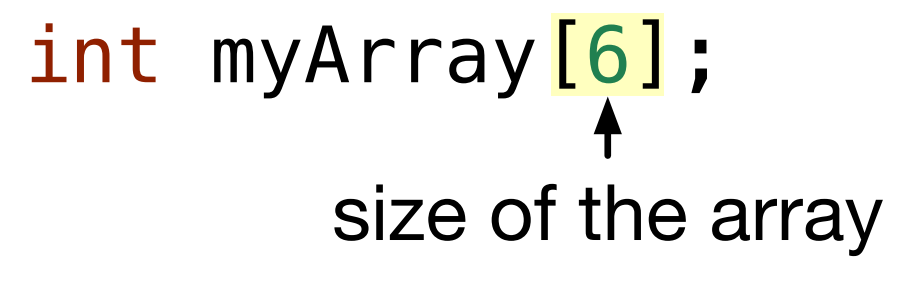
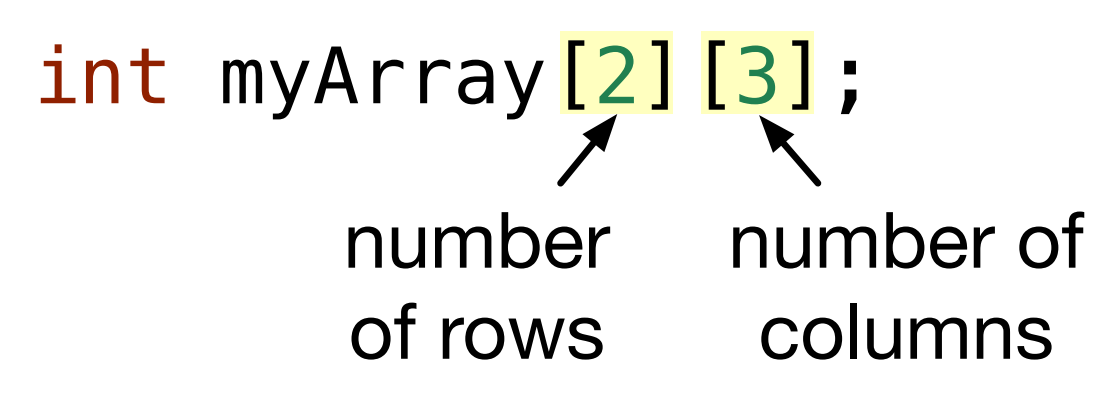
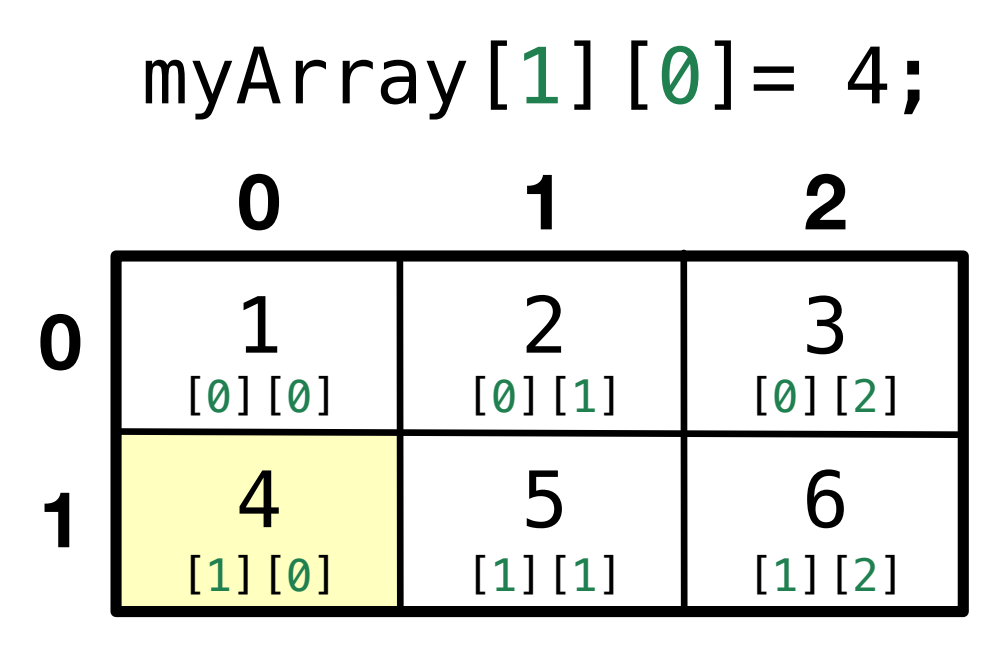


```c
int myArray[6];
```



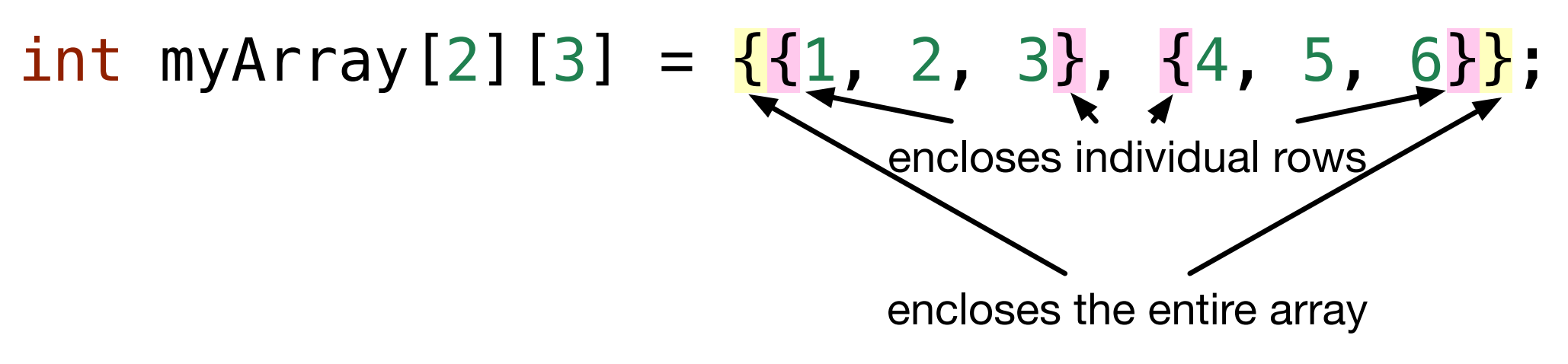
```c
int myArray[2][3];
```



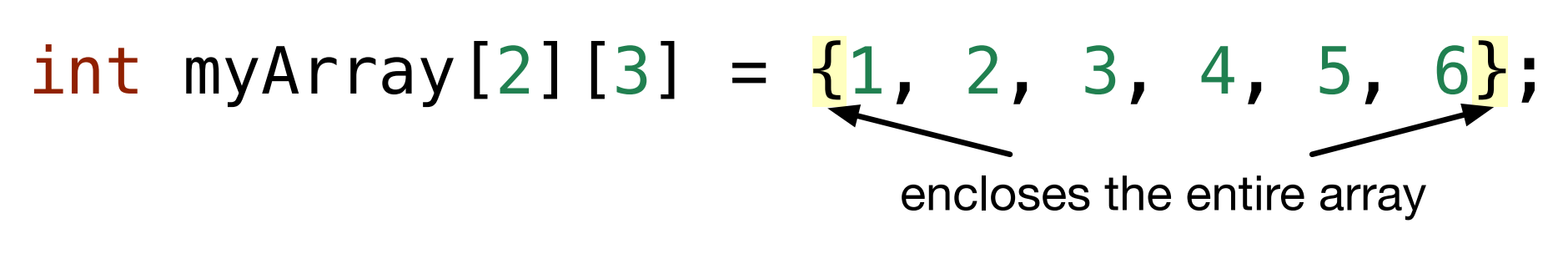
```c
myArray[1][0]= 4;
```

	0	1	2
0	1 [0][0]	2 [0][1]	3 [0][2]
1	4 [1][0]	5 [1][1]	6 [1][2]

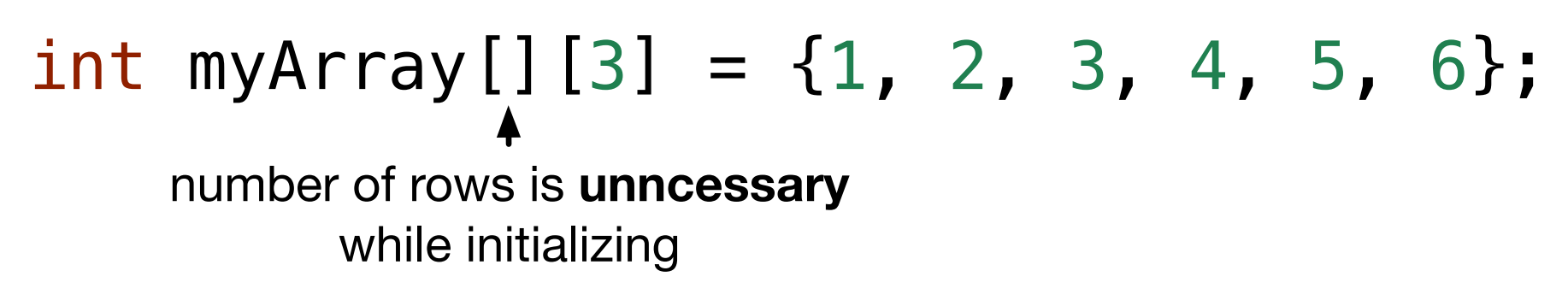
```c
int myArray[2][3] = {{1, 2, 3}, {4, 5, 5}};
```



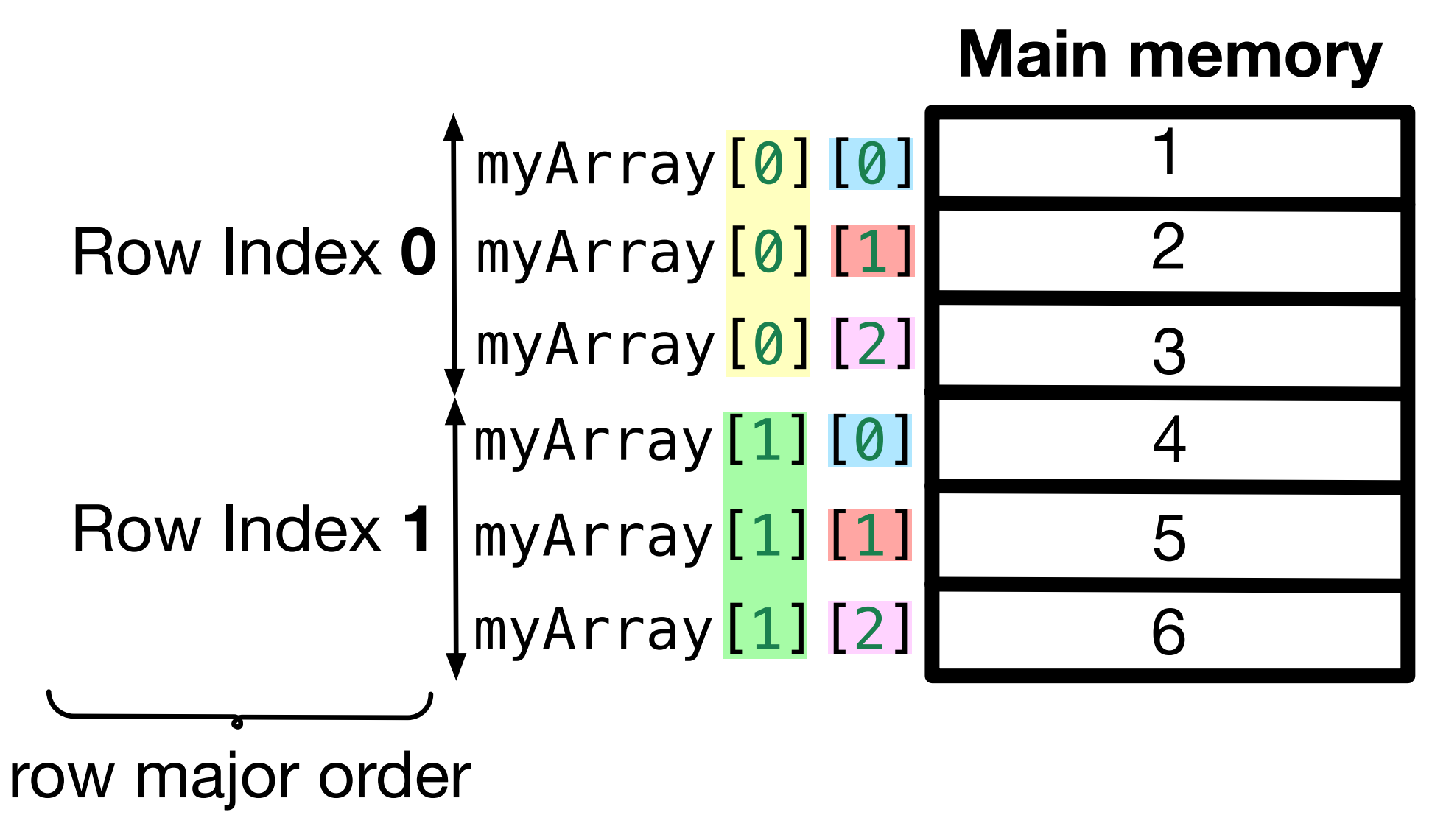
```c
int myArray[2][3] = {{1, 2, 3}, {4, 5, 6}};
```

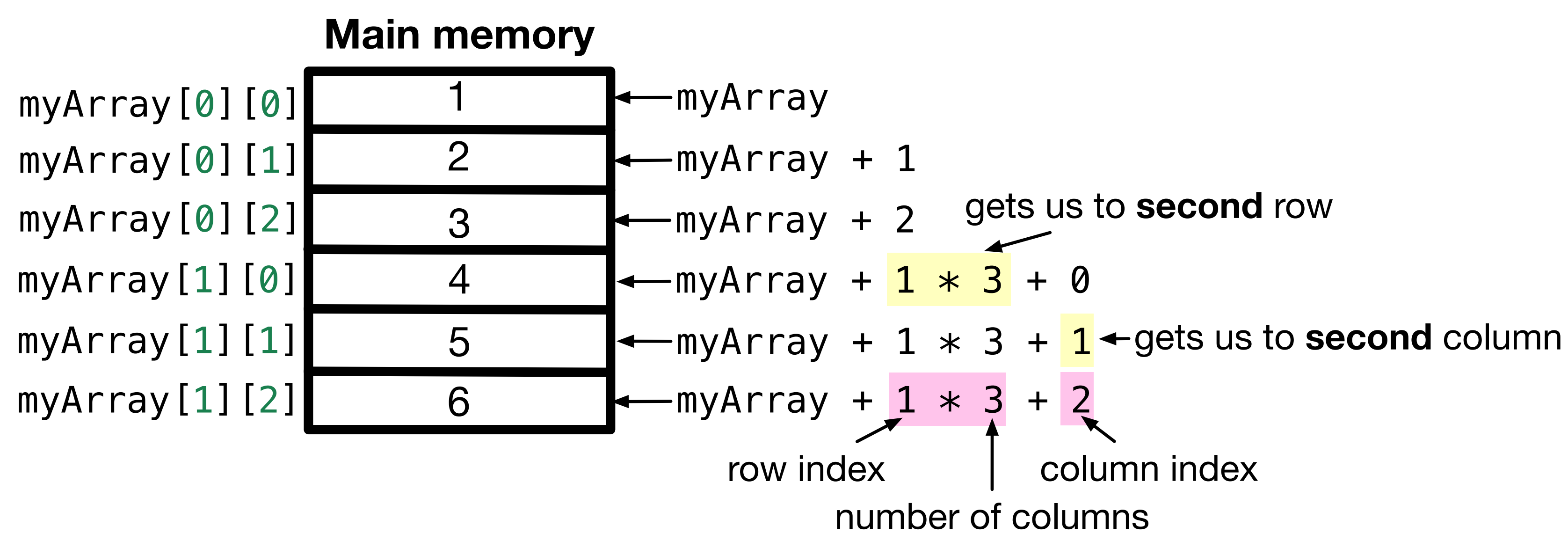


```c
int myArray[2][3] = {1, 2, 3, 4, 5, 6};
```



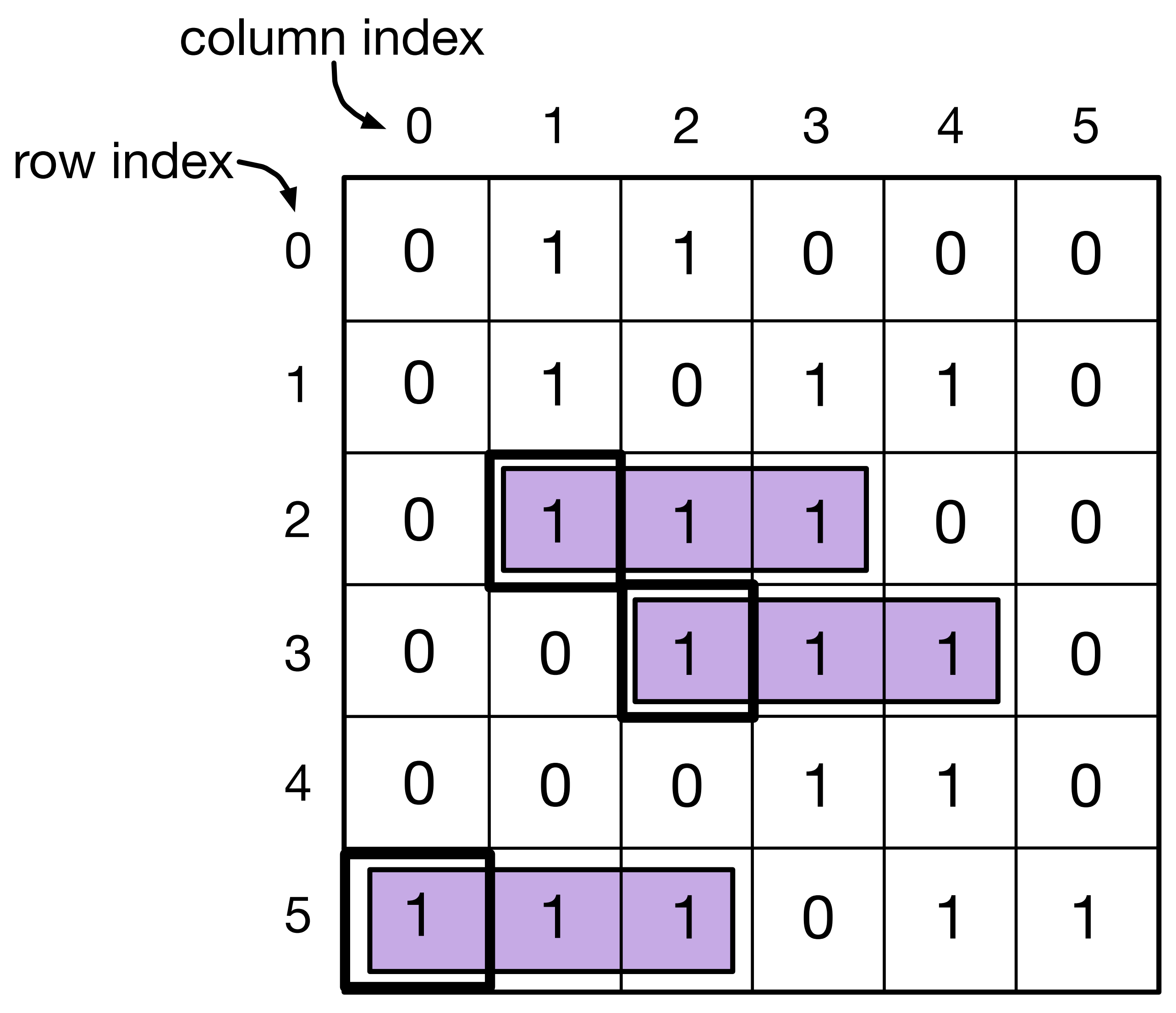
```c
int myArray[][3] = {1, 2, 3, 4, 5, 6};
```



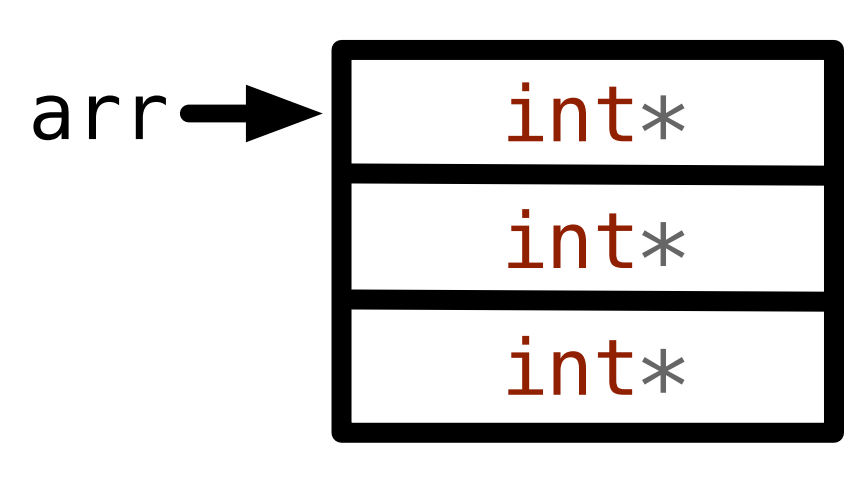


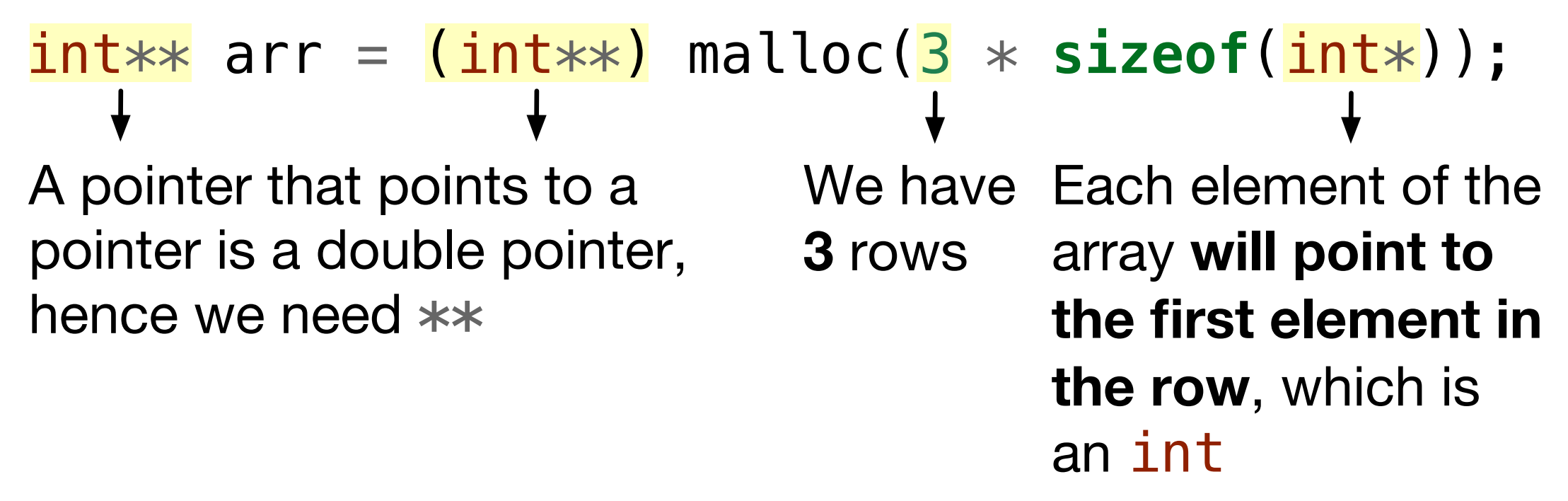
```c
myArray[i][j] ↔ *(myArray + i * <num of columns> + j)

&myArray[i][j]↔ myArray + i * <num of columns> + j
```

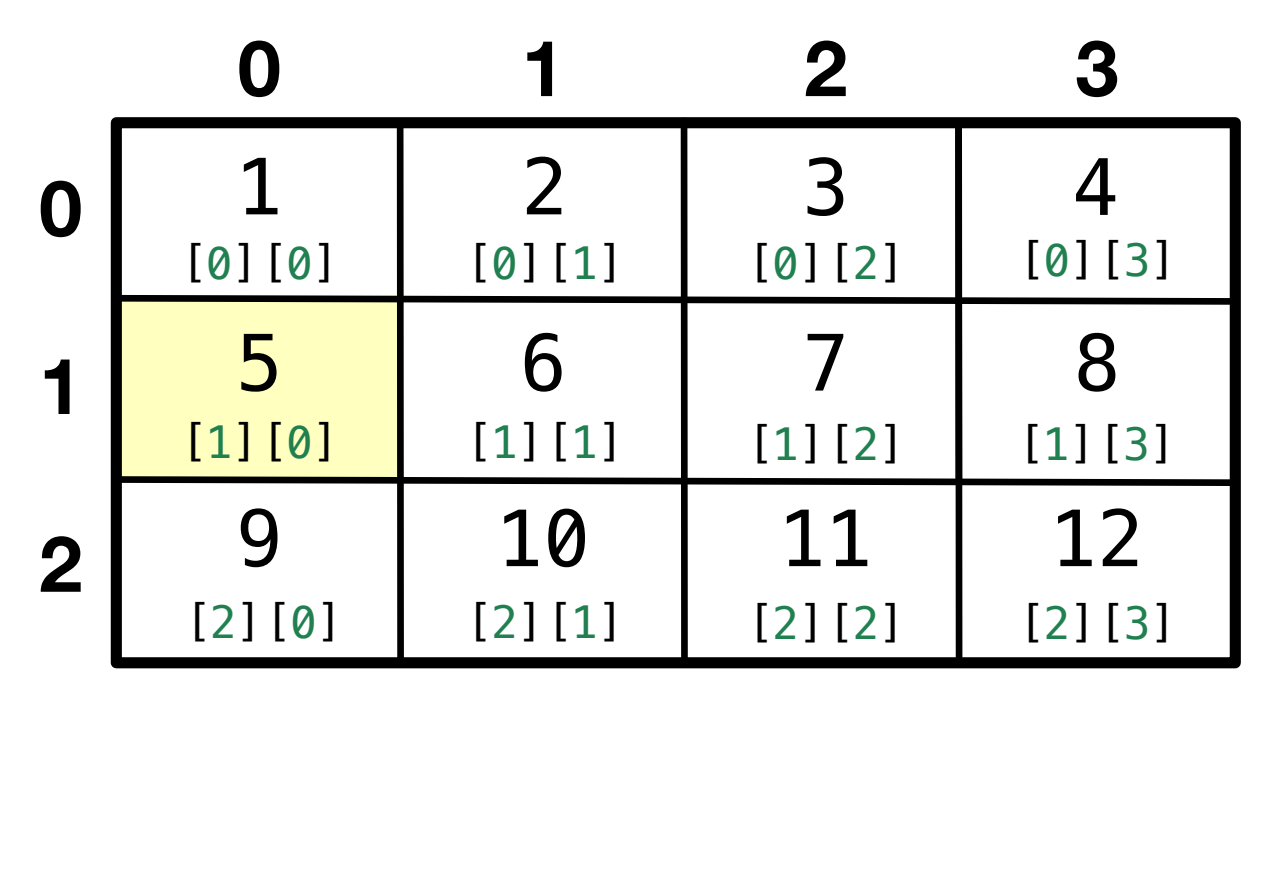


	0	1	2	3	4	5
0	0	1	1	0	0	0
1	0	1	0	1	1	0
2	0	1	1	1	0	0
3	0	0	1	1	1	0
4	0	0	0	1	1	0
5	1	1	1	0	1	1

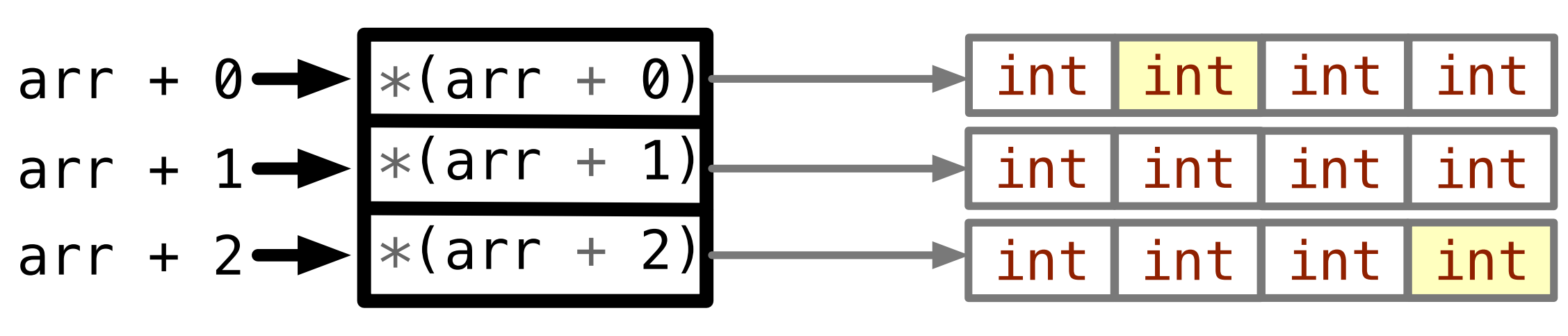


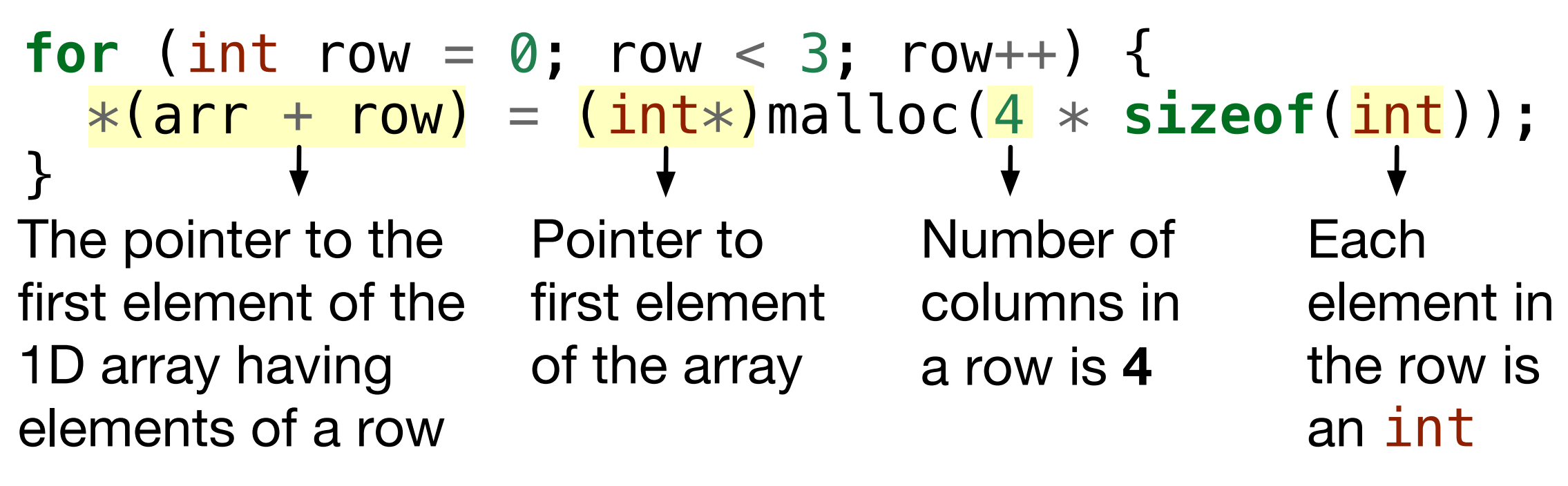


```c
int** arr = (int**) malloc(3 * sizeof(int*));
```

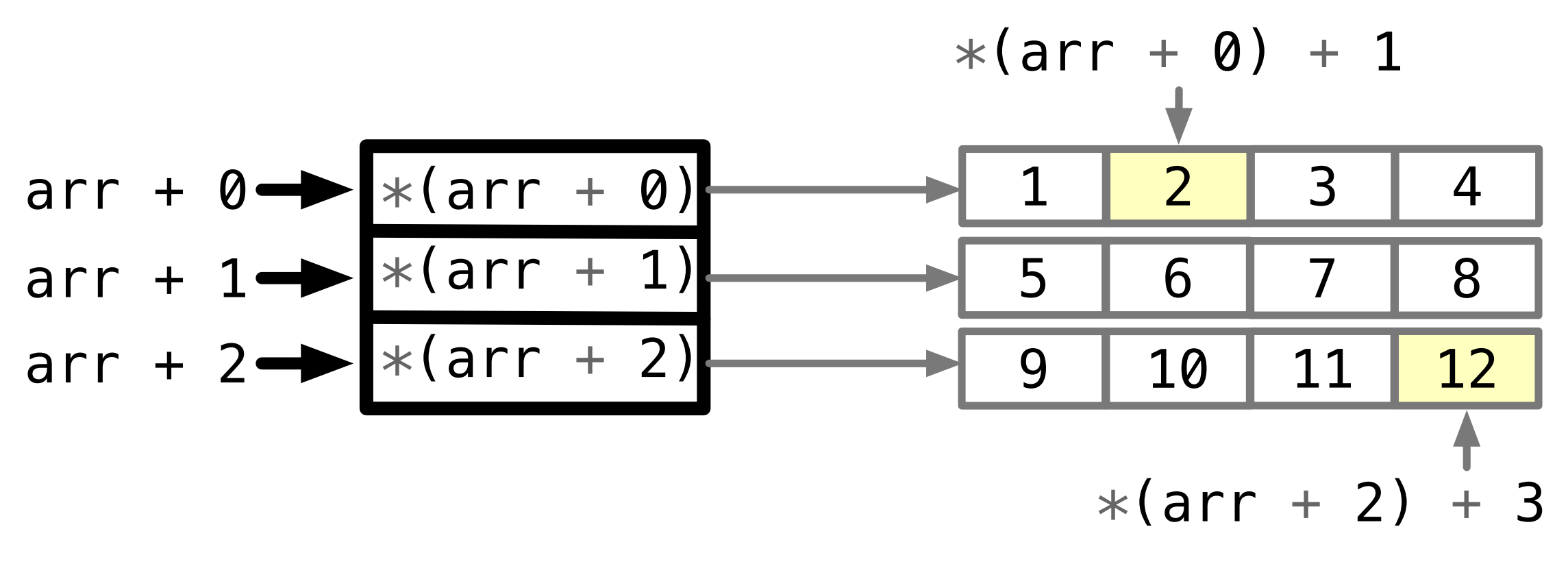


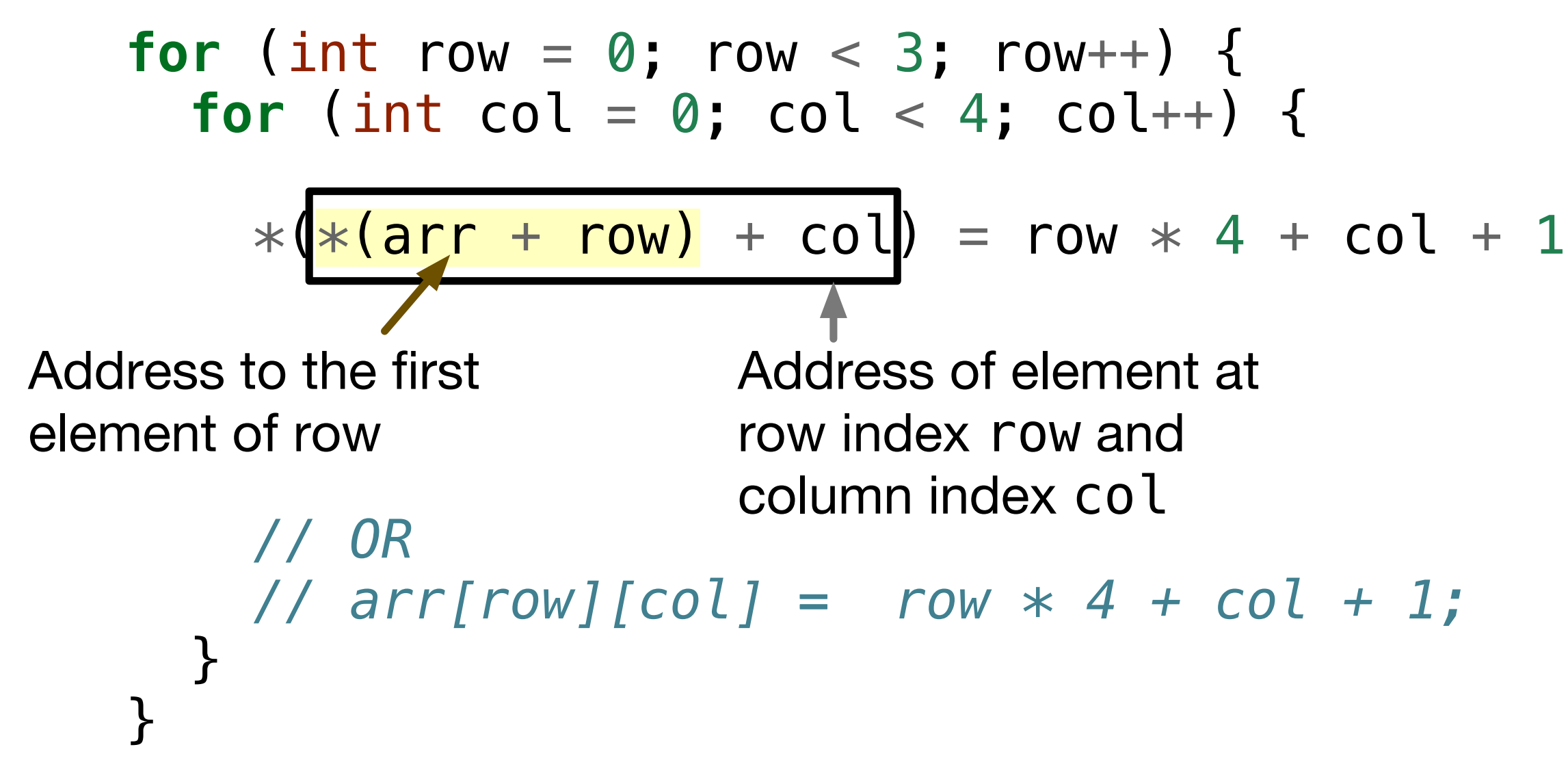
	0	1	2	3
0	1 [0][0]	2 [0][1]	3 [0][2]	4 [0][3]
1	5 [1][0]	6 [1][1]	7 [1][2]	8 [1][3]
2	9 [2][0]	10 [2][1]	11 [2][2]	12 [2][3]





```c
for (int row = 0; row < 3; row++) {
  *(arr + row) = (int*)malloc(4 * sizeof(int));
}
```





```c
for (int row = 0; row < 3; row++) {
  for (int col = 0; col < 4; col++) {

    *(*(arr + row) + col) = row * 4 + col + 1;

    // OR
    // arr[row][col] =  row * 4 + col + 1;
  }
}
```