

NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES ISLAMABAD

PROGRAMMING FUNDAMENTALS LAB SPRING 2021

Lab Manual 14

Pointers - I

C++ Pointers

In C++, pointers are variables that store the memory addresses of other variables. Here is how we can declare pointers.

```
int *pointVar;
```

Here, we have declared a pointer `pointVar` of the `int` type. We can also declare pointers in the following way.

```
int* pointVar; // preferred syntax
```

Let's take another example of declaring pointers.

Assigning Addresses to Pointers

Here is how we can assign addresses to pointers:

```
int *pointVar, var;  
var = 5;  
  
// assign address of var to pointVar pointer  
pointVar = &var;
```

Here, `5` is assigned to the variable `var`. And, the address of `var` is assigned to the `pointVar` pointer with the code `pointVar = &var`.

Get the Value from the Address Using Pointers

To get the value pointed by a pointer, we use the `*` operator. For example:

```
int *pointVar, var;
```

```
var = 5;

// assign address of var to pointVar
pointVar = &var;

// access value pointed by pointVar
cout << *pointVar << endl; // Output: 5
```

In the above code, the address of `var` is assigned to `pointVar`. We have used the `*pointVar` to get the value stored in that address.

When `*` is used with pointers, it's called the **dereference operator**. It operates on a pointer and gives the value pointed by the address stored in the pointer. That is, `*pointVar = var`.

Changing Value Pointed by Pointers

If `pointVar` points to the address of `var`, we can change the value of `var` by using `*pointVar`. For example,

```
int var = 5;
int* pointVar;

// assign address of var
pointVar = &var;

// change value at address pointVar
*pointVar = 1;

cout << var << endl; // Output: 1
```

Here, `pointVar` and `&var` have the same address, the value of `var` will also be changed when `*pointVar` is changed.

C++ Pointers and Arrays.

In C++, [Pointers](#) are variables that hold addresses of other variables. Not only can a pointer store the address of a single variable, it can also store the address of cells of an [array](#). Consider this example:

```
int *ptr;
```

```
int arr[5];

// store the address of the first element of arr in ptr
ptr = arr;
```

Here, `ptr` is a pointer variable while `arr` is an `int` array. The code `ptr = arr;` stores the address of the first element of the array in variable `ptr`. Notice that we have used `arr` instead of `&arr[0]`. This is because both are the same. So, the code below is the same as the code above.

```
int *ptr;
int arr[5];
ptr = &arr[0];
```

The addresses for the rest of the array elements are given by `&arr[1]`, `&arr[2]`, `&arr[3]`, and `&arr[4]`.

Point to Every Array Elements

```
ptr + 1 is equivalent to &arr[1];
ptr + 2 is equivalent to &arr[2];
ptr + 3 is equivalent to &arr[3];
ptr + 4 is equivalent to &arr[4];
```

Similarly, we can access the elements using the single pointer. For example,

```
// use dereference operator
*ptr == arr[0];
*(ptr + 1) is equivalent to arr[1];
*(ptr + 2) is equivalent to arr[2];
*(ptr + 3) is equivalent to arr[3];
*(ptr + 4) is equivalent to arr[4];
```

Suppose if we have initialized `ptr = &arr[2];` then

```
ptr - 2 is equivalent to &arr[0];
ptr - 1 is equivalent to &arr[1];
ptr + 1 is equivalent to &arr[3];
ptr + 2 is equivalent to &arr[4];
```

Lab Tasks

Problem 01

Write a program that asks the user to enter integers as inputs to be stored in the variables 'a' and 'b' respectively. Also create two integer pointers named ptrA and ptrB. Assign the values of 'a' and 'b' to ptrA and ptrB respectively. Display the values and addresses of a and b using ptrA and ptrB.

Problem 02

Write a C++ program to accept five integer values from user. These five values will be stored in an array using a pointer. Then print the elements of the array in reverse order using the same pointer.

Problem 03

Write a C++ program to accept five integer values from user and store them in an array. Then find and print the largest and smallest integer value in the array using pointers.

Problem 04

Write a program to take a string as an input from the user and then calculate the length of the string using a pointer.

Problem 05

Write a program to take a string as an input from the user and then count the number of vowels and consonants in a string using a pointer.

Problem 06

Write a C++ program to accept five integer values from user and store them in an array. Then sort the array using pointer.

THE END