

Элементы ЯПВУ

Данные в программах и алгоритмах

Программы и их алгоритмы пишутся для обработки данных. Чтобы реализовать алгоритм, программам необходимо работать с **данными**: объектами, которые несут в себе информацию, предназначенную для использования.

Данные - это отдельные факты, характеризующие объекты, процессы и явления предметной области, а также их свойства.

Данные можно представлять как собственно данные и информацию.

Собственно данные – это формальные (синтаксические) характеристики описания свойств объектов, процессов и явлений.

Информация – это смысловые (семантические) характеристики описания свойств объектов, процессов и явлений. Информация появляется при наличии субъекта интерпретирующего смысл содержащийся в данных.

Данные могут быть числами, символами, множеством чисел или символов, а также иметь более сложную структуру.

Константы – элементы данных, который занимает место в памяти ЭВМ, имеет имя и определенный тип, причем значение константы никогда не меняется.

Переменные – именованная часть памяти ЭВМ, в которую могут помещаться разные значения переменной. Причем в каждый момент времени переменная имеет единственное значение. В процессе выполнения программы значение переменной может изменяться.

Тип переменных определяется типом данных, которые они представляют.

Кроме различия между переменными и константами существует еще различие между типами данных.

Тип данных - характеристика набора данных, которая определяет:

- способ хранения этих значений в памяти (количество ячеек оперативной памяти компьютера – байт, отведенных для данных этого типа);
- диапазон возможных значений данных из набора;
- допустимые операции, которые можно выполнять над этими значениями;

Данные в программах и алгоритмах

Pascal

Алфавит

C

Имена данных в программе строятся на основе алфавита конкретного языка программирования.

1. Прописные и строчные латинские и русские буквы, а также символ "подчеркивание", он считается буквой
2. Арабские цифр: от 0 до 9
3. Двадцать два специальных символа: + - * / = > < . , ; : @ ' () [] { } # \$ ^
4. Пробельные символы: любые символы ASCII в диапазоне кодов от 0 до 32

1. Прописные и строчные латинские и русские буквы, а также символ "подчеркивание", он считается буквой
2. Арабские цифр: от 0 до 9, шестнадцатеричные цифры от A до F
3. Набор специальных символов: " { } , | [] + - % / \ ; ' : ? < > = ! & # ~ ^ . *
4. Пробельные символы: пробел, символы табуляции, символы перехода на новую строку

Из символов алфавита формируются лексемы языка :

- идентификаторы – имена объектов программы (состоят из букв и цифр, начинаются с буквы)
- значения констант
- ключевые (зарезервированные) слова – зависят от ЯП
- знаки операций
- разделители – пробелы, скобки и др.

Данные в программах и алгоритмах

Типы данных делятся на две группы: простые и составные, состоящие из элементов простых типов.

Простые типы данных

Целые (целочисленные). Набор целых чисел. В памяти для переменной этого типа обычно выделяется 2 байта. Принимают значения из промежутка от -32768 до 32767.

Вещественные. Набор чисел состоящих из целой и дробной части, разделённых десятичной точкой. В памяти выделяется 4 или 6 байт, количество цифр после запятой до 11-12. Принимают значения из промежутка (по модулю) от $2.9E-39$ до $1.7E+38$.

Символьные. Набор символов алфавита ЯП. В качестве своего значения могут иметь один символ. В памяти для переменной этого типа выделяется 1 байт.

Логические. Флаг или переключатель. Принимают только два значения: **True** (Истинно) и **False** (Ложно).

Составные типы данных называют **структурами данных** – это некоторым образом организованная совокупность данных, состоящая из данных простых типов или других структур данных.

Структуры данных это:

Массивы – фиксированный набор элементов одного и того же типа,

Строки – линейно упорядоченная последовательность символов, принадлежащих конечному множеству символов, называемому алфавитом,

Записи – набор элементов (полей данных), характеризующихся различными типами данных,

Файлы – набор записей на внешнем носителе данных.

Подробнее составные типы данных рассматриваются позже, при рассмотрении типовых алгоритмов работы с ними.

Основные типы данных

Pascal

C

Некоторые простые типы

Integer - Целые

принимают значения из промежутка от -32768 до 32767. В памяти для переменной этого типа выделяется 2 байта

Real - Вещественные

принимают значения из промежутка (по модулю) от $2.9E-39$ до $1.7E+38$. В памяти выделяется 6 байт, количество цифр после запятой до 11-12

Boolean - Логические

принимают только два значения: True (истина) и False (ложь). 1 байт.

double – вещественные с удвоенной точностью

8 байт, диапазон приблизительно от $5.0E-324$ до $1.7E+308$.

Char – Литерные (символьные)

в качестве своего значения могут иметь один символ. В памяти для переменной этого типа выделяется 1 байт.

Некоторые стандартные типы данных

int – целые

в памяти занимает 2 байта (на 16- и 32-битовых ЭВМ)

short - короткие целые

в памяти 2 байта, диапазон значений от -32 768 до 32 767

long - длинные целые

4 байта, диапазон от -2 147 483 648 до 2 147 483 647

float - вещественные

4 байта, диапазон приблизительно от $3.4E-38$ до $3.4E+38$.

double – вещественные с удвоенной точностью

8 байт, диапазон приблизительно от $1.7E-308$ до $1.7E+308$.

char - символные

1 байт, от -128 до 127

Примеры описания типов данных И+ПРГ

Pascal

В разделе описаний (декларации) именованные константы обозначаются ключевым словом `const`, а переменные словом `var`.

Тип именованной константы определяется способом записи её значения:

`const`

```
f1 = 13;      - Integer
rg = 3.14;   - Real
c3 = 'A';    - Char
ae6 = False; - Boolean
```

Неименованная константа не объявляется в блоке `const`, а используется в теле кода программы в соответствии со своим значением.

`var`

```
alfa   : Real;
d,fs,t3 : Integer;
u,g    : Char;
flag   : Boolean;
```

Однотипные переменные можно объединять в список, разделяя их запятыми.

```
d,fs,t3 : integer;
```

C

Однотипные переменные можно объединять в список, разделяя их запятыми.

```
int fdr,t,log;
float lok,g56,pot;
char d;
```

Константы. Число без десятичной точки и без показателя степени рассматривается как целое, иначе, как вещественное.

При вводе константы типа `long` нужно указать признак `L` или `l` в конце числа.

Если целое начинается с цифры `0`, то эта константа интерпретируется как восьмеричное число, если же целое начинается с символа `0x` или `0X` – как шестнадцатеричное число.

Инициализация переменных целого типа.

Константы применяются при инициализации переменных – для присваивания переменной некоторого значения перед началом обработки.

```
int dog=5L;
int rad=077;
int nina =0X99;
```

Константе можно давать имя двумя способами:

- С помощью инструкции препроцессора (`#define SIZE 10`)
- С помощью квалификатора (`const int size=10;`), – или использовать как неименованную.

Элементы ЯПВУ

Pascal

Выражения

C

Выражение – это синтаксическая единица языка, определяющая способ **вычисления** некоторого значения.

Выражение состоит из **операндов** (переменных, констант, функций, выражений), **знаков операций** и **скобок**.

Примеры выражений:

dfg; 21; (a + d)*c; sin(t); a>2 (a + 0.12)/6; y && x || !z;

Выражения бывают **математические** (результат вычисления – число) и **логические** (результат вычисления Истинно или Ложно)

Операции

Операции выполняются (выражения вычисляются) в соответствии с **приоритетами**.

Для изменения порядка выполнения операций применяются круглые скобки.

Виды операций (в порядке убывания приоритета при вычислении)

Унарные - not, @

Мультипликативные - *, /, div,
mod, and, shl, shr

Аддитивные - +, -, or, xor

Отношения - =, <>, <, >, <=, >=, in

Унарные - +, -, !, ++, --, sizeof, (type)

Бинарные - *, /, %, +, -, <, <=, >, >=,
==, !=, &, ||, =

Тернарная (3 операнда) - ? :
(условная операция)

Задание на дом:

(выполняется в письменном виде)

- **изучить виды и значения операций в C и Pascal, составить и предъявить преподавателю таблицу основных операций языка (Pascal и C).**
- **изучить стандартные арифметические функции в языке Pascal и математические функции в C, а также способы их подключения к программам. Составить списки функций для каждого ЯП (с указанием типов данных для параметров функции) и предъявить списки преподавателю.**