

# СТРУКТУРА ПРОГРАММЫ НА ЯЗЫКЕ ПАСКАЛЬ

---

- Программа состоит из заголовка и блока.
- В заголовке указывается имя программы и список параметров. Общий вид:

`program n (input, output, x, y,...);`

здесь `n` – имя программы;

`input` – файл ввода;

`output` – файл вывода;

`x, y` – внешние файлы,

используемые в программе.

# БЛОК

---

Блок программы состоит из шести разделов, следующих в строго определенном порядке:

- раздел меток (**label**)
- раздел констант (**const**)
- раздел типов (**type**)
- раздел переменных (**var**)
- раздел процедур и функций
- раздел действий (операторов)

Раздел действий должен присутствовать всегда, остальные разделы могут отсутствовать.

Каждый из первых четырех разделов начинается с соответствующего ключевого слова (**label**, **const**, **type**, **var**), которое записывается один раз в начале раздела и отделяется от последующей информации только пробелом, либо концом строки, либо комментарием.

# РАЗДЕЛ КОНСТАНТ (CONST)

---

Если в программе используются константы, имеющие достаточно громоздкую запись (например, число  $\pi$  с 8-ю знаками), либо сменные константы (для задания варианта программы), то такие константы обычно обозначаются какими-либо именами и описываются в разделе `const`, а в программе используются только имена констант. Это делает программу более наглядной и удобной при отладке и внесении изменений.

Общий вид:

`const a1 = c1; a2 = c2; ...` здесь `a1` – имя константы, `c1` – значение константы.

Пример. `const pi = 3.14; e = 2.71828;`

# РАЗДЕЛ ПЕРЕМЕННЫХ (VAR)

---

Каждая переменная должна быть описана до ее использования в программе и отнесена к одному и только одному типу. Названия разделов (`const`, `type`, `var...`) указываются только один раз.

Пример.

```
var k, i, j: integer; a, b: real;
```

Таким образом, в разделе `var` вводится имя каждой переменной и указывается, к какому типу эта переменная принадлежит. Тип переменной можно задать двумя способами: указать имя типа (например, `real`, `color` и т.д.), либо описать сам тип, например: `array[1..16] of char`

# РАЗДЕЛ ДЕЙСТВИЙ (ОПЕРАТОРОВ)

---

Эта часть программы начинается с ключевого слова **begin** и заканчивается словом **end**, после которого должна стоять точка. Раздел действий есть выполняемая часть программы, состоящая из операторов.

Под оператором в языке Паскаль подразумевают только описание действий. Операторы отделяются друг от друга только точкой с запятой. Если оператор стоит перед **end**, **until** или **else**, то в этом случае точка с запятой не ставится.

# ОПЕРАТОР ПРИСВАИВАНИЯ

---

Общий вид:  $v := a;$

здесь  $v$  – переменная,  $a$  – выражение,  $:=$  - операция присваивания. Выражение  $a$  может содержать константы, переменные, названия функций, знаки операций и скобки.

Пример.  $f := 3 * C + 2 * \sin(x);$

Вид выражения однозначно определяет правила его вычисления: действия выполняются слева направо с соблюдением следующего старшинства (в порядке убывания): 1) not; 2) \*, /, div, mod, and; 3) +, -, or; 4) =, <, >, <>, <=, >=, in.

Любое выражение в скобках вычисляется раньше, чем выполняется операция, предшествующая скобкам.

Присваивание допускается для переменных всех типов, за исключением типа файл.

В операции  $v := a$  переменная  $v$  и выражение  $a$  должны иметь один и тот же тип, а для интервального типа – одно и то же подмножество значений.

**Замечание 1.** Разрешается присваивать переменной типа real выражение типа integer.

**Замечание 2.** Нельзя присваивать переменной типа integer выражение типа real.

# ЦЕЛЫЕ ТИПЫ

---

Переменные целого типа могут принимать только целые значения. Такие переменные в программе описываются следующим образом:

a, b, c: **integer**;

Здесь a, b, c... - имена переменных, **integer** – тип переменных. Транслятор, встретив такое описание переменных a, b, c, запоминает, что эти переменные могут принимать только целые значения и формирует соответственно этому команды программы.

**div** - целочисленное деление (остаток отбрасывается).

**mod** - остаток от деления: a mod b.



# ВЕЩЕСТВЕННЫЕ ТИПЫ

---

В языке Паскаль существует несколько типов для представления действительных чисел. Однако чаще всего для их представления используется тип **real**.

Над действительными числами выполнимы операции сложения (+), вычитания (-), умножения (\*) и деления (/). Результатом этих операций является также действительное число. Даже если хотя бы один из операндов вещественный, то результат этих операций также будет вещественным.

- Функция **int** возвращает в виде действительного значения целую часть аргумента, **frac** возвращает дробную часть аргумента.
- Функции **trunc** и **round** возвращают результат целого типа. Первая отсекает дробную часть от аргумента, а вторая выполняет округление до ближайшего целого.
- Функция **random** без аргументов возвращает равномерно распределенное случайное число от 0 до 1.

# СТАНДАРТНЫЕ ФУНКЦИИ PASCAL, ПРИМЕНИМЫЕ К АРГУМЕНТАМ ЦЕЛЫХ И ДЕЙСТВИТЕЛЬНЫХ ТИПОВ

Функция	Тип результата	Результат выполнения
abs(x)	Целый	Модуль x (абсолютная величина x)
sqr(x)	Целый	Квадрат x
random(x)	Целый	Случайное целое число из интервала 0..x-1.
sin(x)	Действительный	Синус x (угол в радианах)
cos(x)	Действительный	Косинус x (угол в радианах)
ln(x)	Действительный	Натуральный логарифм x
exp(x)	Действительный	Экспонента x
sqrt(x)	Действительный	Квадратный корень из x