



Computer Programming (a) - E1123

(Fall 2021-2022)

Lecture 11



Multidimensional Arrays

INSTRUCTOR

DR / AYMAN SOLIMAN

➤ Contents

- 1) Introduction.
- 2) Multidimensional Arrays.
- 3) Declaration and assignment of a 3D&4D arrays.
- 4) Examples.

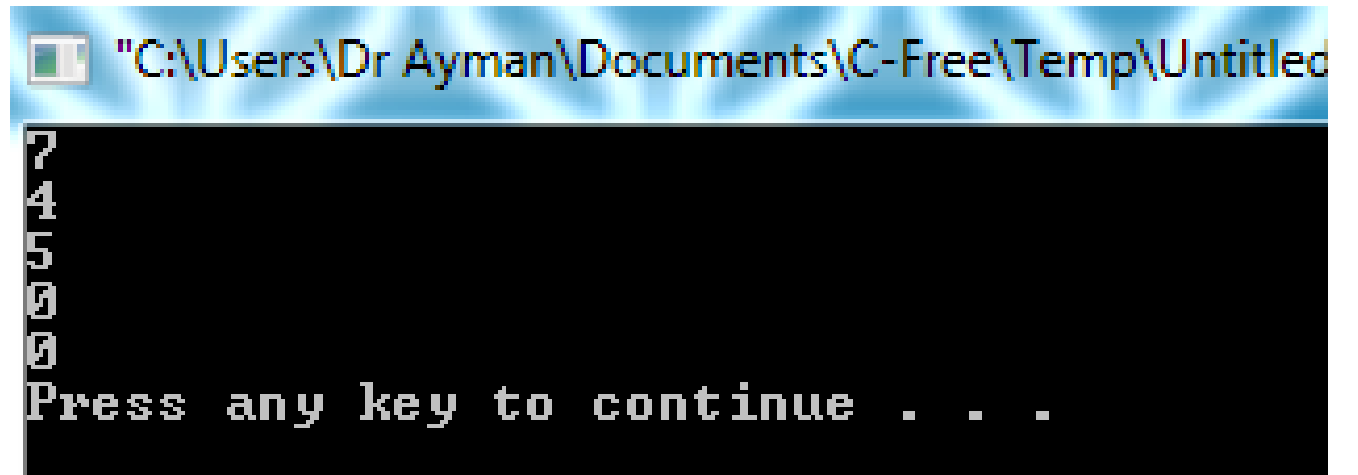


➤ Introduction

```
#include <iostream.h>

int main()
{
    int array[5]={ 7, 4, 5 }; // only initialize first 3 elements

    cout << array[0] << '\n';
    cout << array[1] << '\n';
    cout << array[2] << '\n';
    cout << array[3] << '\n';
    cout << array[4] << '\n';
    return 0;
}
```



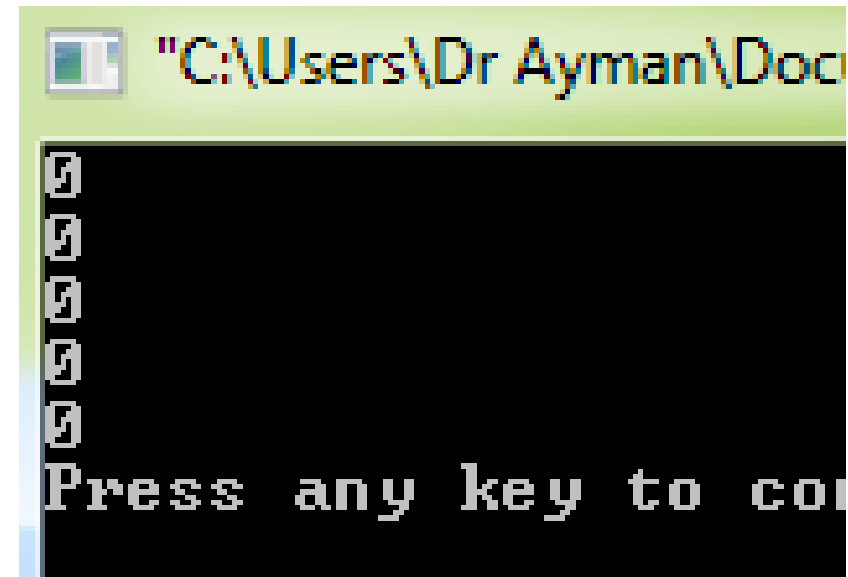
```
"C:\Users\Dr Ayman\Documents\C-Free\Temp\Untitled
7
4
5
0
0
Press any key to continue . . .
```

➤ Example 1

```
#include <iostream.h>

int main()
{
    int array[5]={ }; // only initialize all elements to zero

    cout << array[0] << '\n';
    cout << array[1] << '\n';
    cout << array[2] << '\n';
    cout << array[3] << '\n';
    cout << array[4] << '\n';
    return 0;
}
```



```
"C:\Users\Dr Ayman\Doc
0
0
0
0
0
Press any key to continue
```

➤ Multidimensional Arrays

- ❑ Arrays could be **more than** one dimension.

```
int array[3][5]; // declaration of a 3*5 element array
```

```
[0] [0]  [0] [1]  [0] [2]  [0] [3]  [0] [4]  // row 0  
[1] [0]  [1] [1]  [1] [2]  [1] [3]  [1] [4]  // row 1  
[2] [0]  [2] [1]  [2] [2]  [2] [3]  [2] [4]  // row 2
```

- ❑ To access the elements of a two-dimensional array, simply use two subscripts:

```
array[0][0] = 3;
```

```
array[0][1] = 30;
```

```
array[2][3] = 7;
```

➤ Multidimensional Arrays (cont.)

❑ Example 2

```
#include <iostream.h>
```

```
int main()
```

```
{ int array[3][5] =
```

```
{ { 1, 2, 3, 4, 5 }, // row 0
```

```
{ 6, 7, 8, 9, 10 }, // row 1
```

```
{ 11, 12, 13, 14, 15 } // row 2
```

```
};
```

```
return 0;}
```

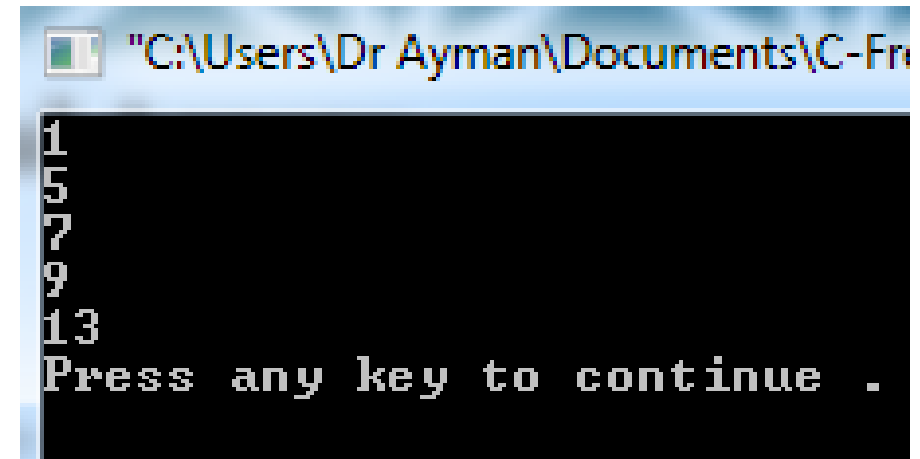
```
cout << array[0][0] << '\n';
```

```
cout << array[0][4] << '\n';
```

```
cout << array[1][1] << '\n';
```

```
cout << array[1][3] << '\n';
```

```
cout << array[2][2] << '\n';
```



```
"C:\Users\Dr Ayman\Documents\C-Fr  
1  
5  
7  
9  
13  
Press any key to continue .
```

➤ Multidimensional Arrays (cont.)

- ❑ Initializing two-dimensional arrays

//Acceptable

```
int array[3][5] =  
{  
  { 1, 2, 3, 4, 5 },      // row 0  
  { 6, 7, 8, 9, 10 },    // row 1  
  { 11, 12, 13, 14, 15 } // row 2  
};
```

//Acceptable

```
int array[3][5] =  
{  
  { 1, 2 },              // row 0 = 1, 2, 0, 0, 0  
  { 6, 7, 8 },          // row 1 = 6, 7, 8, 0, 0  
  { 11, 12, 13, 14 }    // row 2 = 11, 12, 13, 14, 0  
};
```

```
int array[3][5] = { };
```

//Acceptable

➤ Multidimensional Arrays (cont.)

- ❑ Initializing two-dimensional arrays

```
int array[ ][5] =  
{  
  { 1, 2, 3, 4, 5 },    // row 0  
  { 6, 7, 8, 9, 10 },  // row 1  
  { 11, 12, 13, 14, 15 } // row 2  
};
```

//Acceptable

//Compiler Error

```
int array[ ][ ] =  
{  
  { 1, 2, 3, 4, 5 },    // row 0  
  { 6, 7, 8, 9, 10 },  // row 1  
  { 11, 12, 13, 14, 15 } // row 2  
};
```

//Compiler Error

```
int array[3][ ] = {{ 1, 2, 3, 4, 5 },    // row 0  
                  { 6, 7, 8, 9, 10 },    // row 1  
                  { 11, 12, 13, 14, 15 } // row 2  
};
```


➤ Example 3

```
#include <iostream.h>
```

```
int main()
```

```
{
```

```
int array[3][5] =
```

```
{
```

```
1, 2, 3,
```

```
6, 7, 8, 9, 10 ,
```

```
11, 12, 13, 14, 15
```

```
};
```

1	2	3	6	7
8	9	10	11	12
13	14	15	0	0

//Acceptable

```
cout << array[0][0] << '\n';
```

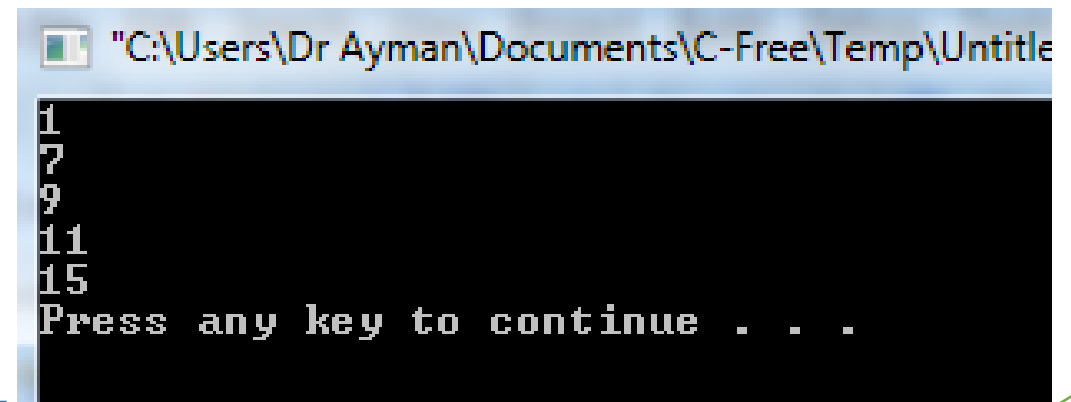
```
cout << array[0][4] << '\n';
```

```
cout << array[1][1] << '\n';
```

```
cout << array[1][3] << '\n';
```

```
cout << array[2][2] << '\n';
```

```
return 0;}
```



```
"C:\Users\Dr Ayman\Documents\C-Free\Temp\Untitled  
1  
7  
9  
11  
15  
Press any key to continue . . .
```

➤ Declaration and assignment of a 3D array

```
int array[2][2][3];
```

```
array[0][0][0] = 2;
```

```
array[0][0][1] = 3; .....etc
```

➤ Initialize all elements to 0

```
int array[2][3][7] = {};
```

➤ Declaration and assignment of a 4D array

```
int array[2][2][3][7];
```

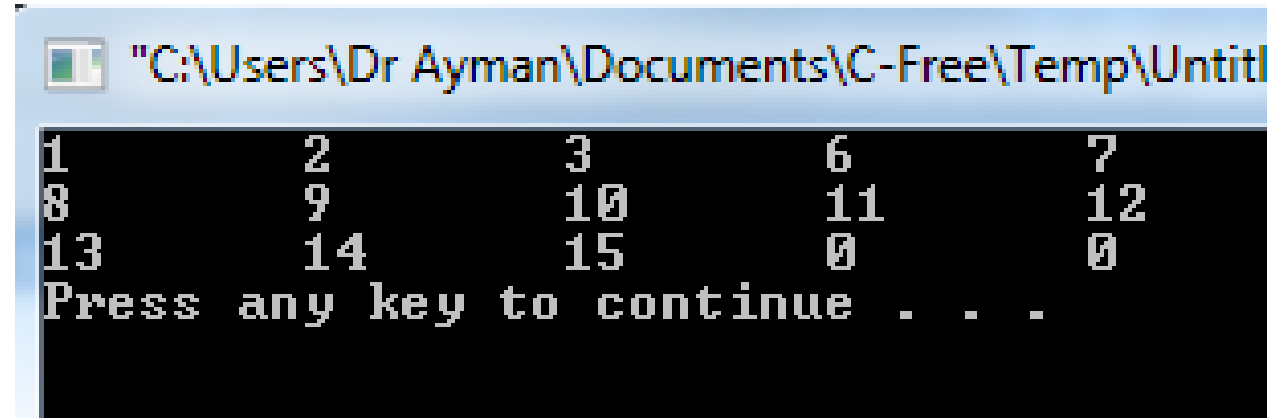
➤ Printing all elements of array

```
#include <iostream.h>

int main()
{
int array[3][5] =
{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15};

for ( int x=0;x<3;x++)
{for (int y=0; y<5;y++)
cout<<array[x][y]<<"\t";
cout<<endl;}

return 0;}
```



The screenshot shows a Windows command prompt window with the following output:

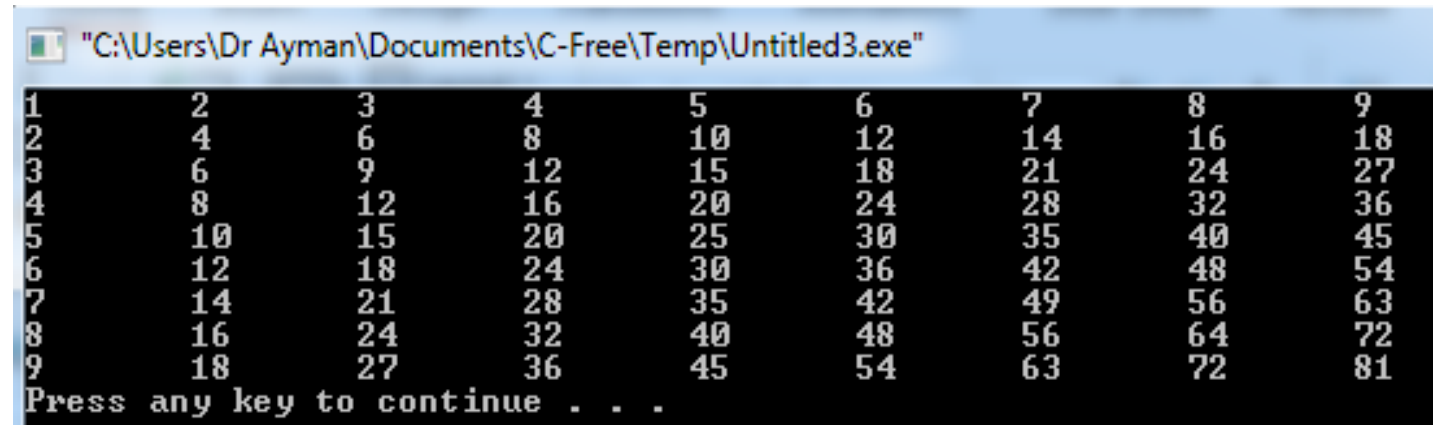
```
"C:\Users\Dr Ayman\Documents\C-Free\Temp\Un titl
1      2      3      6      7
8      9      10     11     12
13     14     15     0      0
Press any key to continue . . .
```

➤ Example 4 (multiplication table)

```
#include <iostream.h>

int main()
{
    int multiplication[9][9]={};
    for (int x=0;x<9;x++)
        for (int y=0; y<9;y++)
            multiplication[x][y]=(x+1)*(y+1);
    // print the table:-
    for (int x=0;x<9;x++)
        {for (int y=0; y<9;y++)
            cout<<multiplication[x][y]<<'t';
        cout<<endl;}
    return 0;}

```



```
"C:\Users\Dr Ayman\Documents\C-Free\Temp\Untitled3.exe"
1 2 3 4 5 6 7 8 9
2 4 6 8 10 12 14 16 18
3 6 9 12 15 18 21 24 27
4 8 12 16 20 24 28 32 36
5 10 15 20 25 30 35 40 45
6 12 18 24 30 36 42 48 54
7 14 21 28 35 42 49 56 63
8 16 24 32 40 48 56 64 72
9 18 27 36 45 54 63 72 81
Press any key to continue . . .

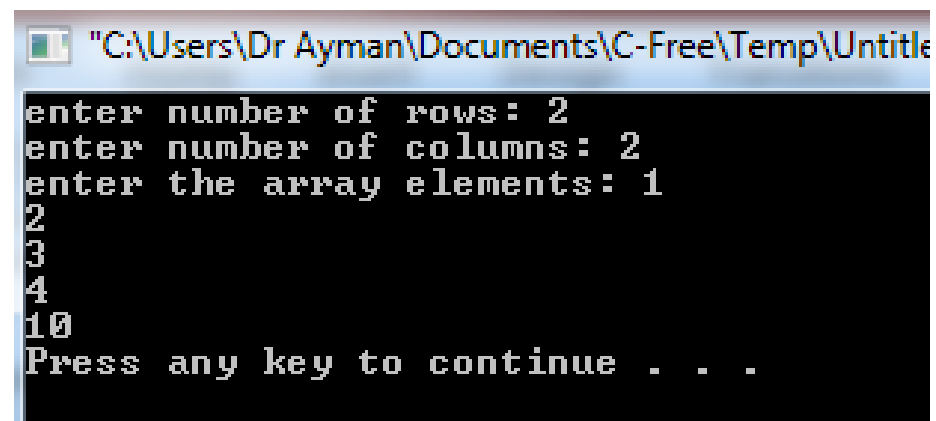
```

➤ Example 5 Find the summation of all array elements

```
#include <iostream.h>

int main()
{int i,j,sum=0;
cout<<"enter number of rows: ";
cin>>i;
cout<<"enter number of columns: ";
cin>>j;
cout<<"enter the array elements: ";
int array[i][j];
    for (int x=0;x<i;x++)
    for (int y=0; y<j;y++)
cin>>array[x][y];
```

```
// find the summation of all elements
    for (int x=0;x<i;x++)
    for (int y=0; y<j;y++)
    sum=sum+array[x][y];
// print the summation
cout<<sum<<endl;
return 0;}
```



```
"C:\Users\Dr Ayman\Documents\C-Free\Temp\Untitled
enter number of rows: 2
enter number of columns: 2
enter the array elements: 1
2
3
4
10
Press any key to continue . . .
```

Thank
you

