

## 15. Указатели, массивы, адресная арифметика.

План ответа:

1. понятие «указатель»;
2. связь между указателями и статическими массивами;
3. адресная арифметика (сложение указателя с числом, сравнение указателей, вычитание указателей).

### 1. понятие «указатель»;

Указатель – это объект, содержащий адрес объекта или функции, либо выражение, обозначающее адрес объекта или функции. [Жешке Р. «Толковый словарь стандартного языка Си»]

`int a = 2, *p = &a;        // p – указатель-объект, &a – указатель-выражение`

### 2. связь между указателями и статическими массивами;

Результат выражения, состоящего из имени массива, представляет собой адрес с области памяти, выделенной под этот массив

```
int a[10], *pa;
pa = a; // pa = &a[0]; но НЕ pa = &a;
```

Исключение 1: операция `sizeof` для массива

```
int a[10];
int *pa = a;
printf("sizeof(a) = %d\n", sizeof(a));
// sizeof(a) = 40
printf("sizeof(pa) = %d\n", sizeof(pa));
// sizeof(pa) = 4
```

Исключение 2: массив – операнд операции получения адреса

```
int a[10];
int *pa;
printf("a = %p, &a = %p\n", a, &a);
// a = 0022fefff0, &a = 0022fefff0
&a возвращает указатель, НО тип этого указателя - указатель на массив (не указатель на целое)
pa = &a; // warning: assignment from incompatible pointer type [enabled by default]
```

Исключение 3: строковый литерал-инициализатор массива `char[]`

```
char a[] = "abcdef";
printf("sizeof(a) = %d\n", sizeof(a));
// sizeof(a) = 7
```

При определении массива и указателя под соответствующие переменные выделяется разное количество памяти

Выражению из имени массива нельзя присвоить другое значение

### 3. адресная арифметика (сложение указателя с числом, сравнение указателей, вычитание указателей).

#### 3.1 сложение указателя с числом

```
тип *p = ...;

p += n;
новый адрес в p = старый адрес из p + n * sizeof(тип)

p -= m;
новый адрес в p = старый адрес из p - m * sizeof(тип)

int a[10];
&a[i] == a + i
Выражение a[i] компилятор заменяет на *(a + i)
```

#### 3.2 сравнение указателей

Указатели допускается использовать в операциях сравнения.

При сравнении указателей сравниваются адреса.

При этом можно сравнивать указатель с `NULL` и сравнивать два однотипных указателя.

### 3.3 вычитание указателей

тип a[10];

тип \*pa = a, \*pb = &a[2];

pb - pa = (адрес из pb - адрес из pa) / sizeof(тип)