Passing Array to a Function in C++ Programming

[Arrays](https://www.programiz.com/cpp-programming/arrays) can be passed to a [function](https://www.programiz.com/cpp-programming/function) as an argument. Consider this example to pass one-dimensional array to a function:

## Example 1: Passing One-dimensional Array to a Function

**C++ Program to display marks of 5 students by passing one-dimensional array to a function.**

#include <iostream>

using namespace std;

void display(int marks[5]);

int main()

{

int marks[5] = {88, 76, 90, 61, 69};

display(marks);

return 0;

}

void display(int m[5])

{

cout << "Displaying marks: "<< endl;

for (int i = 0; i < 5; ++i)

{

cout << "Student "<< i + 1 <<": "<< m[i] << endl;

}

}

**Output**

Displaying marks:

Student 1: 88

Student 2: 76

Student 3: 90

Student 4: 61

Student 5: 69

### Example 2: Passing Multidimensional Array to a Function

**C++ Program to display the elements of two dimensional array by passing it to a function.**

#include <iostream>

using namespace std;

void display(int n[3][2]);

int main()

{

int num[3][2] = {

{3, 4},

{9, 5},

{7, 1}

};

display(num);

return 0;

}

void display(int n[3][2])

{

cout << "Displaying Values: " << endl;

for(int i = 0; i < 3; ++i)

{

for(int j = 0; j < 2; ++j)

{

cout << n[i][j] << " ";

}

}

}

**Output**

Displaying Values:

3 4 9 5 7 1

In the above program, the multi-dimensional array num is passed to the function display().

Inside, display() function, the array n (num) is traversed using a nested for loop.

The program uses 2 [for loops](https://www.programiz.com/cpp-programming/for-loop) to iterate over the elements inside a 2-dimensional array. If it were a 3-dimensional array, you should use 3 for loops.

Finally, all elements are printed onto the screen.