



Язык программирования C

Тема 1.9 Базовые типы и агрегаты данных в языке C

Типы данных языка Си++

- Целые числа различных размеров со знаком или без
 - int, short, char
- Числа с плавающей запятой различной размерности
 - float, double, long double
- Логический тип
 - bool
- Перечисляемые типы (enum)
- Структуры (struct)
- Объединения (union)
- Массивы

Базовые типы данных

- Типы данных целых чисел
 - char
 - int
 - квалификаторы
 - short/long
 - unsigned/signed
- Логический тип
 - bool
- Типы данных вещественных чисел
 - float
 - double

Константы перечисления

- **Задают список целых констант**
 - `enum WeekDay`
 - `{Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday};`
- **Особенности**
 - Имена в различных перечислениях должны отличаться друг от друга
 - Значения внутри одного перечисления могут совпадать:
 - `enum Status {Ok, Failure, Success = Ok}`

Ключевое слово `typedef`

- Язык Си++ предоставляет оператор **`typedef`**, позволяющий давать типам данных новые имена
 - После этого новое имя типа может использоваться в качестве синонима оригинала
- Причины использования **`typedef`**
 - Решение проблемы переносимости
 - Желание сделать текст программы более ясным

Пример использования оператора typedef

```
typedef int Length;  
Length len, maxlen;  
len = 1;  
  
typedef int int32;  
typedef short int16;  
typedef char int8;  
  
int32 counter = 0;
```

Перечисляемые типы данных (перечисления)

- Позволяет задать ограниченный набор именованных целочисленных значений
 - День недели
 - Состояние конечного автомата
 - Тип файла
 - Модель компьютера и т.д

Структуры

- **Структура** - это одна или несколько переменных (возможно, различных типов), которые для удобства работы с ними сгруппированы под одним именем.
 - Структуры помогают в организации сложных данных, позволяя группу связанных между собой переменных трактовать не как множество отдельных элементов, а как единое целое

Пример использования структур

```
typedef struct tagPoint
{
    int x;
    int y;
}Point;

typedef struct tagRectangle
{
    Point leftTop;
    Point rightBottom;
}Rectangle;

int main()
{
    Point p0 = {0, 3};
    Point p1 = {30, 20};
    Rectangle r1 = {{1,1}, {20, 30}};

    Rectangle r2;

    r2.leftTop = p0;
    r2.rightBottom = r1.rightBottom;

    return 0;
```

Агрегат данных

- Агрегат данных - это функция, которая возвращает единственное значение, основанное на множестве строк.

Модификаторы в объявлениях

-
- Модификаторы `signed`, `unsigned`, `long` и `short` могут применяться к целочисленным типам. К символам можно применять `signed` и `unsigned`, `long` может применяться к типу `double`

Спецификаторы в объявлениях

- Спецификаторы типа в объявлениях определяют тип объявления переменной или функции.
- Спецификаторы типов:
- void, char, short, int, long, float, double

Объединения

- **Объединение** - это тип данных, который может содержать (в разные моменты времени) объекты различных типов и размеров
 - Объединения позволяют хранить разнородные данные в одной и той же области памяти без включения в программу машинно-зависимой информации

Массивы

- Простая статическая структура данных, предназначенная для хранения набора единиц данных, каждая из которых идентифицируется индексом или набором **индексов**
 - **Индекс** —целое число, либо значение типа, приводимого к целому, указывающее на конкретный элемент массива
 - Количество используемых индексов определяет **размерность** массива

```
#include <stdio.h>

int main()
{
    int studentRatings[3] = {5, 4, 4};
    char student0[] = "Bill";
    char student1[] = {'J', 'o', 'h', 'n', '\0'};
    char student2[6] = "Peter";

    printf("Students' marks:\n");
    printf("1. %s - %d\n", student0, studentRatings[0]);
    printf("2. %s - %d\n", student1, studentRatings[1]);
    printf("3. %s - %d\n", student2, studentRatings[2]);

    return 0;
}
```

Students' marks:

1. Bill - 5
2. John - 4
3. Peter - 4