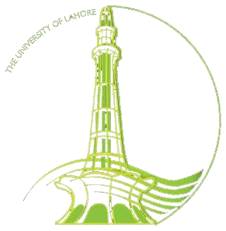

THE UNIVERSITY OF LAHORE
ISLAMABAD CAMPUS

Data Structures and Algorithms
(CS09203)

Lab Report

Name: Jassim Bashir
Registration #: SEU-F16-101

Lab Report #: 01
Dated: 16-04-2018
Submitted To: Sir. Usman Ahmed

The University of Lahore, Islamabad Campus

Department of Computer Science & Information Technology

Experiment # 01
Introduction to Arrays and its operation

Objective
The objectives of this lab session are to understand the basic and various operations on arrays in C++.

Software Tool
1. I use Code Blocks with GCC compiler.

1 Theory

We have already studied array in our computer programming course. We would be using the knowledge we learned there to implement different operation on arrays.

Traversing Linear Arrays:-

Let A be the collection of data elements stored in the memory of the computer. Suppose we want to print the contents of each element of A or suppose we want to count the number of elements of A with a given property. This can be accomplished by traversing A that is by accessing and Processing each element of A exactly once.

The following algorithm traverses a linear array. The simplicity of the algorithm comes from the fact that LA is a linear structure. Other linear structures such as linked list can also be easily traversed. On the other hand the traversal of non-linear structures such as trees and graphs is considerably more complicated.

Algorithm:- (Traversing a Linear Array) Here LA is a linear Array with lower Bound LB and upper Bound UB. This algorithm traverses LA. Applying an operation PROCESS to each element of LA.

[Initialize Counter] Set X=LB. 1. Repeat Step 3 and 4 while K_i=UB. [Visit element] Apply PROCESS to LA[X]. [Increase Counter] Set X=X+1. [End of Step 2 Loop] 5. Exit.

Inserting and Deleting: - Let A be a collection of data elements in the memory of computer. Inserting refers to the operation of adding another element to the collection A and deleting refers to the operation of removing one of the elements from A. Here we discuss the inserting and deleting when A is a linear array. Inserting an element at the end of the linear array can be easily done provided the memory space allocated for the array is large enough to accommodate the additional element. On the other hand suppose we need to insert an element in the middle of the array. Then on average half of the elements must be moved downward to the new location to accommodate the new element and keep the order of other elements.

Similarly deleting the element at the end of an array presents no difficulties but deleting the element somewhere in the middle of the array would require that each subsequent element be moved one location upward in order to fill up the array.

Algorithm of Insertion operation: - (Inserting into Linear Array) INSERT (LA, N, K, ITEM)

Here LA is a linear array with N elements and K is a positive integer such that KN. This algorithm inserts an element ITEM into the Kth position in LA.

1. [Initialize Counter] Set J=N. 2. Repeat Step 3 and 4 while JK. 3. [Move Jth element downward] Set LA [J+1] =LA[J]. 4. [Decrease Counter] Set J=J-1. End of Step 2 Loop. 5. [Insert element] Set LA[K]=ITEM. 6.
[Reset N] Set N=N+1. 7. Exit.

2 Task

2.1 Procedure: Task 5

Write a C++ program to implement all the above described algorithms and display the following menu and ask the user for the desired operation.

2.2

#include<iostream>

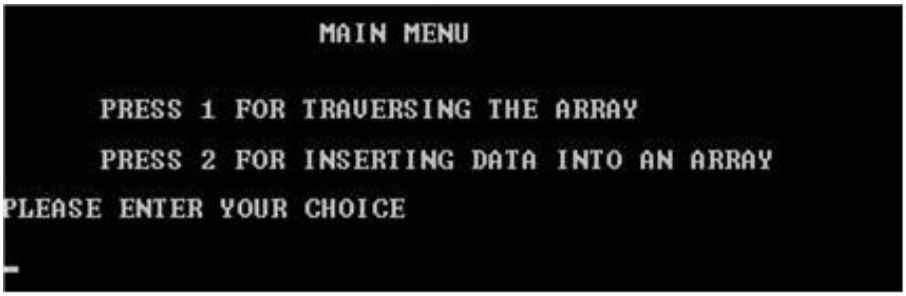

Figure 1: output

using namespace std ; **int**
main()
{ **int** A[100] ,K=0,UB; **int**
LB=0,counter=0,n; loop :
cout<<endl<<"MAIN MENUE"<<endl ; cout<<"1: ENTER ELEMENT INT HE ARRAY : "<<endl ; cout<<"2: FOR TRIVERSES THE ARRAY :"<<endl ; cin>>n; **switch**(n)
{
case 1:{ cout<<"Enter the Array size less than 100:" ; cin>>UB;
cout<<"Enter the elements
while(K<UB){ counter++;
cin>>A[K] ; K++; }} **break**; in array: \n"; **case** 2:{ cout<<"The Traverse of array is :\n" ; K=LB; **while**(K<UB)
{ cout<<A[K]<<" " ;
K++; }} **break**;} **goto**
loop ; **return** 0;
}

3 Conclusion

In today lab we study about the array (1D and 2D) both how to create and fill or display the array first we have written the code on copy and the implemented it on computers.