

Министерство общего и профессионального образования
Государственное автономное образовательное учреждение среднего
профессионального образования Свердловской области
«Белоярский многопрофильный техникум»

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

п. Белоярский
2014

РАЗДЕЛ 1. ОСНОВЫ АЛГОРИТМИЗАЦИИ

Этапы решения задачи на ЭВМ

Работа по решению любой задачи с использованием компьютера делится на следующие этапы:

1. Постановка задачи (выделение исходных данных, результата и промежуточных данных)
2. Формализация задачи (математическая постановка)
3. Построение алгоритма
4. Составление программы на языке программирования
5. Отладка и тестирование программы
6. Проведение расчетов и анализ полученных результатов

Пример 1. Найти периметр прямоугольника.

Исходные данные – те данные, которые известны или которые пользователь может ввести с клавиатуры по ходу выполнения программы.

Для исходных данных необходимо определить переменные.

Исходные данные: а, в – стороны прямоугольника

Результат – переменная, в которой необходимо получить результат или сообщение, которое необходимо получить в результате.

Результат: Р – периметр прямоугольника.

Математическая модель – выражает результат через исходные данные с помощью математических формул.

Математическая модель: $P=2 \cdot (a+v)$

Промежуточные переменные – переменные, которые используются в программе, но не являются ни исходными данными, ни результатом.

Промежуточные переменные: в данной задаче нет



ЗАДАНИЕ 1

Выделите исходные данные, результат, промежуточные переменные (если есть) и постройте математическую модель для следующих задач:

1. Найти площадь круга.
2. Найти площадь кольца.
3. Даны 2 точки. Найти расстояние между ними.
4. Даны 3 точки. Найти площадь, образованного этими точками треугольника.
5. Определить сколько времени в минутах затратит школьник на дорогу от школы до стадиона, если известно расстояние и средняя скорость движения школьника.

Алгоритмы

Алгоритм – последовательность команд или действий, выполнение которых приводит к решению поставленной задачи.

Примером алгоритма может служить открывание двери:

1. Достать ключ
2. Вставить ключ в замочную скважину
3. Повернуть ключ
4. Вынуть ключ



УПРАЖНЕНИЕ 1

Посмотрите внимательно. Здесь иллюстрации хорошо известной вам сказки. Попробуйте восстановить точную последовательность (порядок) действий персонажей этой сказки и впишите буквы в соответствующие ячейки.



Г



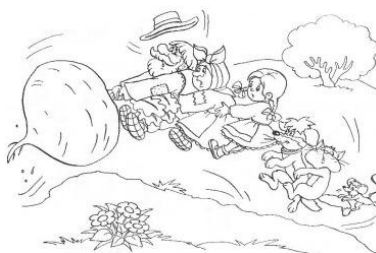
М



И



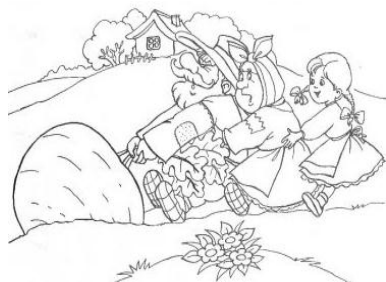
Л



Т



А



О



Р

1	2	3	4	5	6	7	8

Порядок действий считается алгоритмом в том случае, если он обладает определенными свойствами.

Свойства алгоритма

- *Дискретность.* Алгоритм должен представлять процесс решения задачи как последовательность выполнения простых действий (шагов, этапов). При этом для выполнения каждого действия алгоритма требуется время.
- *Детерминированность (Однозначность).* Каждое действие (шаг, этап) должно быть четким, однозначным, исключающим произвольное толкование и не оставляющим места для двусмысленности. Выполнение алгоритма носит, по сути, механический характер и не требует никаких дополнительных указаний.
- *Результативность.* Алгоритм должен приводить к решению задачи или сообщению, что задача решений не имеет за конечное число шагов.
- *Конечность.* Каждое отдельное действие, как и весь алгоритм должны иметь возможность реального исполнения. Поэтому алгоритм имеет предел, т. е. конечен.
- *Массовость.* Алгоритм разрабатывается в общем виде так, чтобы его можно было применять для класса задач, различающихся только исходными данными. При этом исходные данные выбираются из некоторой области, которая называется областью применимости алгоритма.



УПРАЖНЕНИЕ 2

Сравните команды алгоритмов	Свойство, которое нарушено
<i>Положить несколько ложек сахара</i>	
<i>Разбить 3 яйца, вылить тесто в форму, 1 стакан сахара, добавить 2 стакана муки, замесить тесто, выпекать в духовке 20 мин., 2 стакана сметаны</i>	

Способы описания алгоритмов

Существуют разные способы записи алгоритмов – *словесный (словесно-формульный), графический, программный.*

а) *Словесный (словесно-формульный) способ.* Например, к приборам бытовой техники, как правило, прилагается инструкция по эксплуатации, т.е. словесное описание алгоритма, в соответствии с которым данный прибор должен использоваться.

б) *Графический способ* описания алгоритма иначе называют блок - схемой. В блок-схемах используются геометрические фигуры, каждая из которых изображает какую-либо операцию или действие, а также этап процесса решения задачи. Каждая фигура называется блоком. Порядок выполнения этапов показывается стрелками, соединяющими блоки.

Правила построения блок-схем

1. Каждая блок-схема должна иметь начало и конец.
2. Блоки располагаются сверху вниз или слева направо.
3. Между собой блоки соединяются линиями потока, указывающими последовательность выполнения операторов.
4. Если блок-схема не умещается на листе, то её переносят на следующий лист с помощью соединителя.

В таблице 1 приведены наиболее часто употребляемые блоки.

Таблица 1

Название	Символ (рисунок)	Выполняемая функция (пояснение)
1. Блок вычислений		Выполняет вычислительное действие или группу действий
2. Логический блок		Выбор направления выполнения алгоритма в зависимости от условия
3. Блоки ввода/вывода		Ввод или вывод данных вне зависимости от физического носителя
		Вывод данных на печатающее устройство
4. Начало/конец (вход/выход)		Начало или конец программы, вход или выход в подпрограмму
5. Блок модификации		Выполнение действий, изменяющих пункты алгоритма
6. Соединитель		Указание связи между прерванными линиями в пределах одной страницы

в) *Программный способ (алгоритмический язык)*. Алгоритм - это задание для исполнителя. Исполнитель выполняет алгоритм, т. е. делает то, что написано в алгоритме. Если исполнитель точно выполнит то, что написано в алгоритме, то он получит результат.

Человек, автоматическое устройство, компьютер - это разные исполнители алгоритмов. Для того чтобы компьютер мог выполнить алгоритм, его надо написать на понятном компьютеру языке. Компьютер понимает машинный язык. Например, равенство $x = y$ на машинном языке имеет вид: 111101110011110111110101.

Понятно, что человеку трудно писать и читать алгоритмы на машинном языке. Человек легко может писать и читать на естественном языке. Но нельзя научить компьютер понимать естественный язык потому, что в естественном языке много слов и нет строгих правил записи предложений.

Для того чтобы человек и компьютер понимали друг друга, разработаны специальные

языки для записей алгоритмов - алгоритмические языки. Самые известные алгоритмические языки - это Бейсик (Basic), Паскаль (Pascal), Фортран (Fortran).

Алгоритмический язык отличается от естественного языка тем, что в нем мало основных слов (обычно 30-40) и очень строгие правила составления предложений. Основные слова алгоритмического языка называют служебными словами. В алгоритмических языках используют слова английского алфавита.

Алгоритм, который записан на алгоритмическом языке, - это программа для компьютера. Каждое предложение в программе - это оператор.

Например,

```
PROGRAM RR;
VAR A,B,C, max: INTEGER;
BEGIN
WRITE (' ВВЕДИТЕ A, B, C');
READLN (A,B,C);
IF A>B THEN max:=A
ELSE max:=B;
IF C>max THEN max:=C;
WRITELN (max);
END.
```

Типы алгоритмов

Алгоритмы бывают *линейные, разветвляющиеся и циклические*.

Линейный алгоритм - это алгоритм, в котором действия выполняются только один раз и строго в том порядке, в котором они записаны.



Выполните Лабораторную работу №1

Разветвляющийся алгоритм - это алгоритм, в котором то или иное действие выполняется после анализа условия. Процесс анализа условия и выбора одной из ветвей на блок-схеме показывают с помощью логического блока.

Процесс анализа условия и выбора одной из ветвей на блок-схеме показывают с помощью логического блока. Логический блок имеет один вход и два выхода (ветвь «да» и ветвь «нет»).

В блок-схемах разветвляющихся алгоритмов всегда есть логический блок.



Выполните Лабораторную работу №2

Циклический алгоритм (цикл) - это алгоритм, в котором группа операторов выполняется несколько раз подряд. Блок-схема цикла обязательно содержит логический блок.

Выполняется циклический алгоритм так: сначала проверяется условие, если условие верно

(истина), то выполняется тело цикла (действия или группа операторов) и, далее, изменяются значения параметра цикла и снова проверяется условие и т. д. На каком-то шаге условие не выполнится (ложь) и тогда происходит выход из цикла и продолжается выполнение программы.

Существует три вида циклов:

- Цикл с параметром (арифметический)
- Цикл с предусловием
- Цикл с постусловием



Выполните Лабораторную работу №3

Переменные и константы

Идентификатор – это имя, выбираемое программистом, для программы, констант, переменных, функций и т.д.

Имя состоит из латинских букв (прописные и строчные буквы не различаются), цифр, знака подчеркивания, но начинается с буквы или знака подчеркивания. Длина имени – не более 255 символов, различаются имена по первым 63 символам. Нельзя использовать зарезервированные слова в качестве идентификатора.

Константы – это неизменяемая величина определенного типа.

Виды констант:

- числовые (числа целые и вещественные)
- логические (высказывание принимает значение истина или ложь)
- символьные ('a')
- строковые ('ворона')

Переменная – это элемент программы, предназначенный для хранения, коррекции и передачи данных внутри программы. Переменная имеет имя и тип. В памяти компьютера переменная – это ячейка, имеющая адрес и значение. Она может менять свои значения при выполнении программы.

Рассмотрим подробнее правила построения имени:

- 1) имя должно начинаться с буквы латинского алфавита. Пример: leto, a_1
- 2) не различают строчные и заглавные буквы - они эквивалентны
Пример: Kat или kAT или kaT – одинаковые имена
- 3) со второй позиции можно использовать цифры
Пример: A2, f21, 4d
- 4) пробел является разделителем и не может стоять внутри идентификатора
Пример: ~~Read Text~~, Read_Text, ReadText
- 5) зарезервированные слова не используются в качестве имени

Пример: var, end – нельзя, rav, son - можно

б) длина имени может быть до 255 символов, но сравнения между собой производятся по первым 63 символам

Пример: $\underbrace{a111\dots1}_{63}33$ и $\underbrace{a111\dots1}_{63}44$ – имена совпадают

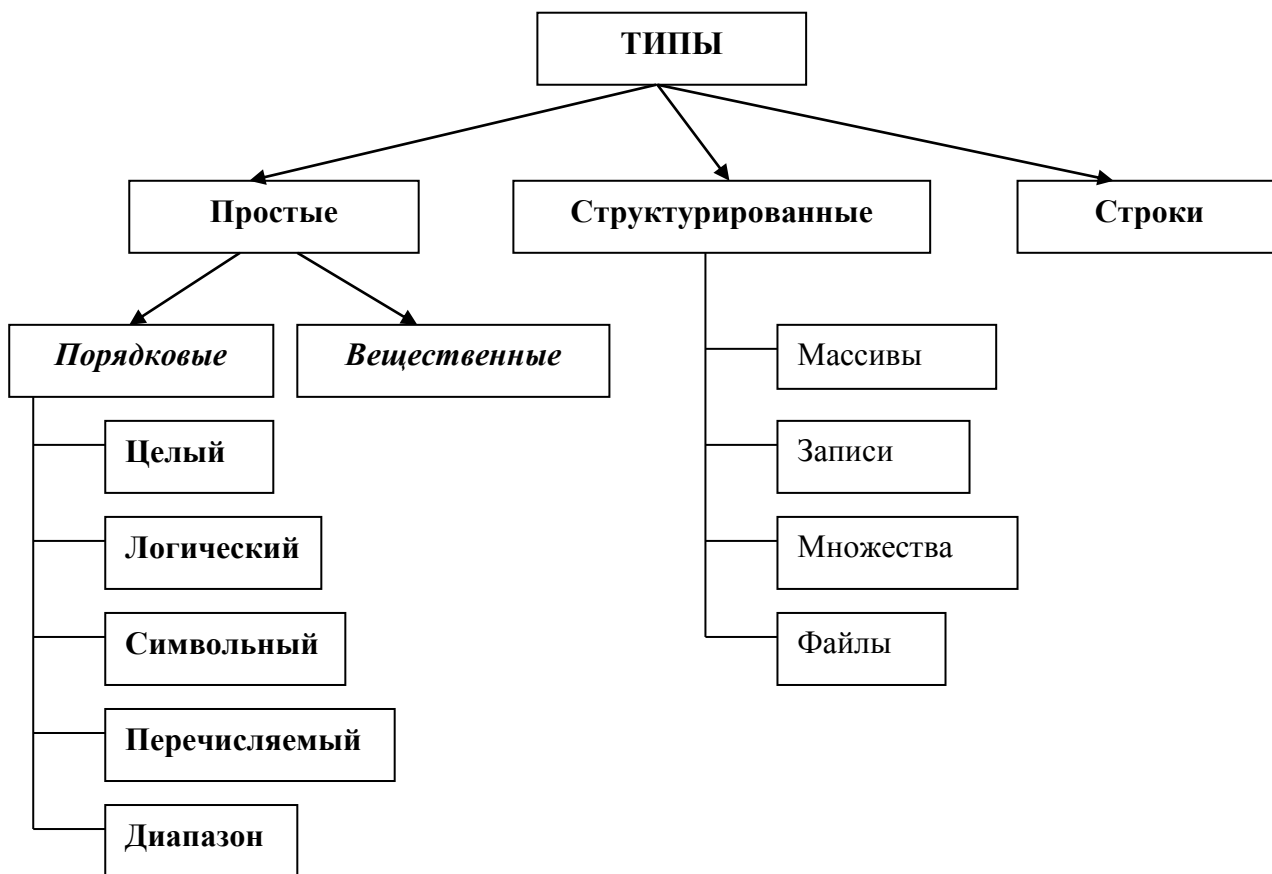
7) применение других символов не допускается.

Типы данных

Любые *данные*, т.е. константы, переменные, значения функций или выражений *характеризуются своими типами*.

Тип определяет:

- Множество допустимых значений, которые могут принимать данные;
- Множество допустимых операций, применяемых к данным;
- Формат внутреннего представления данных в памяти РС.



Целый тип:

Название	Длина, байт	Диапазон значений
Byte	1	0 .. 255
Shortint	1	- 128 .. + 127
Word	2	0 .. 65535
Integer	2	- 32768 .. + 32767
Longint	4	- 2 ³¹ .. +2 ³¹ - 1

Допустимые операции: сложение, вычитание, умножение, DIV (целочисленное деление), MOD (остаток от целочисленного деления).

Логический тип – Boolean (1 байт)

Символьный тип – Char (1 байт)

Вещественный тип:

Вещественные числа – это числа, в записи которых присутствует десятичная точка.

Вещественные числа в компьютере представлены не точно, а приближенно, с некоторой конечной точностью, в зависимости внутреннего формата числа, т.е. его типа.

Название	Длина, байт	Количество значащих цифр	Диапазон значений
SINGLE	4	7 .. 8	$10^{-45} .. 10^{+38}$
REAL	6	11 .. 12	$10^{-39} .. 10^{+38}$
DOUBLE	8	15 .. 16	$10^{-324} .. 10^{+308}$
EXTENDED	10	19 .. 20	$10^{-4951} .. 10^{+4932}$
COMP	8	19 .. 20	$-9,2 * 10^{18} .. +9,2 * 10^{18}$



ЗАДАНИЕ 2

Укажите, какие из записей *являются* идентификаторами:

- | | | |
|-------------------|------------|---------------|
| 1) N ₂ | 5) K1L2 | 9) doc! |
| 2) myname | 6) 4CD | 10) X |
| 3) U2 | 7) max_min | 11) a-b |
| 4) Z' | 8) matr 2 | 12) программа |



ЗАДАНИЕ 3

Определите типы следующих величин:

- | | |
|------------------------|-------------------------------|
| А) вес человека | Г) площадь фигуры |
| Б) марка автомобиля | Д) название месяца года |
| В) год вашего рождения | Е) количество мест в самолете |



ЗАДАНИЕ 4

Составьте блок-схему, определите её тип и выберите подходящие типы данных для исходных данных и результата для следующих алгоритмов:

а) Начало
 Запросить число X
 Если число $X > 7$ то
 $Y = X - 2$
 иначе $Y = X * 2$
 Сообщить результат
 Конец

Чему будет равен Y, если $X = 2$ и $X = 9$?

Ответ: _____

в) Начало
 Запросить число X
 Если число $X > 0$ то
 $Y = 125 : X$
 иначе если число $X = 0$ то
 вывести «На 0 делить нельзя»
 Сообщить результат
 Конец



ЗАДАНИЕ 5

Составьте блок-схему, определите её тип и выберите подходящие типы данных для исходных данных и результата для следующих задач:

1. Найти площадь круга.
2. В кинотеатре 30 рядов кресел. В первом ряду 20 кресел, в каждом последующем на 2 кресла больше, чем в предыдущем. Сколько мест в зрительном зале?
3. Не пользуясь операцией возведения в степень, возвести число A в степень N.
4. Дано число X. Увеличить его на 5, если оно положительное.
5. Ученик в первый день занятий выучил 5 английских слов. В каждый последующий день он выучивал на 2 слова больше, чем в предыдущий. Через сколько дней ученик будет знать 250 английских слов?
6. Дано натуральное N. Выяснить, сколько цифр оно содержит.
7. Дан периметр прямоугольника P и длина стороны a. Найти вторую сторону.
8. В первый час работы рабочий изготавливает 25 деталей, за каждый последующий час на 3 детали больше, чем за предыдущий. Подсчитать, сколько рабочий изготовит деталей за 8 часов работы.
9. Даны два числа a и b. Наименьшее из них возвести в квадрат, наибольшее – в куб и найти среднее арифметическое полученных значений.
10. Дана площадь S квадрата. Вычислить длину стороны a.

Решите две задачи, согласно варианту.

Вариант	0	1	2	3	4	5	6	7	8	9
1 задача	1	2	5	6	7	8	9	10	3	4
2 задача	3	4	10	1	5	9	2	6	7	8