

Объект – объединение свойств (структур данных, характерных для этого объекта), методов их обработки (подпрограмм изменения свойств) и событий, на которые данный объект может реагировать и которые приводят к изменению свойств. Из объектов, имеющих идентичную структуру получается класс. На основе классов можно создавать новые.

Среда программирования – комплекс программ, обрабатывающих исходную программу, запускающих ее и выдающих на экран результаты запуска программы.

В состав среды программирования входят: интерпретатор; программная оболочка; экранный редактор; отладчик программы; вспомогательный модуль помощи.

Процесс преобразования текста программы в машинные коды называется **трансляцией**.

Программа, обеспечивающая процесс трансляции называется **транслятором**.

Существует 2 вида трансляторов:

1. Компилятор - осуществляет перевод всей исходной программы на машинный язык, проверяет ее правильность и в конце работы выдает список ошибок, если они обнаружены. Недостаток компилятора: нет возможностей внесения в программу каких-либо изменений.

2. Интерпретатор – обеспечивает поочередной перевод каждого оператора исходной программы на язык машины, если обнаруживает ошибку, то сообщает о ней и прекращает выполнение программы.

Величина — это отдельный информационный объект, который имеет имя, значение и тип.

Величины бывают *постоянными и переменными*.

Постоянная величина (константа) не изменяет своего значения в ходе выполнения алгоритма. Константа может обозначаться собственным значением (числа 10, 3.5) или символическим именем (число π).

Переменная величина может изменять значение в ходе выполнения алгоритма. Переменная всегда обозначается символическим именем (X, A, R5 и т.п.), начинается с буквы и не содержит пробелов.

Тип величины определяет множество значений, которые может принимать величина, и множество действий, которые можно выполнять с этой величиной. Основные типы величин: *целый, вещественный, символьный, логический*.

Решение любой задачи имеет три части:

- 1) ввод данных;
- 2) обработка данных;
- 3) вывод результата.

Команда ввода — команда, по которой значения переменных задаются через устройства ввода (например, клавиатуру).

Команда вывода — команда, по которой значение величины отражается на устройстве вывода компьютера (например, на экран дисплея).

Обработка данных может включать команды присваивания, ветвления и цикла.

2. Понятие алгоритма. Свойства и способы описания алгоритмов

Алгоритм — это конечная последовательность точно определённых действий, приводящих к решению поставленной задачи.

При составлении алгоритмов следует учитывать ряд требований, выполнение которых приводит к формированию необходимых свойств:

- *определённость* - алгоритм должен быть однозначным, исключающим произвольность толкования любого из предписаний и заданного порядка исполнения;
- любой алгоритм должен иметь только одно начало (один вход) и одно окончание (один выход);
- *результативность* - реализация процесса, предусмотренного алгоритмом, должна выдать результаты или сообщение о невозможности решения задачи;
- *массовость* - способность алгоритма обеспечить решение однотипных задач с различными исходными данными;
- *дискретность* - расчленение процесса, предусмотренного алгоритмом, на отдельные этапы, элементарные операции.

Для строгого задания различных структур данных и алгоритмов, их обработки, требуется иметь такую систему формальных обозначений и правил, чтобы смысл всякого используемого предписания трактовался точно и однозначно. Для выполнения этого условия или требования существуют следующие ***способы описания алгоритмов***:

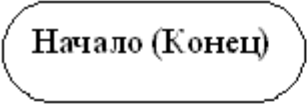
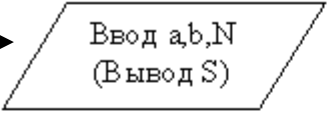
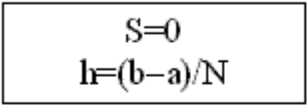
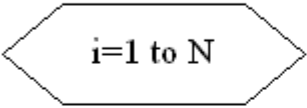
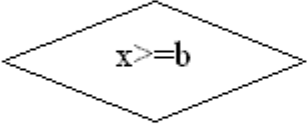
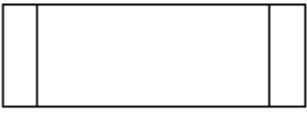
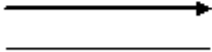


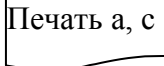
- словесное описание (запись на естественном языке);
- графическое описание (блок-схемы);
- программное описание (тексты программ на алгоритмическом языке).

3. Графический способ описания (блок-схема)

Одним из инструментов, часто используемым для записи алгоритмов, являются блок-схемы.

Блок-схема представляет собой набор некоторых геометрических фигур или блоков, в которых можно записать любой текст, формулу или какую-либо информацию о том, что надлежит сделать на данном этапе выполнения алгоритма.

Для составления алгоритма в виде блок-схемы применяются следующие основные графические изображения.

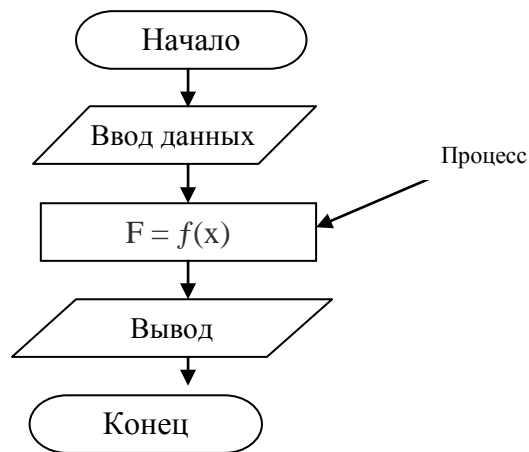
Скругленный прямоугольник	→ 	– Начало или конец алгоритма.
Параллелограмм	→ 	– Ввод исходных данных или вывод результатов.
Прямоугольник	→ 	– Выполнение операции или группы операций (Процесс)
Шестиугольник	→ 	– Безусловный цикл, то есть с определённым (конечным) числом шагов. (Подготовка или модификация)
Ромб	→ 	– Выбор направления выполнения алгоритма в зависимости от некоторых условий. (Решение)
		– Модуль подпрограммы – процедуры или подпрограммы – функции. (Типовой процесс)
		– Линия потока, соединяющая фигуры блок-схемы. Направление линии указывать при её ходе слева направо. Изменение направления линии под прямым углом.
		– Разрыв линии потока
		– Ссылка на другую страницу.
		– Документ (вывод результатов на печать)

Базовые конструкции алгоритмов

При разработке и составлении блок-схемы того или иного алгоритма применяются следующие базовые конструкции: линейная, циклическая и разветвляющаяся.

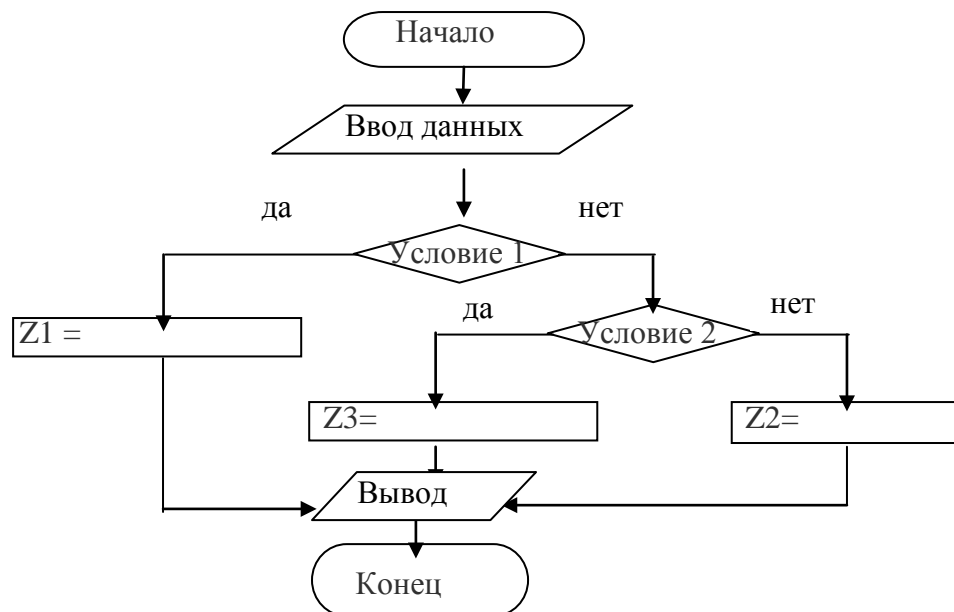
Линейная конструкция – это последовательное выполнение операций без повторов и разветвлений.

Блок-схема линейного алгоритма:



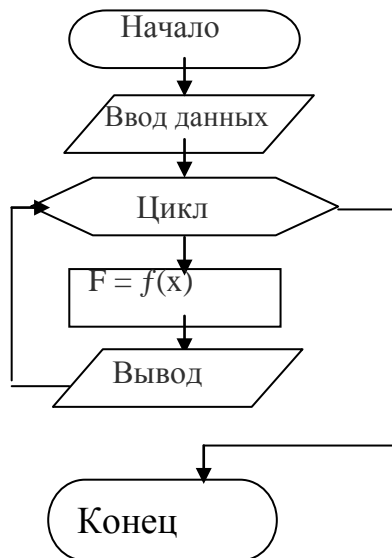
Разветвляющаяся конструкция – это выполнение операций по одному из нескольких направлений в зависимости от заданных условий.

Блок-схема разветвляющего алгоритма:



В циклической конструкции циклы используются для организации повторного выполнения какой-либо операции (инструкции) или блока операций (инструкций).

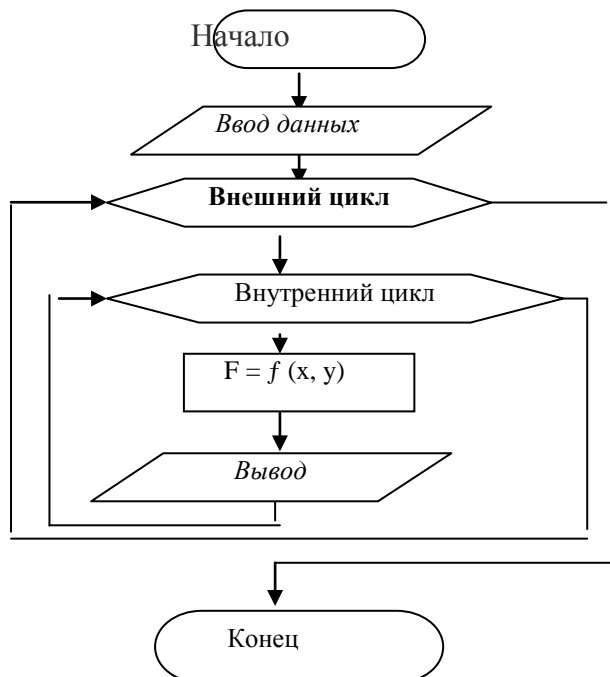
Блок-схема циклического алгоритма:



Простые циклические конструкции могут вкладываться в другую простую циклическую конструкцию, образуя тем самым вложенный (сложный) цикл (т.е. цикл в цикле). При этом необходимо выполнять следующие правила:

- имена параметров всех простых циклов не должны повторяться;
- нельзя войти во внутренний цикл, минуя внешний;
- простые циклы в сложном цикле не должны пересекаться, то есть внешний цикл должен заканчиваться после внутреннего и инструкции тела внешнего цикла не должны быть в теле внутреннего цикла.

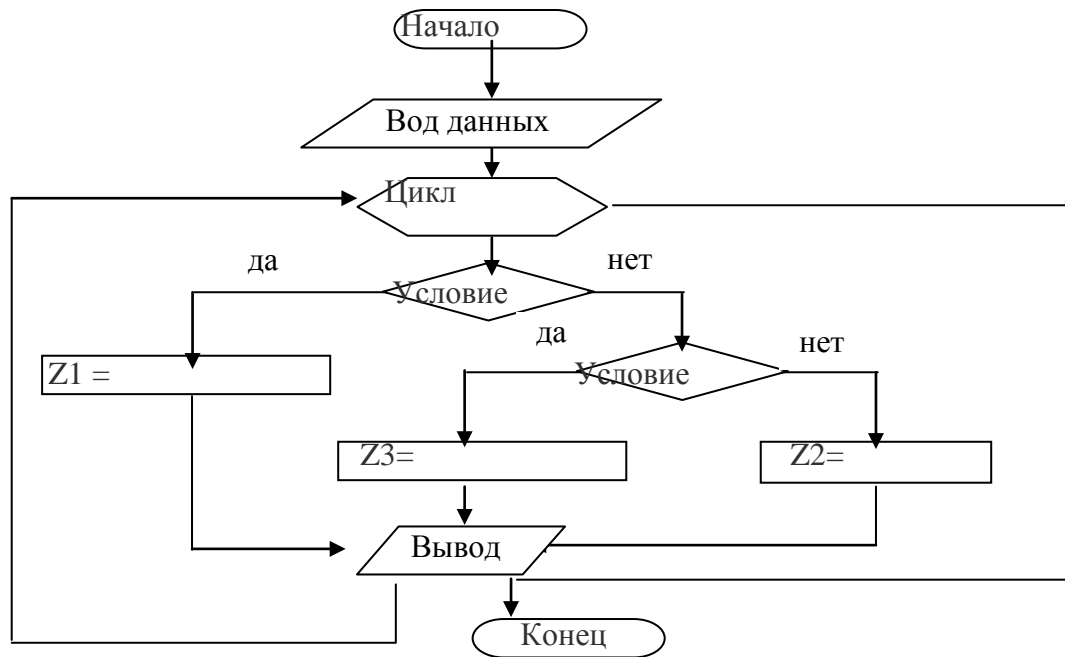
Блок-схема алгоритма с вложенными циклами:



Также существуют *комбинированные конструкции*, которые включают в

себя разветвляющий и циклический алгоритм.

Блок-схема комбинированного алгоритма:



Базовые конструкции алгоритмов в чистом виде на практике не применяются, а используются в сочетании между собой.

4. Использование выражений и функций в программировании

Среда программирования позволяет оперировать сложными арифметическими выражениями, содержащими тригонометрические, логарифмические, степенные, показательные и др. функции.

Выражение представляет собой запись, указывающую, какие операции следует произвести над данными, чтобы получить требуемое значение.

Арифметическое выражение – это символьная запись составленная из чисел, имен переменных и элементов массивов, встроенных функций, знаков арифметических операций, круглых скобок.

Основные стандартные функции и выражения:

МАТЕМАТИЧЕСКИЙ СИМВОЛ (ВЫРАЖЕНИЕ, ФУНКЦИЯ)	ЗАПИСЬ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ
+ - x :	+ - * /

-0,07	-0.07
X+Y	ABS(X+Y)
SINX	SIN(X)
COSX	COS(X)
TGX	TAN(X) или SIN(X)/ COS(X)
CTGX	1/TAN(X) или COS(X)/SIN(X)
ARCTG X	ATN(X)
LN X	LOG(X)
LG X	LOG(X)/LOG(10)
LOG _y X	LOG(X)/LOG(Y)
X ^Y	X^Y
\sqrt{Y}	Y^(1/2) или SQR(Y) или Y^0.5
$\sqrt[n]{Y}$	Y^(1/n)
$\sqrt[3]{Y^5}$	Y^(5/3)
E ^X	EXP(X)
Округление М до целого	INT(M)

При записи арифметических выражений необходимо придерживаться следующих правил и ограничений:

I. Все символы выражения записываются в одну строку. Многоэтажные выражения, верхние и нижние индексы запрещены.

II. Два знака арифметических операций не должны располагаться рядом. Знак умножения опускать нельзя.

III. Операции в арифметическом выражении выполняются в порядке старшинства, т.е.:

- 1) операции внутри скобок;
- 2) вычисления встроенных функций;
- 3) возведение в степень;
- 4) операции умножения и деления;
- 5) операции целочисленного деления;
- 6) операции деления по модулю;
- 7) операции сложения и вычитания.

Операции равного старшинства выполняются по порядку слева направо.

Исключение: $A^B^C = A^{(B^C)}$.

IV. Тип арифметического выражения определяется типом его результата:

а) операция деления (« \ ») с целыми величинами дает вещественный результат;

б) выражение может содержать и целые и вещественные величины. Результат такого выражения - вещественная величина;

в) при наличии в арифметическом выражении величин двойной точности результатом будет величина той же точности.

Приведем примеры записи арифметических выражений на языке программирования:

ЗАПИСЬ арифметических ВЫРАЖЕНИЙ	
$3\sin 5x$	$3*\sin(5*x)$
$3\operatorname{tg}^2 5x$	$3*\tan(5*x)^2$
$2\operatorname{Log}_5^3 22$	$2*(\operatorname{Log}(22)/\operatorname{Log}(5))^3$
$3\arctg X + e^{5.11} - 2e$	$3*\operatorname{atn}(x) + \exp(5.11) - 2*\exp(1)$
$-2\frac{3}{4} \cdot \operatorname{ctg} 5x$	$-(2+3/4)*(1/\tan(5*x))$
$\sqrt[5]{ 2x + e^3 }$	$\operatorname{Abs}(2*X + \operatorname{EXP}(3))^{(1/5)}$

$$\cos^2 RT \Rightarrow \operatorname{COS}(R*T)^2$$

$$\operatorname{ctg}^3 MC^2 \Rightarrow (\operatorname{COS}(M*C^2)/\operatorname{SIN}(M*C^2))^3$$

$$\log_7^2(5x+s)^3 \Rightarrow (\operatorname{LOG}((5*X+S)^3)/\operatorname{LOG}(7))^2$$

$$\frac{7\cos^2 43,7 + \ln 56}{3 \sqrt{|8,33^2 - 62^7|}} \Rightarrow (7*\operatorname{COS}(43.7)^2 + \operatorname{LOG}(56))/(\operatorname{ABS}(8.33^2 - 62^7))^{(1/3)}$$

$$6\frac{3}{4} \cdot \operatorname{ctg}^3 67,88^2 + \operatorname{Log}_7 41 + e^{2\sin 5} \Rightarrow$$

$$\Rightarrow (6+3/4)*(\operatorname{COS}(67.88^2)/\operatorname{SIN}(67.88^2))^3 + \operatorname{LOG}(41)/\operatorname{LOG}(7) + \operatorname{EXP}(2*\operatorname{SIN}(5))$$

Математическое выражение

$$x^2 - 7x + 6$$

$$\frac{|x| - |y|}{1 + |xy|}$$

$$\ln \left| (y - \sqrt{|x|}) \left[x - \frac{y}{z + x^2/4} \right] \right|$$

$$\ln \left| (y - \sqrt{|x|}) \left[x - \frac{y}{z + x^2/4} \right] \right|$$

Выражение на VBA

$$x^2 - 7*x + 6$$

$$(\operatorname{Abs}(x) - \operatorname{Abs}(y)) / (1 + \operatorname{Abs}(x*y))$$

$$\operatorname{Log}(\operatorname{Abs}((y - \operatorname{Sqr}(\operatorname{Abs}(x))) * (x - y / (z + (x)^2/4))))$$