University of Central Punjab
(Incorporated by Ordinance No. XXIV of 2002 promulgated by Government of the Punjab)
FACULTY OF INFORMATION TECHNOLOGY

Programming Fundamentals

Lab 07	
Topic	Pointers
Objective	• POINTERS, 1D ARRAYS & FUNCTIONS ○ Sending a Pointer as a parameter to a function. ○ Explaining the difference between passing of an ordinary variable and of a pointer. ○ Explaining the difference between reference parameter and pointer parameter. ○ Changing the value of the pointed entity through pointer parameter. ○ Discuss use of constants involving pointers as function parameter. ○ Explaining the close affinity between pointers and 1D arrays. ○ Explaining operations allowed on pointers. (+,-) ○ Accessing 1D arrays through pointers. ○ Explaining the equivalence of arr[i] and *(arr+i). ○ Discuss use of constants involving pointers for 1D array. ○ Explaining a pointer parameter as referring to an array. ○ Explaining how to send an array as a pointer to a function. ○ Explaining how to send a part of an array to a function as pointer. ○ Discuss generic functions using pointers.

Lab Description:

This lab is basically designed for the pointer as parameters, constants involving pointers as parameter, operation of pointer and how to send partial array to a function.

Pointer as parameter to a function:

As we discuss earlier we can pass a variable to a function. There are two types of passing a variable, one is pass by value and another is pass by reference. Same rules are implemented on pointers as well. We can pass a pointer as a parameter to a function. Pointer can be pass by value and pass by reference as well.

Prototype:

```
void display(int*,int); //in case of intger array
```

OR

```
void display(int*); //in case of integer variable
```

Calling:

```
display(arr,size); //in case of array
```

OR

```
display(&arr[0]); //in case of specific index of array
```

OR

```
display(&a); // in case of variable
```

Definition:

```
void display(int *ptr, int size)
{
    for (int i = 0; i < size; i++)
        cout << *(ptr+i);
    cout << endl;
}
```

Pointer parameter pass by value:

Pointers can be passed by value to a function as a parameter. As we discussed value of a pointer is an address of a memory location.

Pointer parameter pass by reference:

Pointers can be passed by reference to a function as a parameter. As we discussed earlier in case of reference parameter, actual parameter and formal parameter share same memory location. Same rules are applicable on pointers.

Difference between pointer parameter and reference parameter:

- Simple pointer parameter basically passed by value but the value of pointer is an address
- In reference parameter actual and formal parameters shared same memory location.

Operation on pointers:

We can use +,-, ++ and -- operators with pointers. We can add or subtract a value into a pointer. By adding or subtracting a value we calculate next or pervious address in memory. But in case of adding or subtracting our pointer holds same address. In case of increment or decrement operator we calculate next or previous address in memory and update the value of pointer with new calculated address.

Passing partial array to a function:

We can pass full array or portion of an array. But we need to change parameters of function. If we want to pass a portion of an array than we need to pass starting address and ending address of that portion which we want to pass as parameter. This scenario will also help to create generic functions. As we discuss earlier how to display an array. But we see that function display all elements of array instead of specific portion. But the working of a display function is to display any portion of array not only full array.

Generic Function Example:

Prototype:

```
void display(int *, int *);
```

Calling:

```
display(&arr[0],&arr[size]);
```

OR

```
display(arr, arr+size);
```

Definition:

```
void display(int *start, int *end)
{
    for (int i = 0; start+i < end; i++)
        cout << *(start+i)<<" ";
    cout << endl;
}
```

Lab Tasks

Content problem:

We have two files which have similar names. We are not sure that the content of both files are same or not. Design a solution in order to solve this problem. You have to compare the content of both files with each other.

Tour record problem:

A tour company planned a tour of northern area. Two of the team members maintain the list of people how apply for the tour in their own files. But there is a confusion they write some record in both files which cause duplicate data. You have to design a solution which use both files and remove all duplicate entries from it.

Student record problem:

UCP wants to maintain the contact detail of their employs. But it is not an easy job to identify the record if it is unsorted. Design a solution in which records are inserted in sorted form which may help to find a record.

Irrelevant comments problem:

There are multiple websites which have information regarding different mobiles. Whatmobile is one of them. They also provide comment section as reviews of people regarding mobile. But some people post there ads as comment which is irrelevant as review. Whatmobile wants to remove all those comments which are irrelevant.

YouTube views problem:

As we all know all youtubers want more views in order to increase their earning. YouTube want to communicate with their content creators about subscribers. They want to inform those youtubers how have 100k above views but most of their viewers did not subscribe them. YouTube maintain this record in a file. You need to identify those record to communicate with youtubers.