PDP
IT-ACADEMY

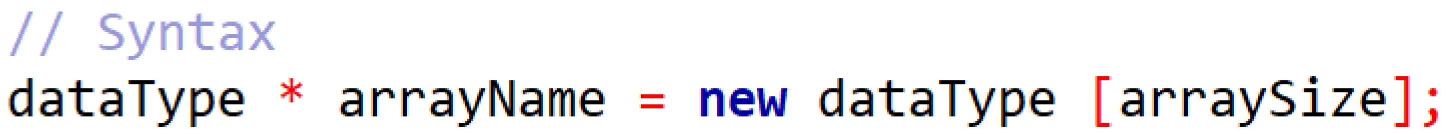
Dinamik massivlar bilan ishlash

Reja:

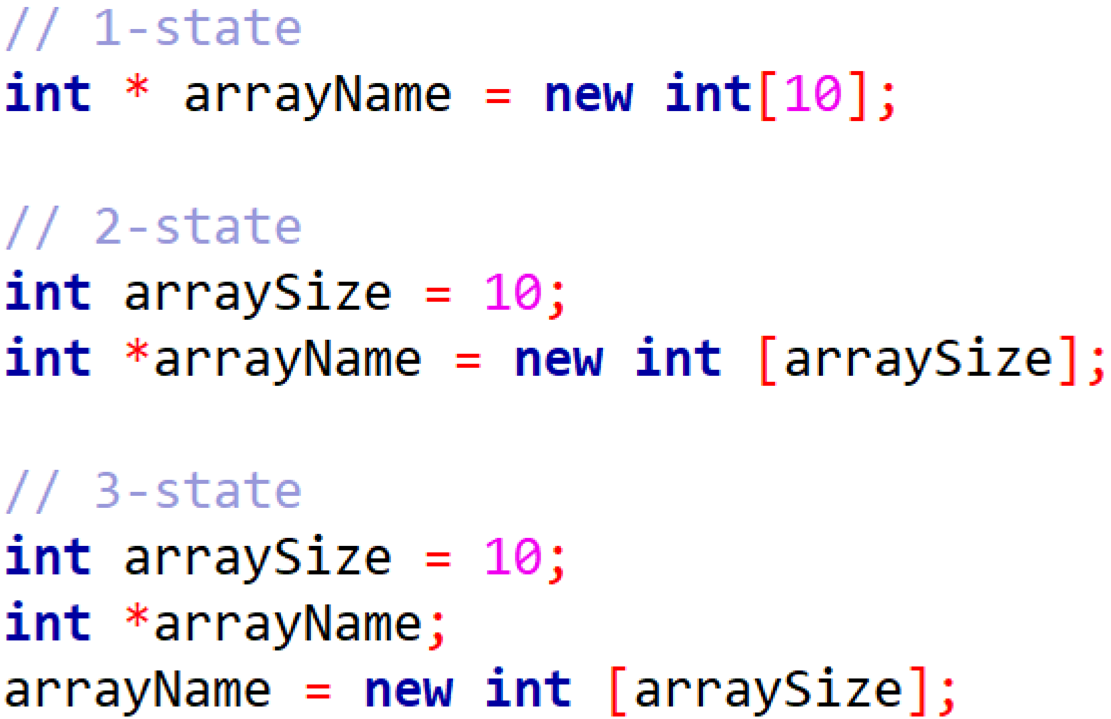
- Dinamik massivni e’lon qilish
- Massiv funksiya parametri sifatida
- Massiv funksiyadan qaytuvchi qiymat sifatida
- Dinamik massiv uchun ajratilgan xotirani bo‘shatish
- Amaliy mashqlar

PDP
IT-ACADEMY

Dinamik massivni e’lon qilish

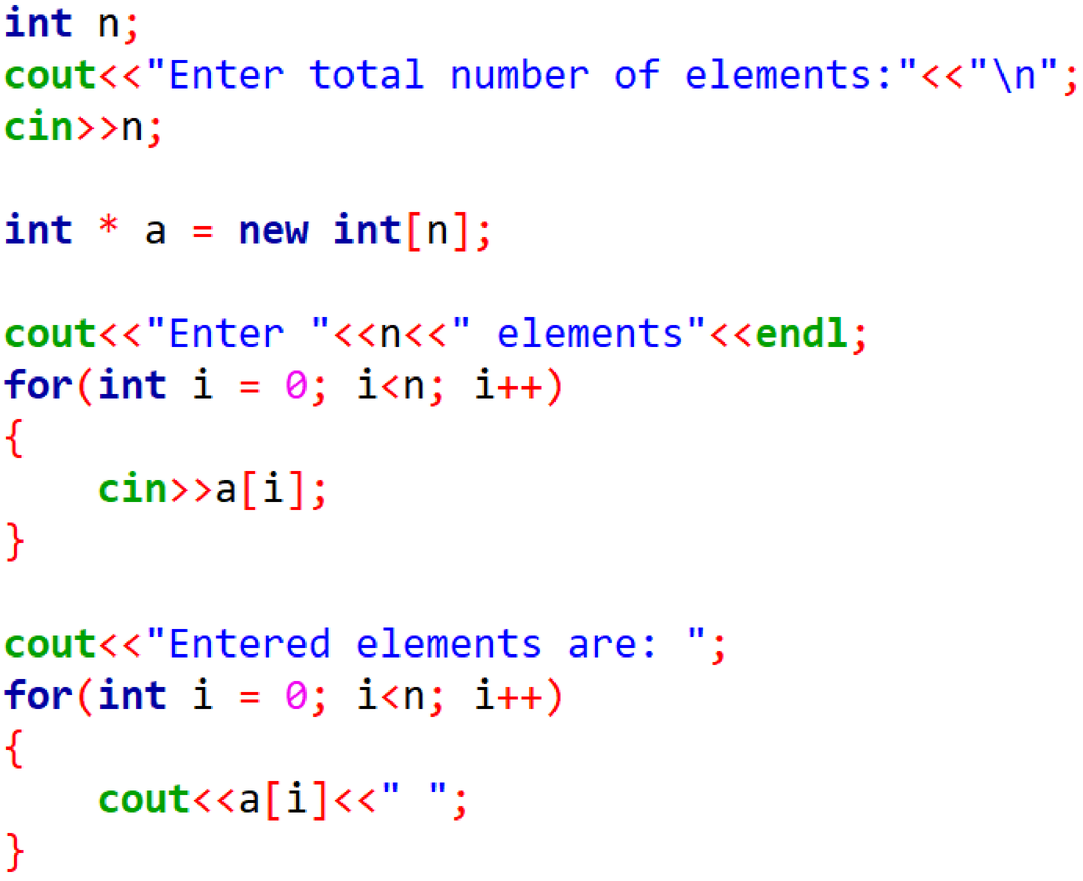


```
// Syntax
dataType * arrayName = new dataType [arraySize];
```

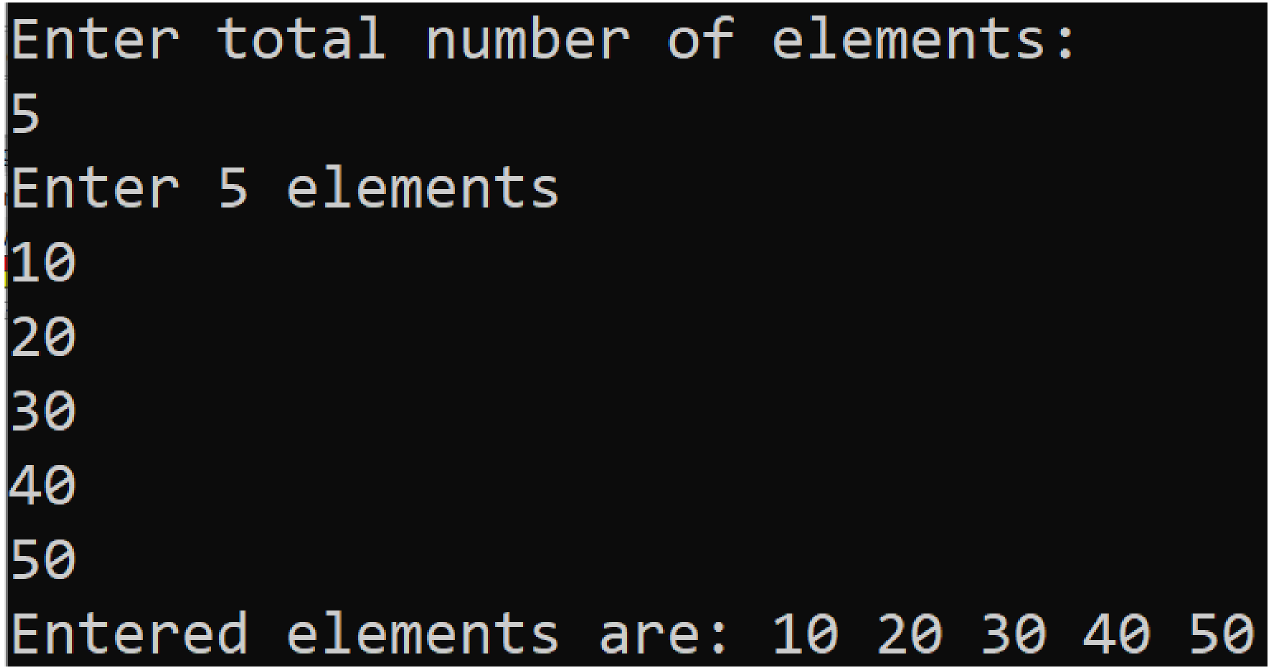


```
// 1-state
int * arrayName = new int[10];

// 2-state
int arraySize = 10;
int *arrayName = new int [arraySize];

// 3-state
int arraySize = 10;
int *arrayName;
arrayName = new int [arraySize];
```

Example

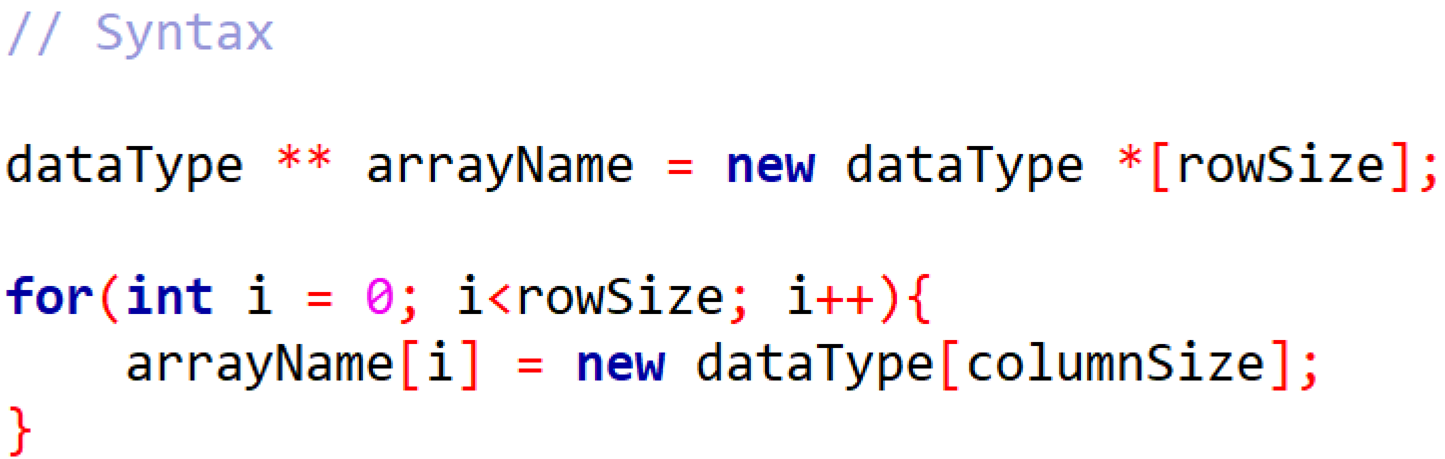


```
int n;
cout<<"Enter total number of elements:"<<"\n";
cin>>n;

int * a = new int[n];

cout<<"Enter "<<n<<" elements"<<endl;
for(int i = 0; i<n; i++)
{
    cin>>a[i];
}

cout<<"Entered elements are: ";
for(int i = 0; i<n; i++)
{
    cout<<a[i]<<" ";
}
```



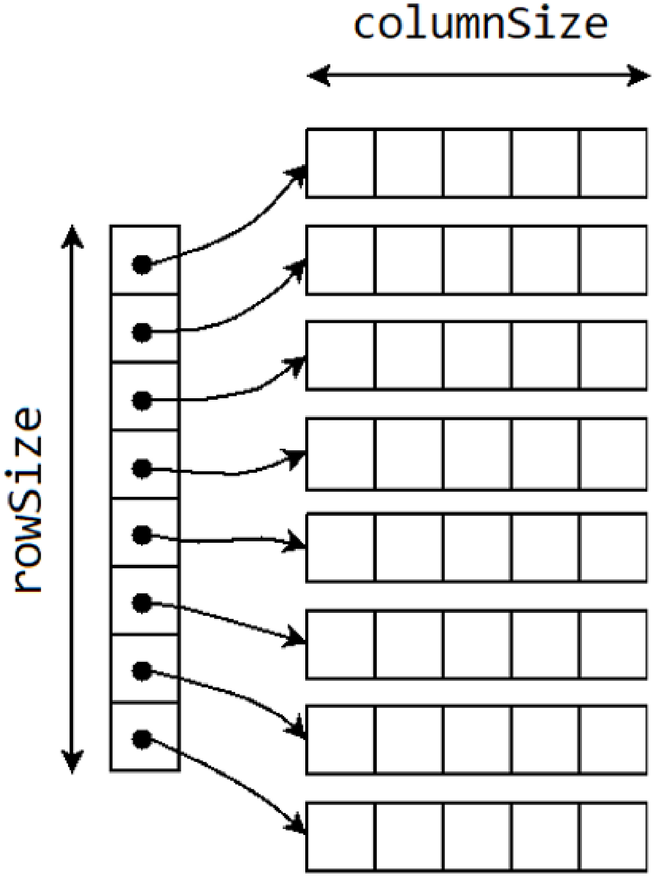
```
Enter total number of elements:
5
Enter 5 elements
10
20
30
40
50
Entered elements are: 10 20 30 40 50
```

2D multidimensional array



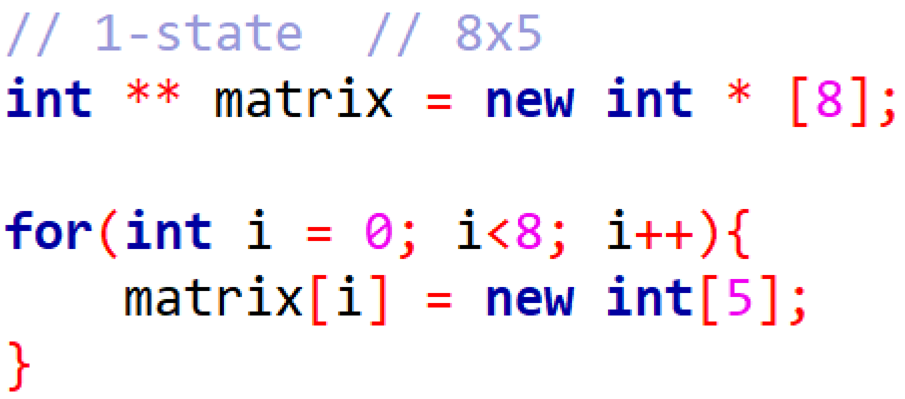
```
// Syntax

dataType ** arrayName = new dataType *[rowSize];

for(int i = 0; i<rowSize; i++){
    arrayName[i] = new dataType[columnSize];
}
```

2D multidimensional array



PDP
IT-ACADEMY

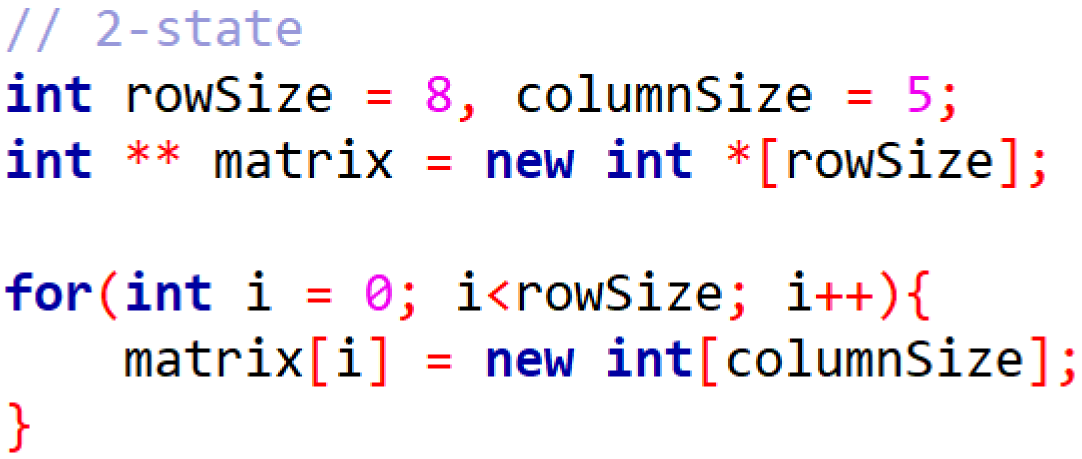
2D multidimensional array



```
// 1-state  // 8x5
int ** matrix = new int * [8];

for(int i = 0; i<8; i++){
    matrix[i] = new int[5];
}
```

PDP
IT-ACADEMY

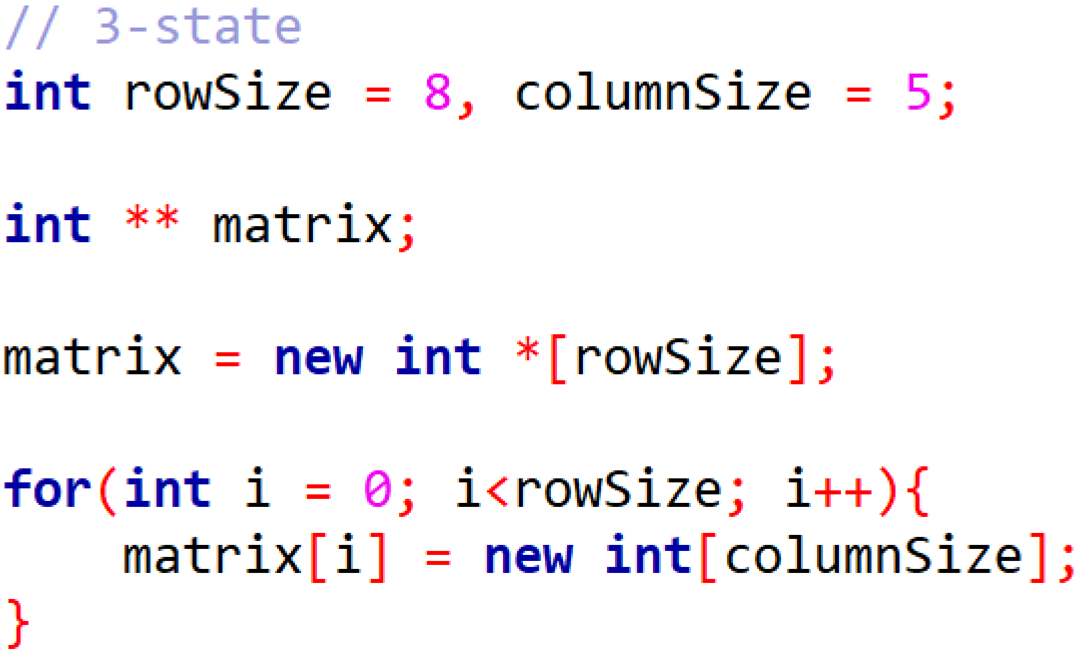
2D multidimensional array



```
// 2-state
int rowSize = 8, columnSize = 5;
int ** matrix = new int *[rowSize];

for(int i = 0; i<rowSize; i++){
    matrix[i] = new int[columnSize];
}
```

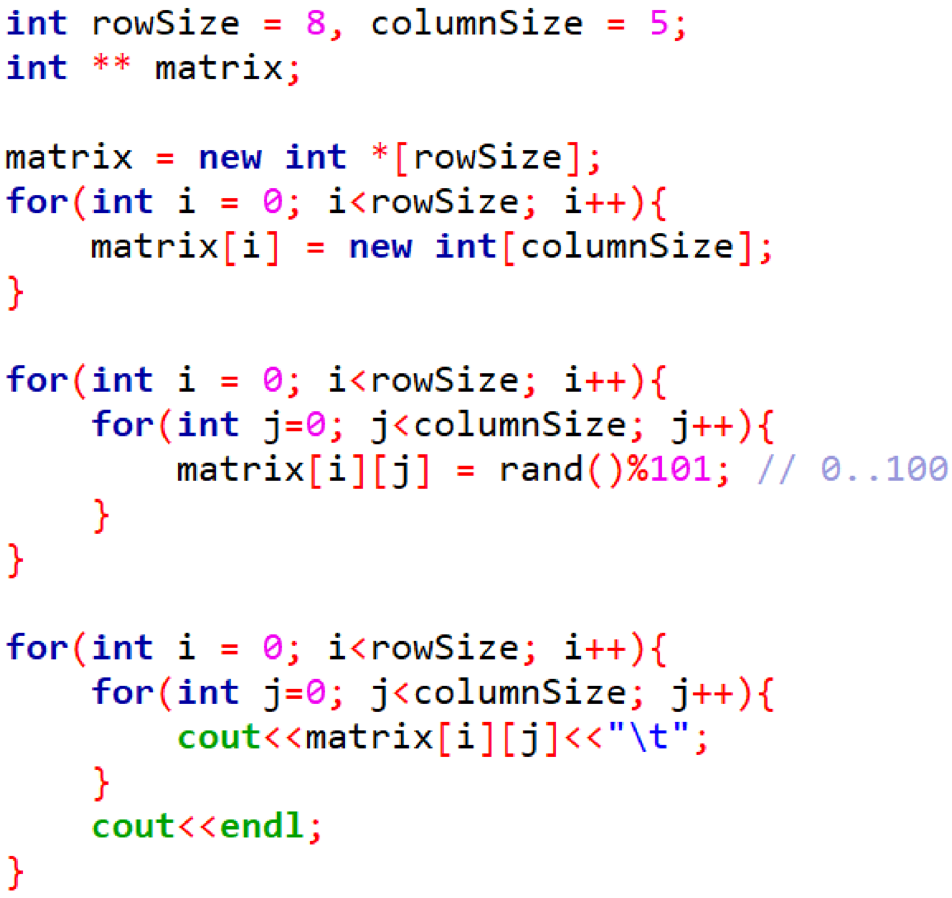
PDP
IT-ACADEMY

2D multidimensional array

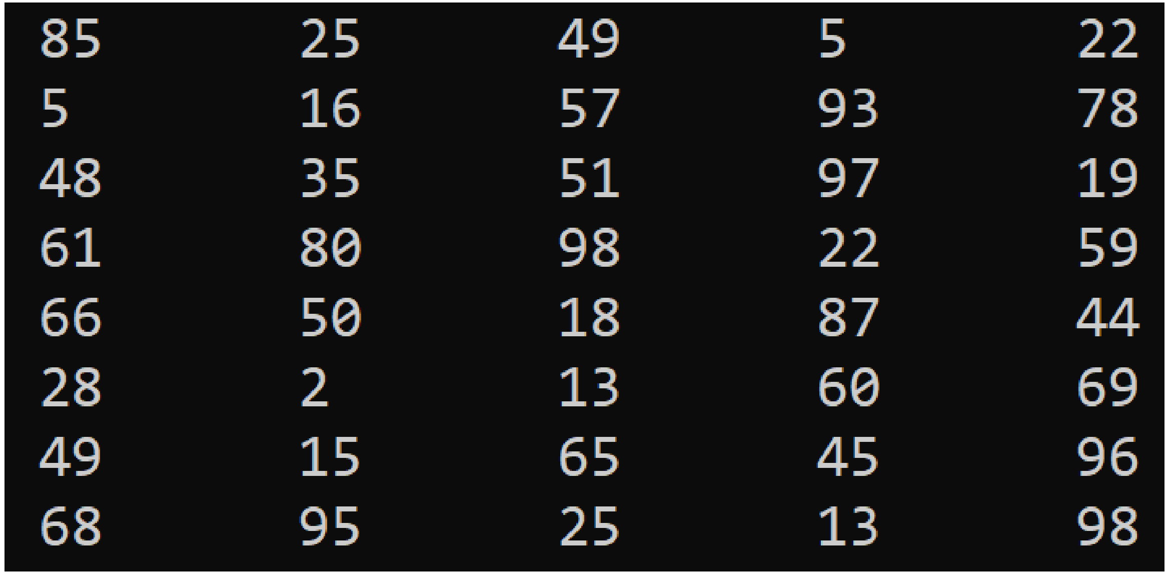


```
// 3-state
int rowSize = 8, columnSize = 5;

int ** matrix;

matrix = new int *[rowSize];

for(int i = 0; i<rowSize; i++){
    matrix[i] = new int[columnSize];
}
```

Example



```
int rowSize = 8, columnSize = 5;
int ** matrix;

matrix = new int *[rowSize];
for(int i = 0; i<rowSize; i++){
    matrix[i] = new int[columnSize];
}

for(int i = 0; i<rowSize; i++){
    for(int j=0; j<columnSize; j++){
        matrix[i][j] = rand()%101; // 0..100
    }
}

for(int i = 0; i<rowSize; i++){
    for(int j=0; j<columnSize; j++){
        cout<<matrix[i][j]<<"\t";
    }
    cout<<endl;
}
```

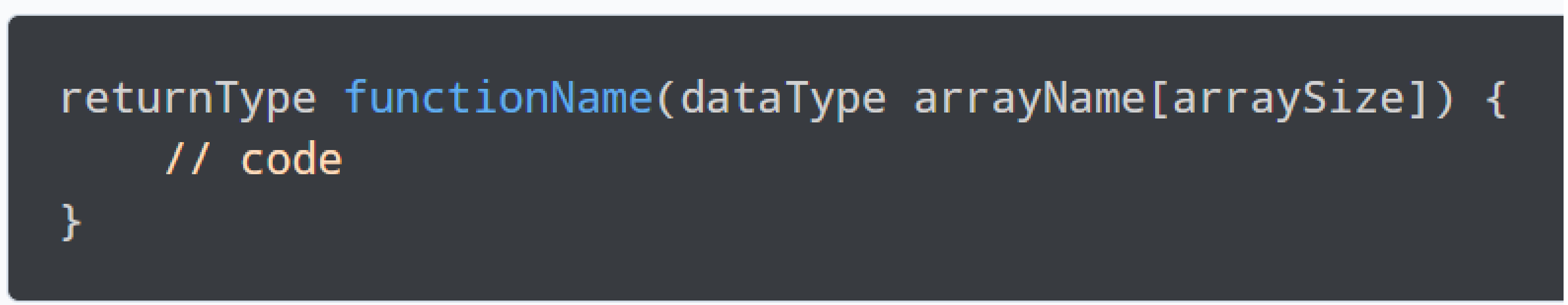


```
85      25      49      5       22
5       16      57      93      78
48      35      51      97      19
61      80      98      22      59
66      50      18      87      44
28      2       13      60      69
49      15      65      45      96
68      95      25      13      98
```

PDP
IT-ACADEMY

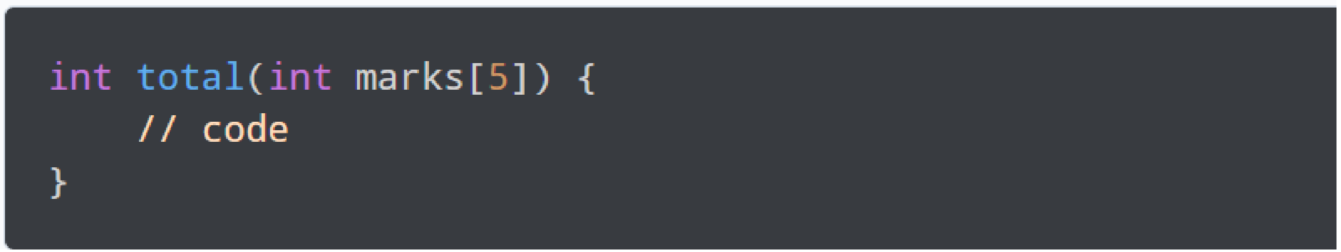
Massiv funksiya parametri sifatida

Syntax

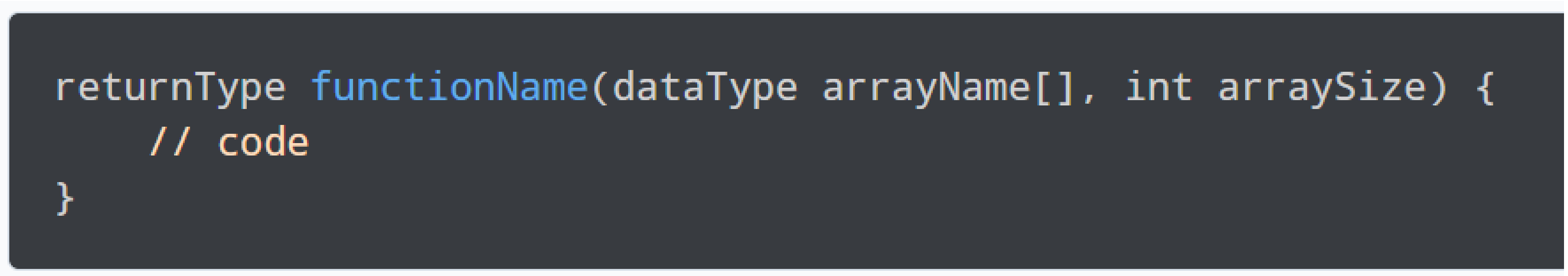


```
returnType functionName(dataType arrayName[arraySize]) {
    // code
}
```

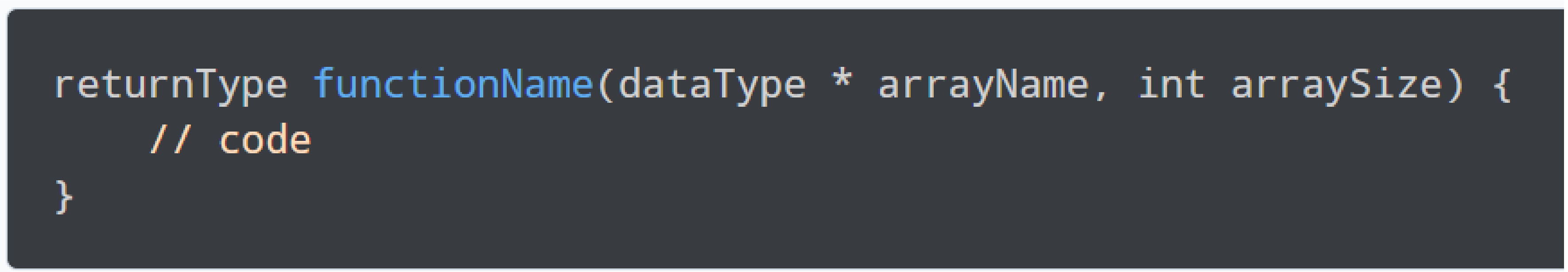
Example



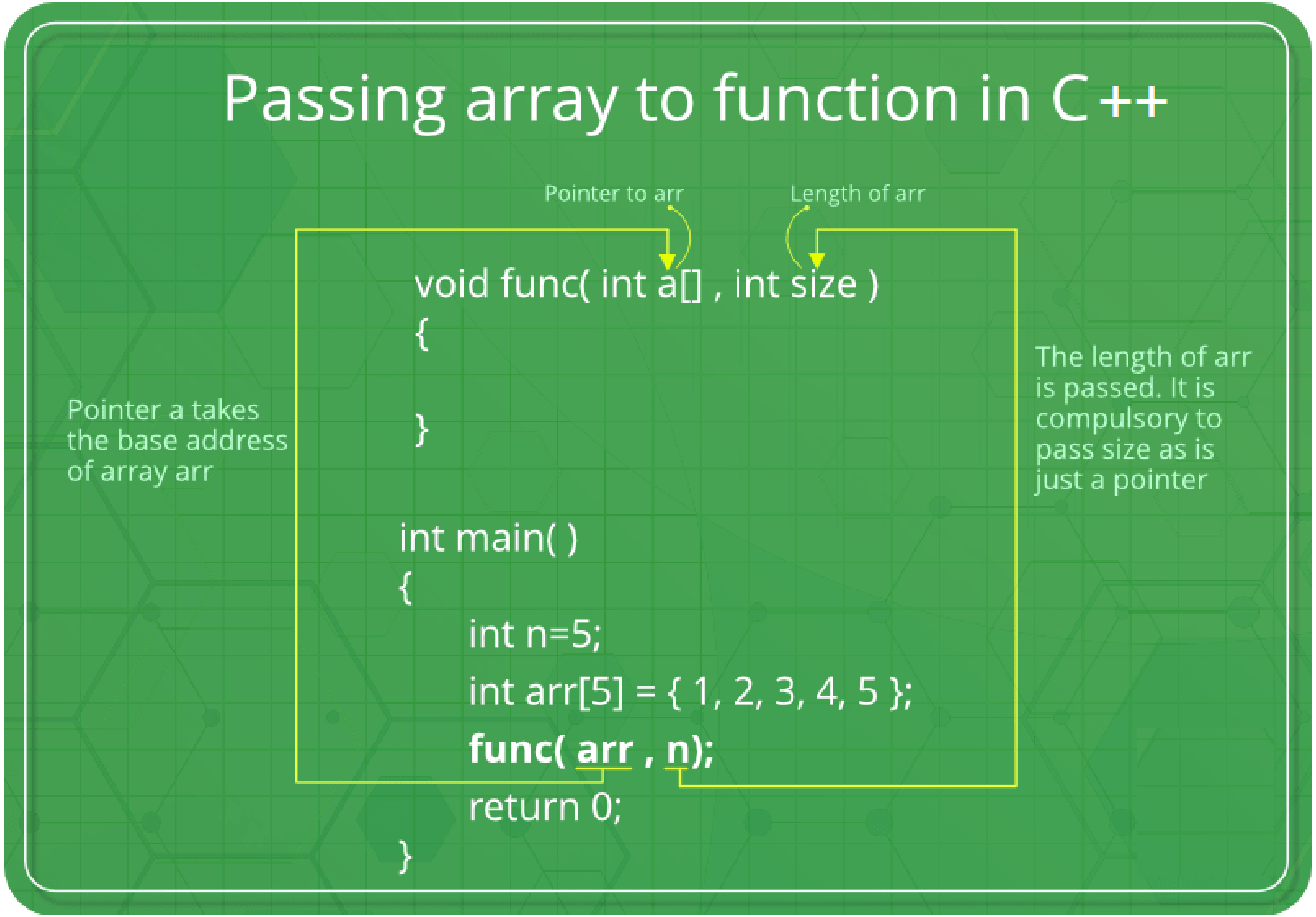
```
int total(int marks[5]) {
    // code
}
```



```
returnType functionName(dataType arrayName[], int arraySize) {
    // code
}
```



```
returnType functionName(dataType * arrayName, int arraySize) {
    // code
}
```



Passing array to function in C++

Pointer to arr

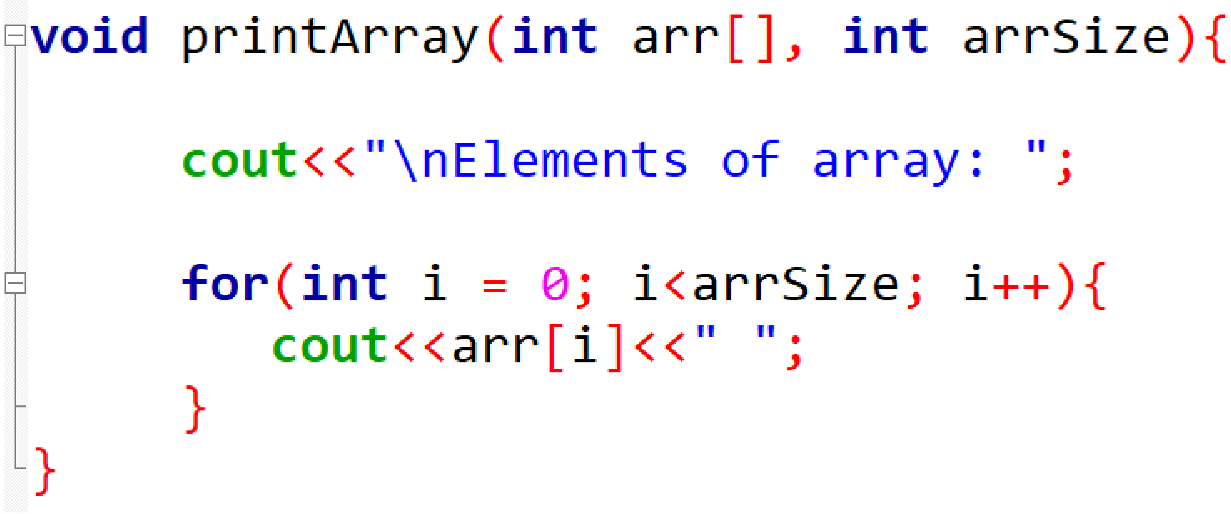
Length of arr

```
void func( int a[] , int size )
{

}

int main( )
{
      int n=5;
      int arr[5] = { 1, 2, 3, 4, 5 };
      func( arr , n);
      return 0;
}
```

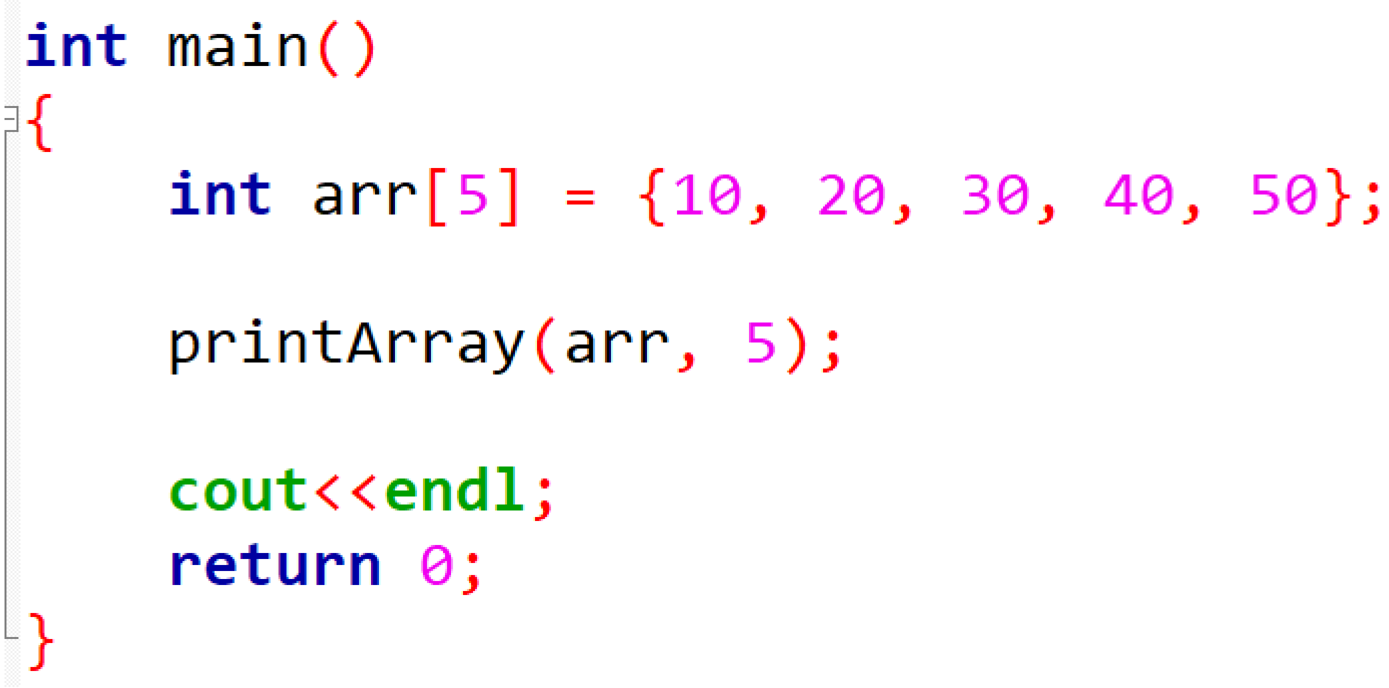
Pointer a takes the base address of array arr

The length of arr is passed. It is compulsory to pass size as is just a pointer

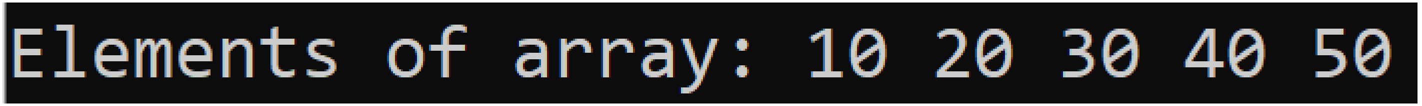
Example



```
void printArray(int arr[], int arrSize){

     cout<<"\nElements of array: ";

     for(int i = 0; i<arrSize; i++){
        cout<<arr[i]<<" ";
     }
}
```



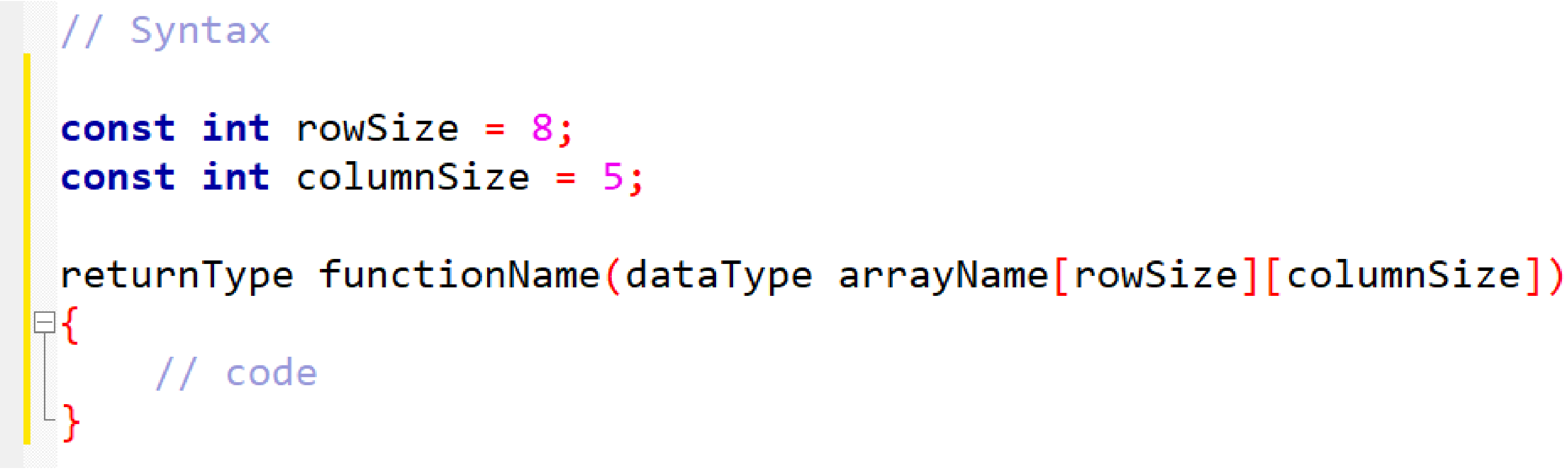
```
int main()
{
    int arr[5] = {10, 20, 30, 40, 50};

    printArray(arr, 5);

    cout<<endl;
    return 0;
}
```



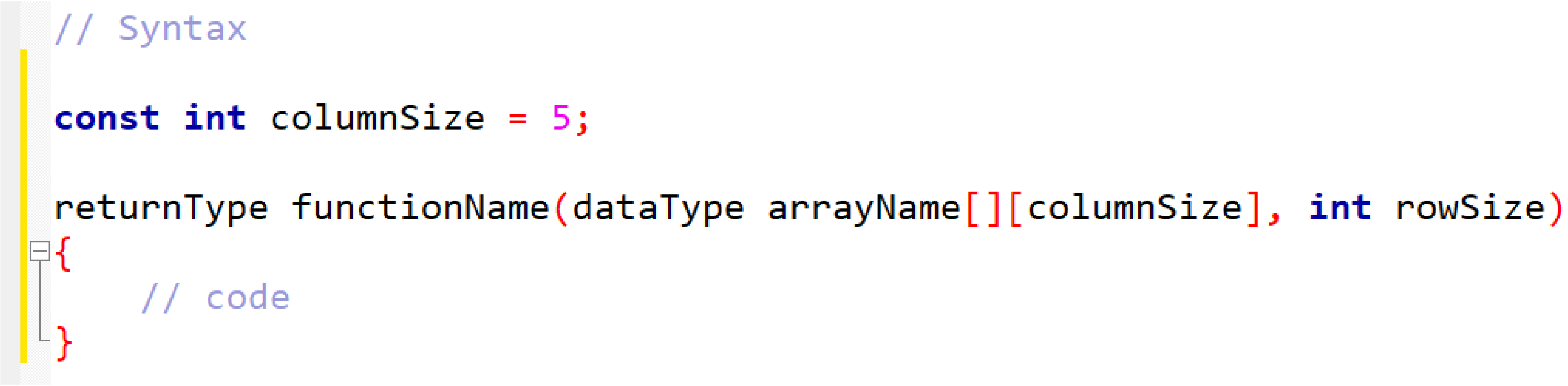
```
Elements of array: 10 20 30 40 50
```

PDP
IT-ACADEMY

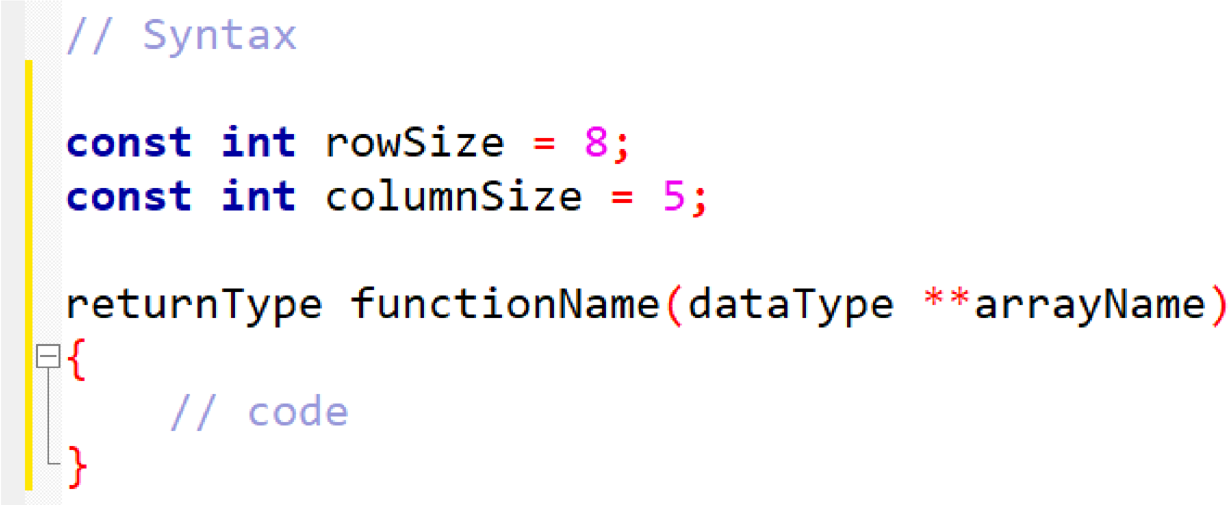
2D multidimensional array



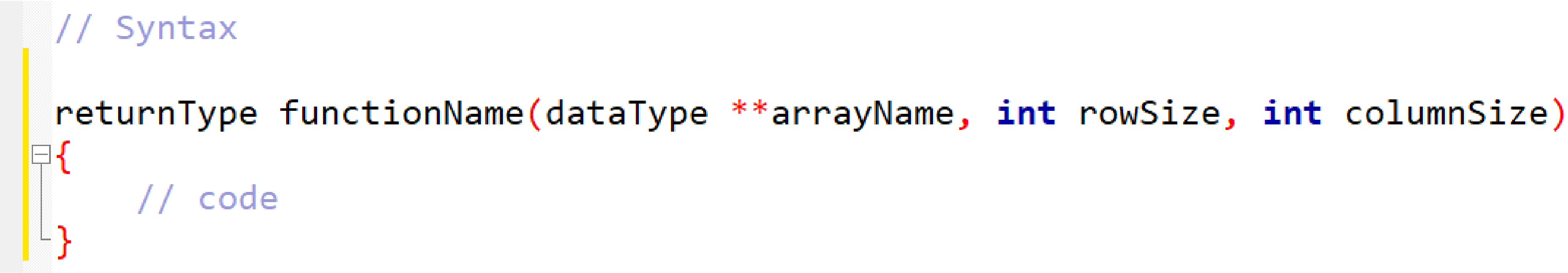
```
// Syntax

const int rowSize = 8;
const int columnSize = 5;

returnType functionName(dataType arrayName[rowSize][columnSize])
{
    // code
}
```



```
// Syntax

const int columnSize = 5;

returnType functionName(dataType arrayName[][columnSize], int rowSize)
{
    // code
}
```

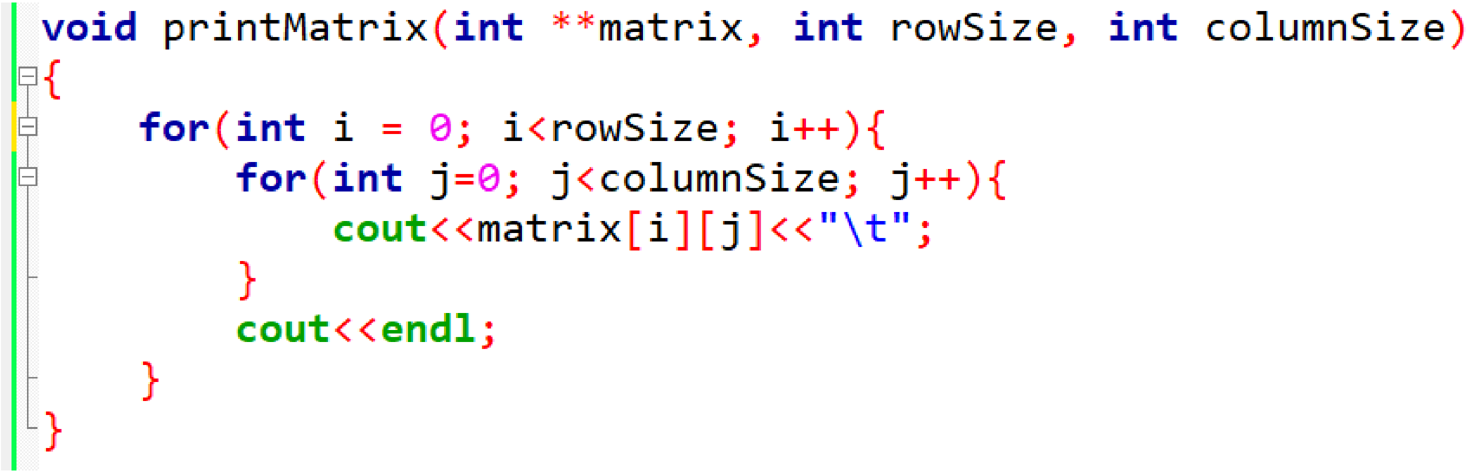


```
// Syntax

const int rowSize = 8;
const int columnSize = 5;

returnType functionName(dataType **arrayName)
{
    // code
}
```

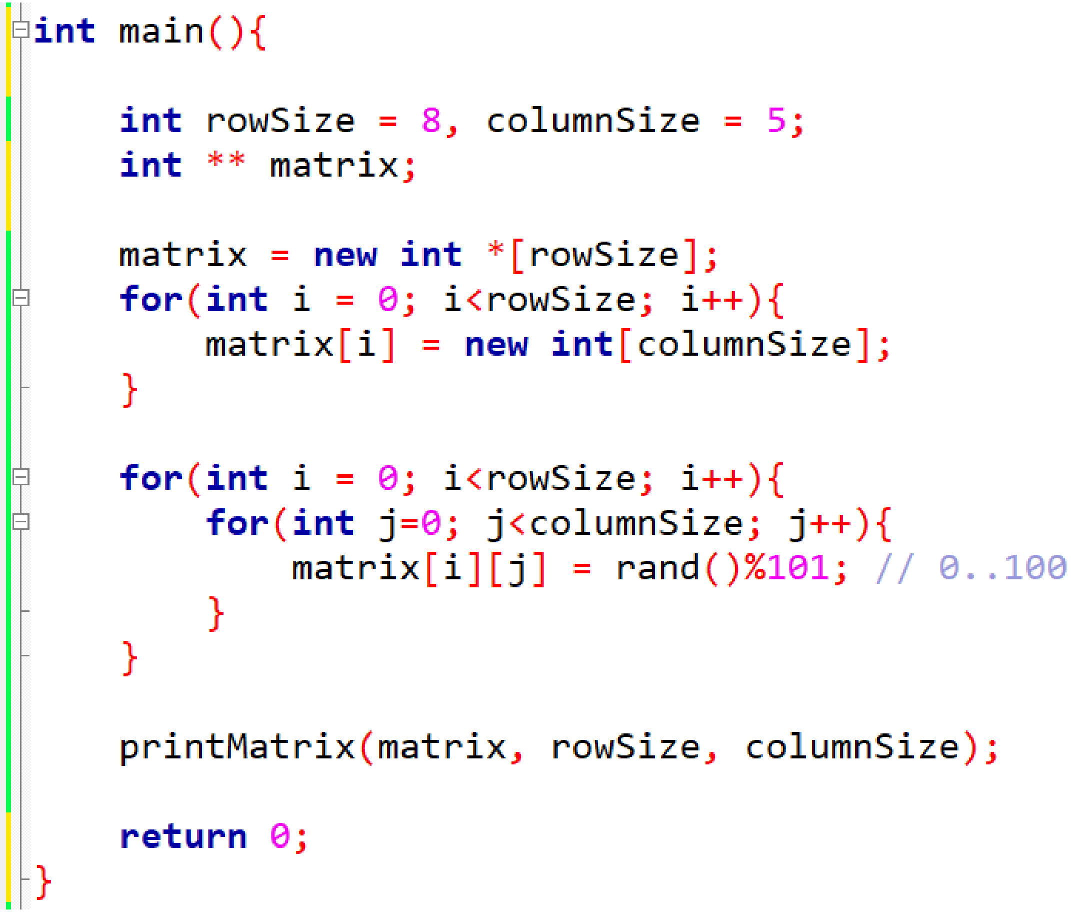


```
// Syntax

returnType functionName(dataType **arrayName, int rowSize, int columnSize)
{
    // code
}
```

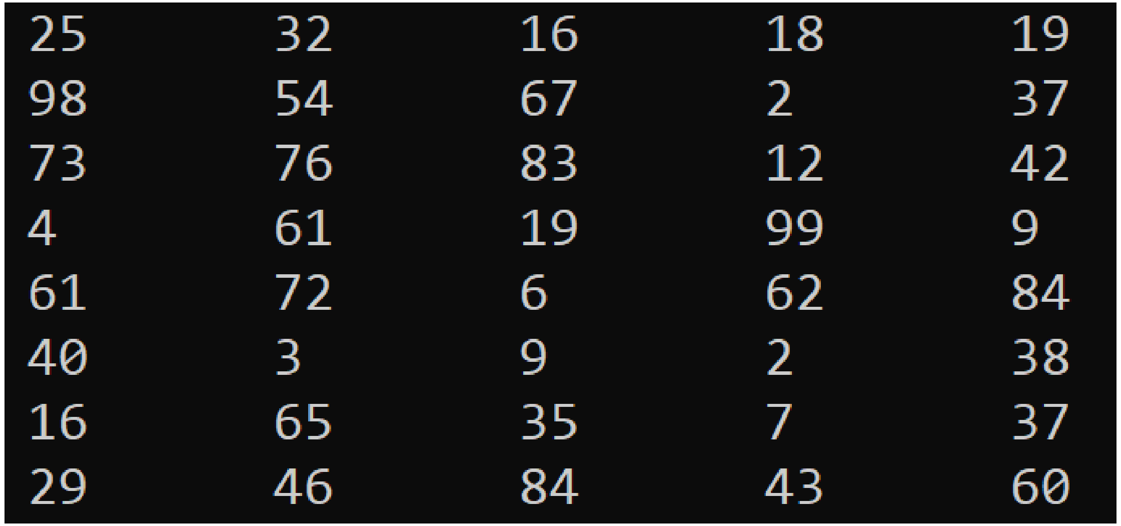
Example



```cpp
void printMatrix(int **matrix, int rowSize, int columnSize)
{
    for(int i = 0; i<rowSize; i++){
        for(int j=0; j<columnSize; j++){
            cout<<matrix[i][j]<<"\t";
        }
        cout<<endl;
    }
}
```



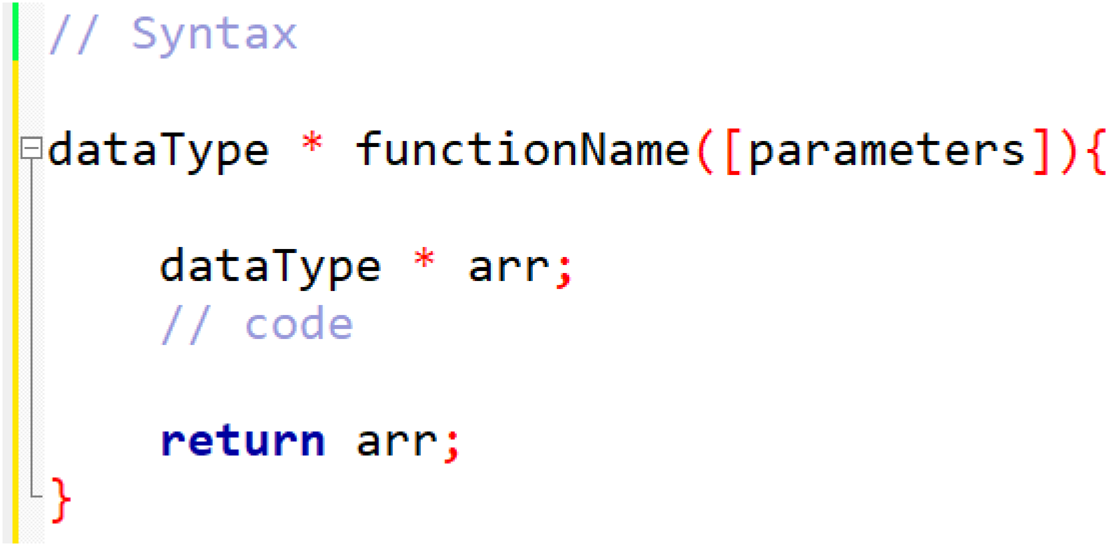
```
int main(){

    int rowSize = 8, columnSize = 5;
    int ** matrix;

    matrix = new int *[rowSize];
    for(int i = 0; i<rowSize; i++){
        matrix[i] = new int[columnSize];
    }

    for(int i = 0; i<rowSize; i++){
        for(int j=0; j<columnSize; j++){
            matrix[i][j] = rand()%101; // 0..100
        }
    }

    printMatrix(matrix, rowSize, columnSize);

    return 0;
}
```



```
25      32      16      18      19
98      54      67      2       37
73      76      83      12      42
4       61      19      99      9
61      72      6       62      84
40      3       9       2       38
16      65      35      7       37
29      46      84      43      60
```

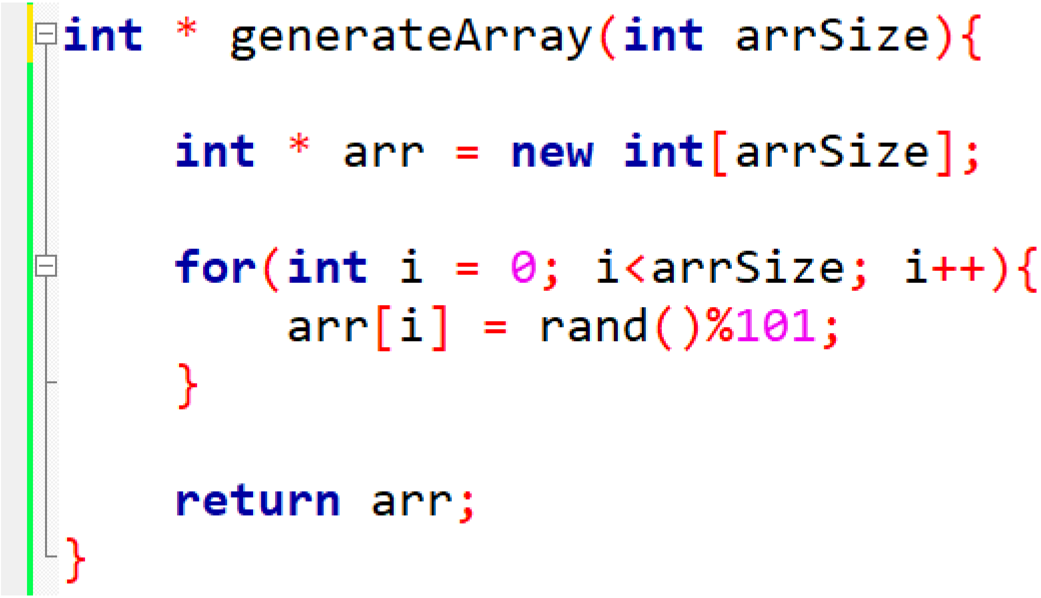
PDP
IT-ACADEMY

Massiv funksiyadan qaytuvchi qiymat sifatida

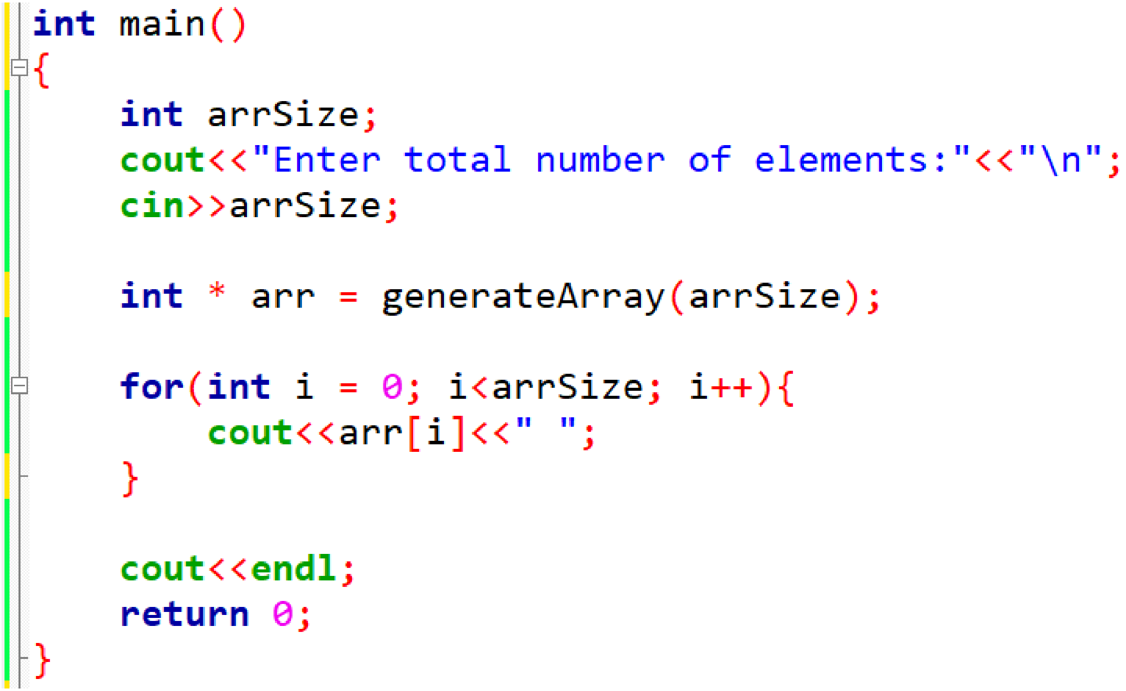


```
// Syntax

dataType * functionName([parameters]){

    dataType * arr;
    // code

    return arr;
}
```

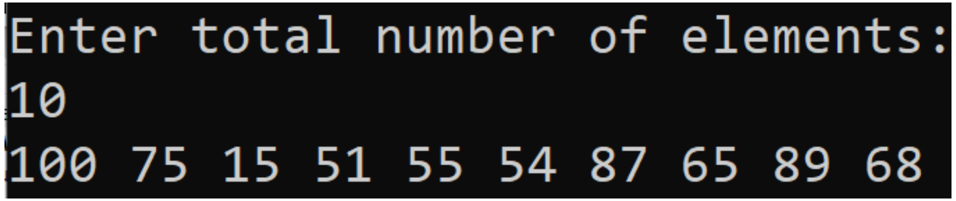
Example



```
int * generateArray(int arrSize){

    int * arr = new int[arrSize];

    for(int i = 0; i<arrSize; i++){
        arr[i] = rand()%101;
    }

    return arr;
}
```

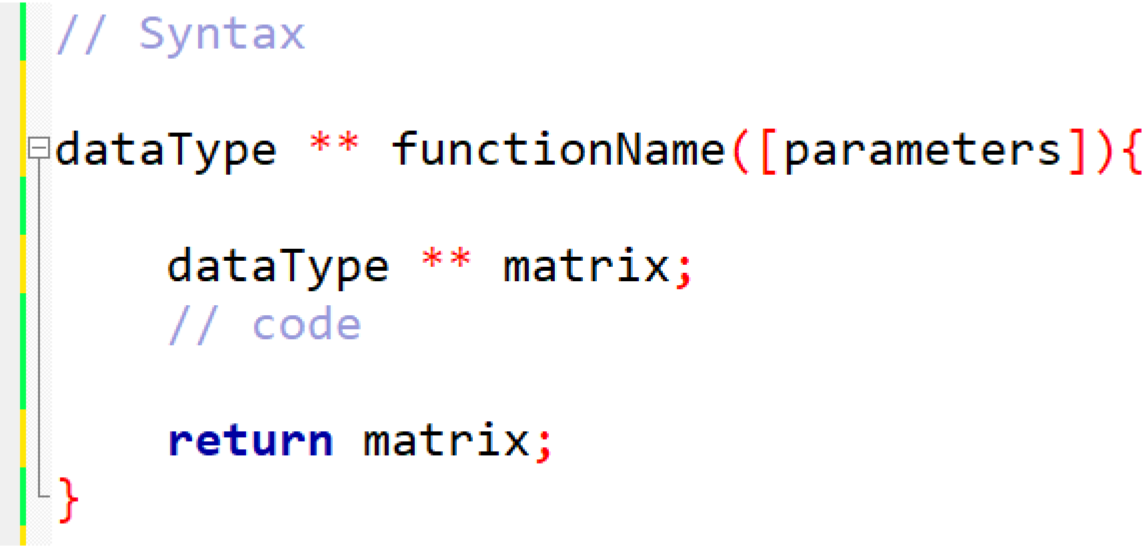


```
int main()
{
    int arrSize;
    cout<<"Enter total number of elements:"<<"\n";
    cin>>arrSize;

    int * arr = generateArray(arrSize);

    for(int i = 0; i<arrSize; i++){
        cout<<arr[i]<<" ";
    }

    cout<<endl;
    return 0;
}
```

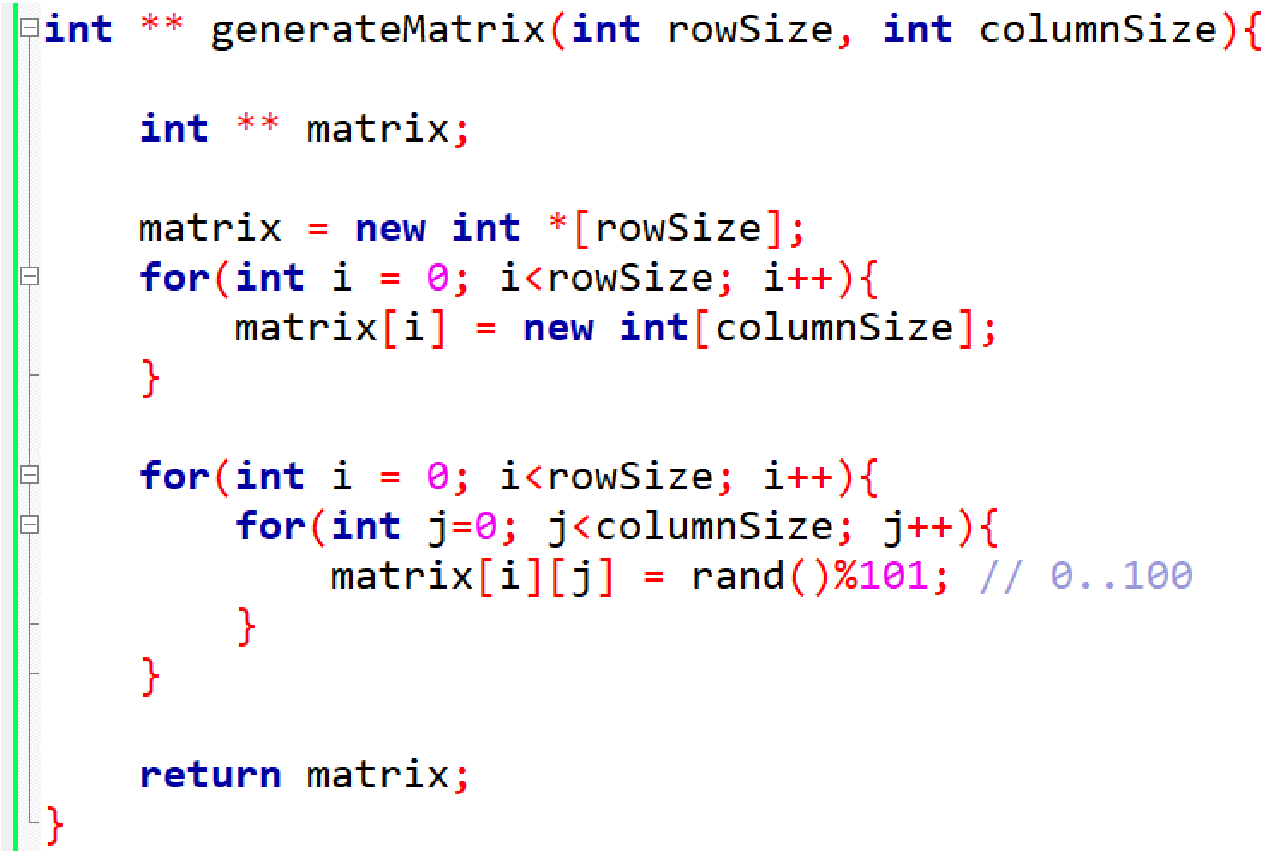


```
Enter total number of elements:
10
100 75 15 51 55 54 87 65 89 68
```

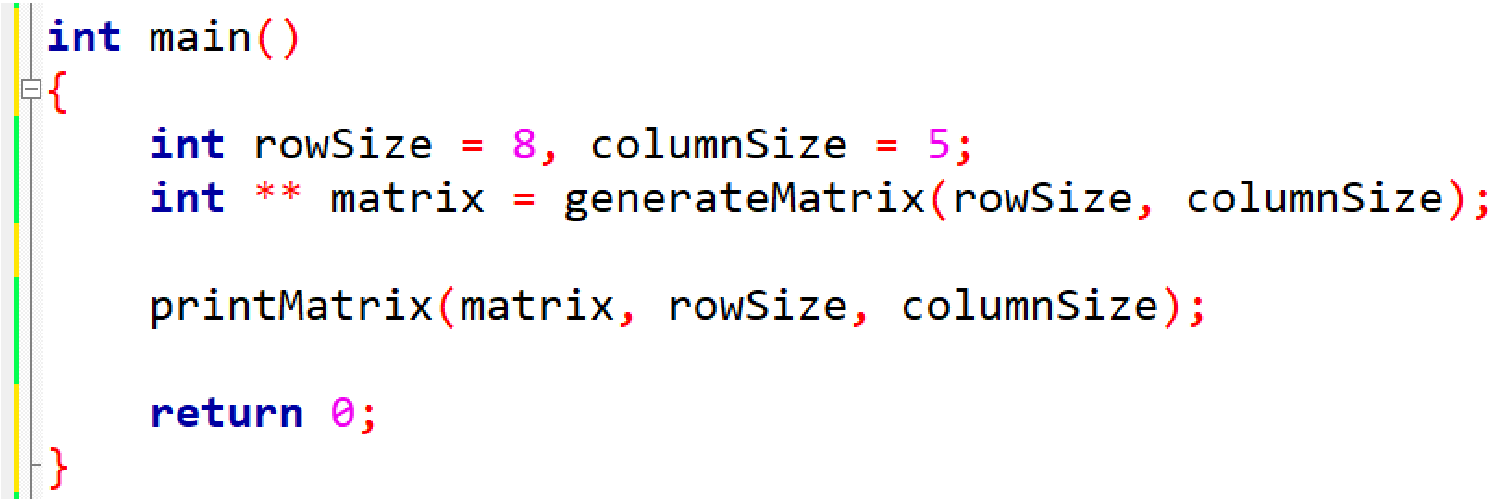
2D multidimensional array



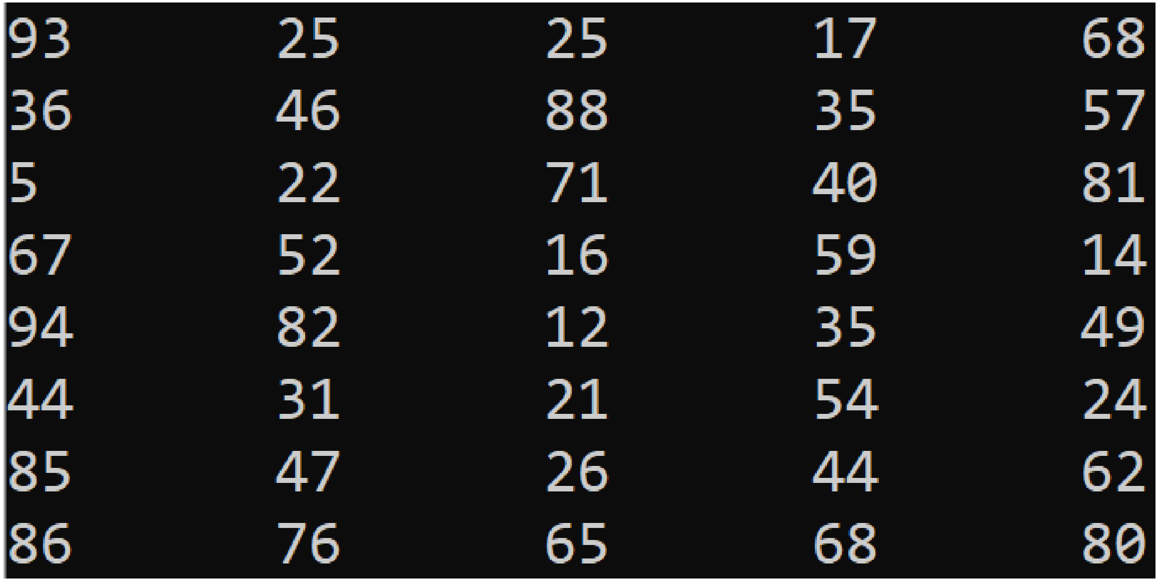
```
// Syntax

dataType ** functionName([parameters]){

    dataType ** matrix;
    // code

    return matrix;
}
```



```
int ** generateMatrix(int rowSize, int columnSize){

    int ** matrix;

    matrix = new int *[rowSize];
    for(int i = 0; i<rowSize; i++){
        matrix[i] = new int[columnSize];
    }

    for(int i = 0; i<rowSize; i++){
        for(int j=0; j<columnSize; j++){
            matrix[i][j] = rand()%101; // 0..100
        }
    }

    return matrix;
}
```



```
int main()
{
    int rowSize = 8, columnSize = 5;
    int ** matrix = generateMatrix(rowSize, columnSize);

    printMatrix(matrix, rowSize, columnSize);

    return 0;
}
```



```
93              25              25              17              68
36              46              88              35              57
5               22              71              40              81
67              52              16              59              14
94              82              12              35              49
44              31              21              54              24
85              47              26              44              62
86              76              65              68              80
```

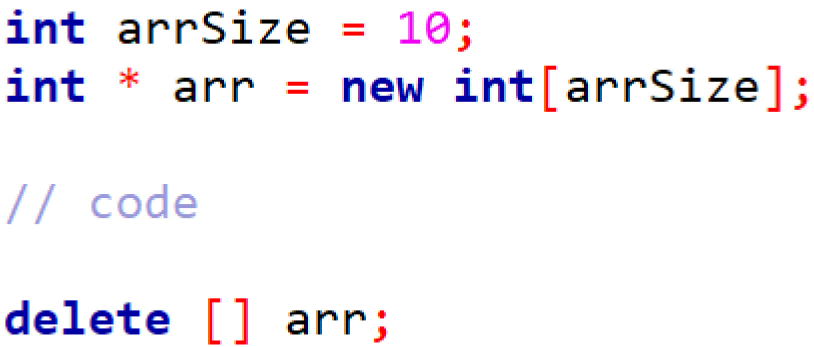
PDP
IT-ACADEMY

Dinamik massiv uchun ajratilgan xotirani bo‘shatish

PDP
IT-ACADEMY

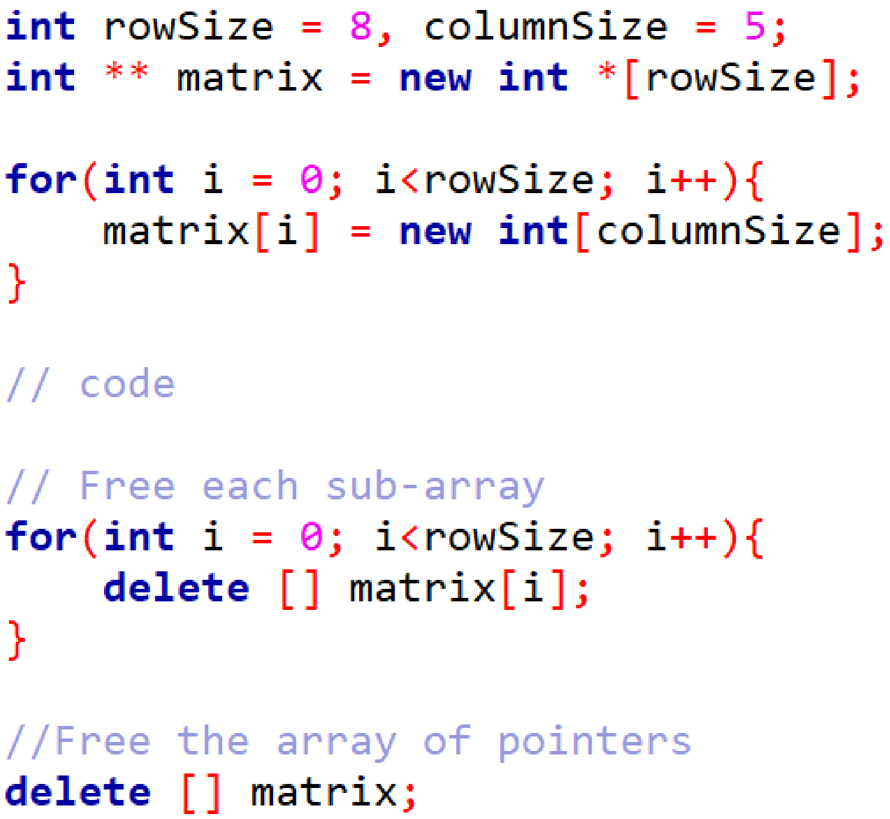
Dinamik massivlar uchun ajratilgan xotirani bo’shatish uchun **delete** kalit so’zidan foydalaniladi.

Example



```
int arrSize = 10;
int * arr = new int[arrSize];

// code

delete [] arr;
```

Example

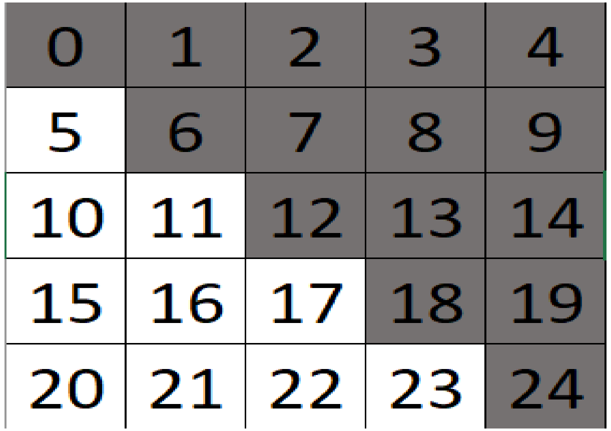


```cpp
int rowSize = 8, columnSize = 5;
int ** matrix = new int *[rowSize];

for(int i = 0; i<rowSize; i++){
    matrix[i] = new int[columnSize];
}

// code

// Free each sub-array
for(int i = 0; i<rowSize; i++){
    delete [] matrix[i];
}

//Free the array of pointers
delete [] matrix;
```

Amaliy mashqlar

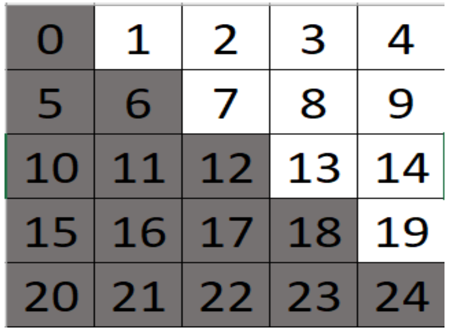
Butun sonlardan iborat N ta elementli massiv berilgan. Ushbu massivni parametr sifatida qabul qilib, undagi juft elementlar sonini natija sifatida qaytaruvchi funksiya yozing.

Haqiqiy sonlardan iborat N ta elementli massiv berilgan. Ushbu massivni parametr sifatida qabul qilib, uning eng katta elementini natija sifatida qaytaruvchi funksiya yozing.

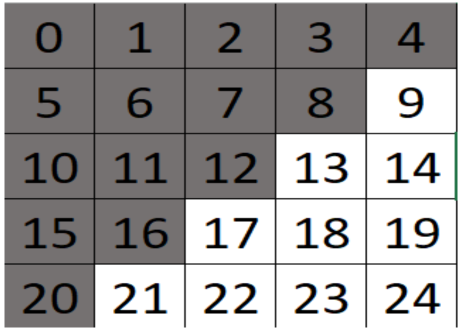


0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24

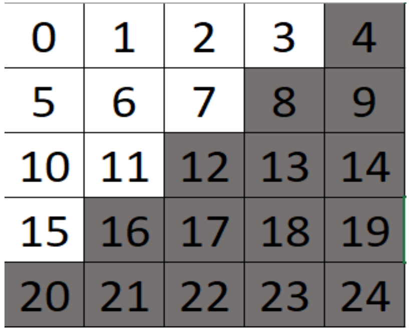
N x N o’lchamdagi butun sonli massiv berilgan. Ushbu massivni parametr sifatida qabul qilib, undagi ajratib ko’rsatilgan sohadagi elementlar yig’indisini natija sifatida qaytaruvchi funksiya yozing.



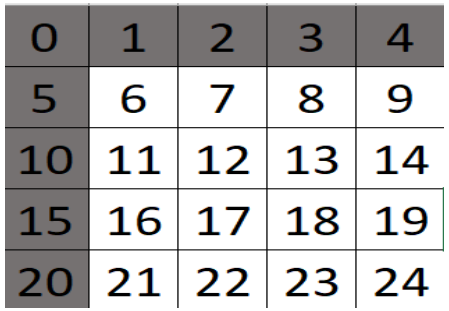
0	1	2	3	4
5	**6**	7	8	9
10	**11**	**12**	13	14
15	**16**	**17**	**18**	19
20	**21**	**22**	**23**	**24**



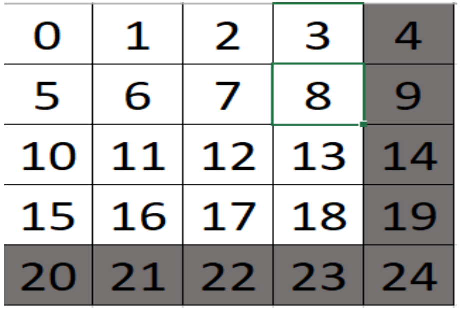
0	**1**	**2**	**3**	**4**
5	**6**	**7**	**8**	9
10	**11**	**12**	13	14
15	**16**	17	18	19
20	21	22	23	24



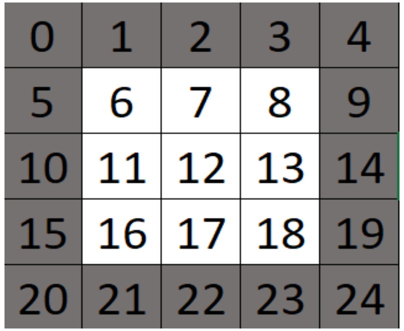
0	1	2	3	**4**
5	6	7	**8**	**9**
10	11	**12**	**13**	**14**
15	**16**	**17**	**18**	**19**
20	**21**	**22**	**23**	**24**



0	**1**	**2**	**3**	**4**
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24



0	1	2	3	**4**
5	6	7	8	**9**
10	11	12	13	**14**
15	16	17	18	**19**
20	**21**	**22**	**23**	**24**



0	**1**	**2**	**3**	**4**
5	6	7	8	**9**
10	11	12	13	**14**
15	16	17	18	**19**
20	**21**	**22**	**23**	**24**

Uzunligi 7 ga teng bo’lgan **int** toifasidagi qiymatlarni saqlovchi array berilgan. Barcha elementlari yuqoridagi arrayning minimum elementiga teng bo’lgan shunday uzunlikdagi arrayni natija sifatida qaytaruvchi funksiya tuzing. Masalan:

[10, 12, **3**, 24, 55, 41, 31] → [3, 3, 3, 3, 3, 3, 3]

N ta elementli array berilgan. Xuddi shunday faqat elementlari teskari tartibda joylashgan yangi arrayni natija sifatida qaytaruvchi funksiya tuzing.

Masalan: [-1, 22, 37, 4, 15] → [15, 4, 37, 22, -1].

Natural N soni berilgan. Uni parametr sifatida qabul qilib, elementlari 1 dan N ning kvadratigacha bo’lgan sonlardan tashkil topgan NxN o‘lchamdagi massivni natija sifatida qaytaruvchi funksiya tuzing.

PDP
IT-ACADEMY

E`tiboringiz uchun rahmat!