
Переменные и константы. Идентификаторы. Операция присваивания. sizeof и size_t. Квалификатор const. Перечисления. Примеры.

Переменные и константы

Переменная – область памяти, предназначенная для хранения данных. В языках высокого уровня переменная характеризуется:

- именем(идентификатором) - позволяет компилятору отличать одну переменную от других.
- **типом данных** - определяет то, какую информацию можно хранить в этой ячейке.

Тип данных определяет:

1. формат представления информации.
2. диапазон хранимых значений.
3. набор допустимых операций.

Примеры переменных:

- `int number;`
- `float x, y, z;`
- `char letter;`

Идентификаторы - зарезервированные слова для обозначения определенных выражений.

Идентификаторы используются для:

1. формирования имен объектов языка: переменных и функций.
 2. ключевых слов языка: типы данных, управляющие конструкции.
-

Примеры идентификаторов:

- `int`, `double`, `char` - типы данных.
 - `for`, `while`, `if` - управляющие конструкции.
-

Операция присваивания

Операция присваивания "=" позволяет задать значение переменной.

Простая инициализация переменной (`int a;`) только резервирует место в памяти;

Операция присваивания (`a = 3;`) задаёт ей значение;

Инициализация с присваиванием значения (`int a = 3;`) резервирует место в памяти и сразу записывает в него значение.

sizeof и size_t

sizeof - оператор, возвращающий размер памяти, который занимает переменная.

- `sizeof(int)` - возвращает 4
 - `sizeof(char)` - возвращает 1
 - `sizeof var` - возвращает количество занимаемых байт памяти переменной `var`.
`sizeof` возвращает целочисленное значение типа `size_t`.
-

size_t - беззнаковый целочисленный тип данных, имеет размер в два раза больше чем `int`.

Возвращается в результате вызова оператора `sizeof`.

Квалификатор const

Константа – именованное постоянное значение определенного типа данных, которое нельзя изменить в процессе выполнения команды, задается с помощью `const`.

```
const char NAME[] = "cool name";
```

```
const int AGE = 18;
```

Константы с модификатором `const` принято описывать внутри блока `int main()`.

Перечисления

Перечисление (enum) - тип данных, который используется для объединения констант, которые являются целочисленными.

Варианты создания перечислений:

```
{
    // создаем перечисление с именем eDirection
    enum eDirection {RIGHT, LEFT, DOWN, UP};

    // создаем переменную dir с перечислимым типом eDirection
    enum eDirection dir;

    // присваиваем переменной dir константу UP
    dir = UP;
}
```

```
// создаем перечисление без имени, просто набор констант
enum {RIGHT, LEFT, DOWN, UP};

// создаем переменную dir с перечислимым типом int
int dir; // присваиваем переменной dir константу UP
dir = UP;
```

Все элементы перечисления по умолчанию имеют значение равное нумерации константы.

```
enum eDirection
{
    RIGHT, // по умолчанию = 0
    LEFT,  // = 1
    DOWN,  // = 2
    UP     // = 3
};
```

Если же задать значение для какой либо константы вручную, то значение следующей будет больше на 1, а не равное нумерации:

```
enum eDirection
{
    RIGHT = 8,    // = 8
    LEFT,         // = 9
    DOWN = 100,   // = 100
    UP            // = 101
};
```