

Указатель на указатель

`int **ptrptr; // указатель на указателя типа int (две звёздочки)`

```
4 #include <iostream>
5 int main()
6 {
7 int value = 7;
8
9 int *ptr = &value;
1 std::cout << *ptr << std::endl; // разыменовываем
0 указатель, чтобы получить значение типа int
1
1 int **ptrptr = &ptr;
1 std::cout << **ptrptr << std::endl;
2
1 return 0;
3 }
1
4
```

Массив указателей

```
int **array = new int*[20]; // выделяем массив из 20 указателей типа  
int
```

Это тот же обычный динамически выделенный массив, за исключением того, что элементами являются указатели на тип `int`, а не значения типа `int`.

Двумерные массивы

```
int array[15][7]; //классический способ
```

```
int **array = new int*[15]; // выделяем массив из 15 указателей  
типа int — это наши строки
```

```
for (int count = 0; count < 15; ++count)
```

```
    array[count] = new int[7]; // а это наши столбцы
```

Треугольная матрица

- `int **array = new int*[15]; // выделяем массив из 15 указателей типа int — это наши строки`
- `for (int count = 0; count < 15; ++count)`
- `array[count] = new int[count+1]; // а это наши столбцы`

Удаление

```
for (int count = 0; count < 15; ++count)
```

```
    delete[] array[count];
```

```
delete[] array; // это следует выполнять в конце
```

Преобразование в одномерный

```
// Вместо следующего:  
2 int **array = new int*[15]; // выделяем массив из 15 указателей типа  
3 int — это наши строки  
4 for (int count = 0; count < 15; ++count)  
5     array[count] = new int[7]; // а это наши столбцы  
6 // Делаем следующее:  
7 int *array = new int[105]; // двумерный массив 15x7 "сплющенный" в  
одномерный массив
```

Практика