

```
1  #include <iostream>
2  #include <cstdlib>
3
4  using namespace std;
5
6  //1. Указатели
7
8  int main1()
9  {
10     int a = 5;
11     int b = 12;
12
13     int* ptr1;
14     // int* - это тип переменной, означающий, что переменная
15     // хранит в себе адрес ячейки памяти, в которой хранится
16     // целое(int) чисто
17     // int* - это указатель
18
19     ptr1 = &a;
20     // присваиваем переменной "ptr1" АДРЕС ячейки
21     // памяти, соответствующей переменной "a"
22     // "&" - операция извлечения адреса
23
24     cout << "ptr1=" << ptr1 << endl;
25     //выведет на экран какое-то целое число
26
27     cout << "*ptr1=" << *ptr1 << endl;
28     // выведет значение переменной "a"
29     // "*" - операция разыменования указателя, то есть
30     // получения значения, хранящегося в той ячейке,
31     // на которую указатель указывает
32     // (адрес которой хранит указатель)
33
34     // семантика символа "*" в выражении "int* ptr1" и
35     // в выражении "*ptr1" ПРИНЦИПИАЛЬНО ОТЛИЧАЕТСЯ!
36
37     *ptr1 = 7;
38     // помещение в ячейку, на которую указывает указатель "ptr1",
39     // значения 7
40     // данная операция изменяет значение переменной "a",
41     // хотя сама переменная "a" в этой операции формально
42     // не участвует
43
44     cout << "a=" << a << endl; //выведет 7
45
46     ptr1 = &b;
47     // присваиваем переменной "ptr1" АДРЕС ячейки
48     // памяти, соответствующей переменной "b"
49     // фактически мы отвязываем "ptr1" от переменной "a" и
50     // привязываем к переменной "b"
51
52     cout << "*ptr1=" << *ptr1 << endl;
53     // выведет значение переменной "b"
54
55
56     return 0;
57 }
58
59 //2. Указатели и массивы
60
61 int main2()
62 {
63     const unsigned int size = 5;
64
65     float A[size] = {5.2, 6.7, 1.8, 2.4, -9.0};
66     // имя массива - это указатель!
67
68     cout << "*A=" << *A << endl; //выведет 5.2
69
70     float* ptr = A; //присвоение "ptr" значения переменной "A"
71     // операция "&" не используется, так как A само по себе
72     // является указателем
```

```
73
74     cout << "*ptr=" << *ptr << endl; //выведет 5.2
75
76     cout << "ptr+1=" << ptr + 1 << endl;
77     //выведет адрес второго элемента массива
78
79     cout << "*(ptr+1)=" << *(ptr + 1) << endl; //выведет 6.7
80     cout << "ptr[1]=" << ptr[1] << endl; //выведет 6.7
81
82     // *(ptr+i) и ptr[i] - эквивалентны
83
84
85     //вывод всех элементов массива
86     for (int i = 0; i < size; i++)
87     {
88         cout << *(ptr + i) << " ";
89     }
90
91     cout << endl;
92
93     return 0;
94 }
95
```