

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

(Week 16) Lecture 31 and 32

Objectives: Learning objectives of this lecture are

- Returning array from functions
- String data type.



Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

Lecture # 31 Returning Array from Function

We have discussed about different data types as return type of functions. In this lecture we shall be looking at how do we return array from functions? When talking about returning array from a function we actually means that returning address from a function. Let us look at syntax of returning address from a function.

Syntax

As the name of array is pointer and it holds address of first memory location. When we talk about returning array from function it actually means returning address.

```
int* func(      );
```



* Indicates that it will return address of integer type

Figure 1 Returning address from function

Let us use in a program.

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
int* func1()
{
    return new int[3];
}
void main()
{
    int* arr = func1();
}
```

Problem in main how would we know about size of array?

In main we do not know about size of array. Now how could we access size in main or any other function? A function can only return address or a function can return size of array. There is a solution to the problem. If we pass a variable as reference that stores size of array, in this way we could have size of array and address of array outside function.

```
int* func1(int &size)
{
    size = 3;
    return new int[3];
}
void main()
{
    int asize;
    int* arr = func1(asize);
}
```

Problem solved because now we can have size in *assize* variable declared in main.

Program 1: Declare an array of size 10 of integer type. Take input in array pass this array to function that will return array of even numbers

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

Solution:

```
#include "iostream"
using namespace std;
int* geteven(int num[], int total, int& evensize)
{
    evensize = 0;
    for (int i = 0; i < total; i++)
    {
        if (num[i] % 2 == 0)
            evensize++;
    }
    if (evensize == 0)
        return nullptr;
    int* even = new int[evensize];
    int j = 0;
    for (int i = 0; i < total; i++)
    {
        if (num[i] % 2 == 0)
        {
            even[j++] = num[i];
        }
    }
    return even;
}

void main()
{
    int arr[10];
    for (int i = 0; i < 10; i++)
        cin >> arr[i];
    int* evennumber, totaleven;
    evennumber = geteven(arr, 10, totaleven);
    if (evennumber == nullptr)
        cout << "No even numbers \n";
    else
    {
        for (int i = 0; i < totaleven; i++)
            cout << evennumber[i] << endl;
    }
}
```

Program 2: Make a structure with name country having data members as **name, population and literacy rate**. Declare an array of country of 10 size. Input in array and separate those countries whose literacy rate is below 50. Also display them in sorted order.

Solution:

```
#include "iostream"
using namespace std;
struct Country
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
{
    char name[10];
    int population;
    float liter_rate;
};
Country input()
{
    Country c;
    cin >> c.name;
    cin >> c.population;
    cin >> c.liter_rate;
    return c;
}
void input(Country c[], int total)
{
    for (int i = 0; i < total; i++)
    {
        cin >> c[i].name;
        cin >> c[i].population;
        cin >> c[i].liter_rate;
    }
}
void sort(Country c[], int total)
{
    for (int i = 0; i < total; i++)
    {
        int minind = i;
        for (int j = i + 1; j < total; j++)
        {
            if (c[minind].liter_rate > c[j].liter_rate)
                minind = j;
        }
        Country temp = c[i];
        c[i] = c[minind];
        c[minind] = temp;
    }
}
int countcountries(Country c[], int total)
{
    int count = 0;
    for (int i = 0; i < total; i++)
    {
        if (c[i].liter_rate < 50)
            count++;
    }
    return count;
}
Country* getcountries(Country cdetail[], int total, int& newsize)
{
    newsize = countcountries(cdetail, total);
    if (newsize == 0)
        return nullptr;
}
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
Country* belowlit = new Country[newsize];
int j = 0;
for (int i = 0; i < total; i++)
{
    if (cdetail[i].liter_rate < 50)
        belowlit[j++] = cdetail[i];
}
sort(belowlit, newsize);
return belowlit;
}
void main()
{
    Country cdata[10];
    //way 1
    input(cdata, 10);
    //way 2
    for (int i = 0; i < 10; i++)
    {
        cdata[i] = input();
    }
    int totalcount = 0;
    Country* list = getcountries(cdata, 10, totalcount);
}
```

Lecture # 32 String

We will discuss string data type in this lecture. String is used to store textual data. Just like char array data of string is stored consecutively and last index contains null character as terminator. Following points are necessary to remember for string data type.

- Use #include "string" header file.
- String is a sequence of character.
- Each character is stored at a one index.
- Unlike char array content of string can be compared using == operator.

Program 1: Input a string and compute its length.

```
#include "iostream"
#include "string"
using namespace std;
void main()
{
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
string s;
cout << "Enter text ";
cin >> s;
int i = 0;
while (s[i] != '\0')
    i++;
cout << "length " << i << endl;
}
```

2nd way.

```
#include "iostream"
#include "string"
using namespace std;
void main()
{
    string s;
    cout << "Enter text ";
    cin >> s;
    cout << s.length() << endl;
}
```

Program 2: Input two string and tell whether both strings are equal or not.

Solution:

```
#include "iostream"
#include "string"
using namespace std;
void main()
{
    string s1, s2;
    cout << "Enter first string ";
    cin >> s1;
    cout << "Enter second string ";
    cin >> s2;
    if (s1 == s2)
        cout << "Equal";
    else
        cout << "Not equal";
}
```

String concatenation: + operator is used to concatenate two or more than two strings. For example

string s = "PF" + "2020";

Will concatenate **PF2020** as a single string and store in s.

```
#include "iostream"
#include "string"
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
using namespace std;
void main()
{
    string s1 = "Pak", s2 = "ISTAN";
    string s3 = s1 + s2;
    cout << s3 << endl; //prints PakISTAN
}
```

Note that string are passed by value as well as passed by reference. For example.

void f1(string s) ; // f1 receives string s as value.

void f2(string &s); // f2 receives string as reference.

Program 3: Input a string and tell whether string is palindrome or not. For reverse of string pass string to function reverse that will return reverse of a function.

Solution:

```
#include "iostream"
#include "string"
using namespace std;
string reverse(string s)
{
    string rev = "";
    for (int i = s.length() - 1; i >= 0; i--)
        rev = rev + s[i];
    return rev;
}
void main()
{
    cout << "Enter a string ";
    string s;
    cin >> s;
    if (s == reverse(s))
        cout << "paineidrome\n";
    else
        cout << "Not palindrome\n";
}
```

Program 4: Input a string from user and count number of words in a string.

For example

I Love Pakistan

Logic:

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

Has three words. Count spaces and 1 in spaces and you will have words.

But if more than one spaces occurs between words than? Remove spaces.

Let us code it.

Solution:

```
#include "iostream"
#include "string"
using namespace std;
string trimstart(string s)
{
    while (s[0] == ' ')
    {
        for (int i = 0; i < s.length(); i++)
        {
            s[i] = s[i + 1];
        }
    }
    return s;
}
string trimend(string s)
{
    int i = 0;
    while (s[i] != '\0')
        i++;
    i -= 1;
    while (s[i] == ' ')
        s[i--] = '\0';
    return s;
}
string removeextraspaces(string s)
{
    string trimed = trimend(trimstart(s));
    for (int i = 0; i < trimed.length(); i++)
    {
        if (trimed[i] == ' ')
        {
            while (trimed[i + 1] == ' ')
            {
                for (int j = i + 1; j < trimed.length(); j++)
                    trimed[j] = trimed[j + 1];
            }
        }
    }
    return trimed;
}
int words(string s)
{
    s = removeextraspaces(s);
    int space = 0;
    int i = 0;
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
while (s[i] != '\0')
{
    if (s[i] == ' ')
        space++;
    i++;
}
if (i == 0)
    return 0;
return space + 1;
}
void main()
{
    string s;
    cout << "Enter a string ";
    getline(cin, s);
    cout << "Total words are " << words(s) << endl;
    system("pause");
}
```

