NATIONAL UNIVERSITY OF MODERN LANGUAGES

21/18/2021

Department Of Computer Science

Subject: Data Structure and Algorithm Instructor: Ma’am Zainab Malik

Lab No: 2 Date: 21-10-2021

Class: BSCS-3B

Students’ Name

1) Madina Javed Iqbal Khokhar	2426
2) Esha Mansoor	2413
3) Sultan Zahid	2411
4) Abdul Moeed	2419

Lab Repot 3

Task :

Implement a program that will perform following functions using an array

- Initialization
- Printing an array
- Linear search
- Bubble Sort
- Binary Sort
- Insertion Sort
- Selection Sort
- Replace and finding largest number

Description:

This program is illustrating how to implement some basic types of sorting or searching on an array.

In the main function ,we have we have declare a variable named opt of type integer and an array of size ten .Then, we have use do-while condition in which we have asked the user to type any integer value from 0-9 to carry out some particular functions for instance; initializing an array, printing an array, bubble sorting, linear search , binary search, insertion sorting, selection, finding largest number and replacing (Each represent a particular function).Lastly, we have use switch statement to carry out task on the basis of integer value type by the user.

Outside the main function, we have declared a function named Initialization in which we will initialize an array using for loop. Similarly , in the next function named printArray we are printing our array using for loop. Moving onward, we will built a function for linear ,binary search, insertion, bubble, selection sort,printing, replacing and to terminate a program.Each function will be called in the main function according to the value entered by the user.If user entered 0,our program will terminate.

Code:

```
#include<iostream>
using namespace std;
// here are some user defined functions
void initialization(int arr[], int n);
void printArray(int arr[], int n);
int linearSearch(int arr[], int n);
void bubbleSort(int arr[], int n);
int binarySearch(int arr[], int n);
void insertionSort(int arr[], int n);
int largest (int arr[], int n);
void selectionSort(int arr[], int n);
int replace(int arr[], int n);
// here's our main  function start
int main()
{
int opt;//to get any optional value from the user
int arr[10];
int n=10;
do{
cout<<"Press 1 to initialize the array "<<endl;
```

```
cout<<"Press 2 to perform linear search"<<endl;
cout<<"Press 3 to perform binary search"<<endl;
cout<<"Press 4 to perform bubble sort"<<endl;
cout<<"Press 5 to perform selection sort"<<endl;
cout<<"Press 6 to perform insertion sort"<<endl;
cout<<"Press 7 to print the array"<<endl;
cout<<"Press 8 to search and replace in the array"<<endl;
cout<<"Press 0 to terminate the process"<<endl;
cin>>opt;  // Getting an an optional value

 switch(opt)
 {
 case 1:
 initialization(arr,n);
 break;
 case 2:
 linearSearch(arr,n);
 break;
 case 3:
 bubbleSort(arr,n);
binarySearch(arr,n);
 break;
 case 4:
 bubbleSort(arr,n);
 printArray(arr,n);
 break;
 case 5:
 selectionSort(arr,n);
 printArray(arr,n);
 break;
 case 6:
 insertionSort(arr,n);
 printArray(arr,n);
 break;
 case 7:
 printArray(arr,n);
 break;
 case 8:
 replace(arr,n);
 printArray(arr,n);
 break;
case 0:
 exit(-1);

}// end of switch statement

 }//do

 while(true);
```

```
return 0;
}//closing of main

void initialization(int arr[], int n) // start of initialization(int arr[], int n)
{
cout<<"Initializing array"<<endl;
for (int i=0;i<n;i++)
{
cout<<"Provide value at index "<<i<<": \n";
cin>>arr[i];
}//for loop
}//end of initialization(int arr[], int n)

void printArray(int arr[], int n) //start of printArray(int arr[], int n)
{
for (int i=0;i<n;i++)
{
 cout<<arr[i]<<" ";
}// end of for
cout<<endl;
}//end of printArray(int arr[], int n)

int linearSearch(int arr[], int n) //start of linearSearch(int arr[], int n)
{
int item;
```

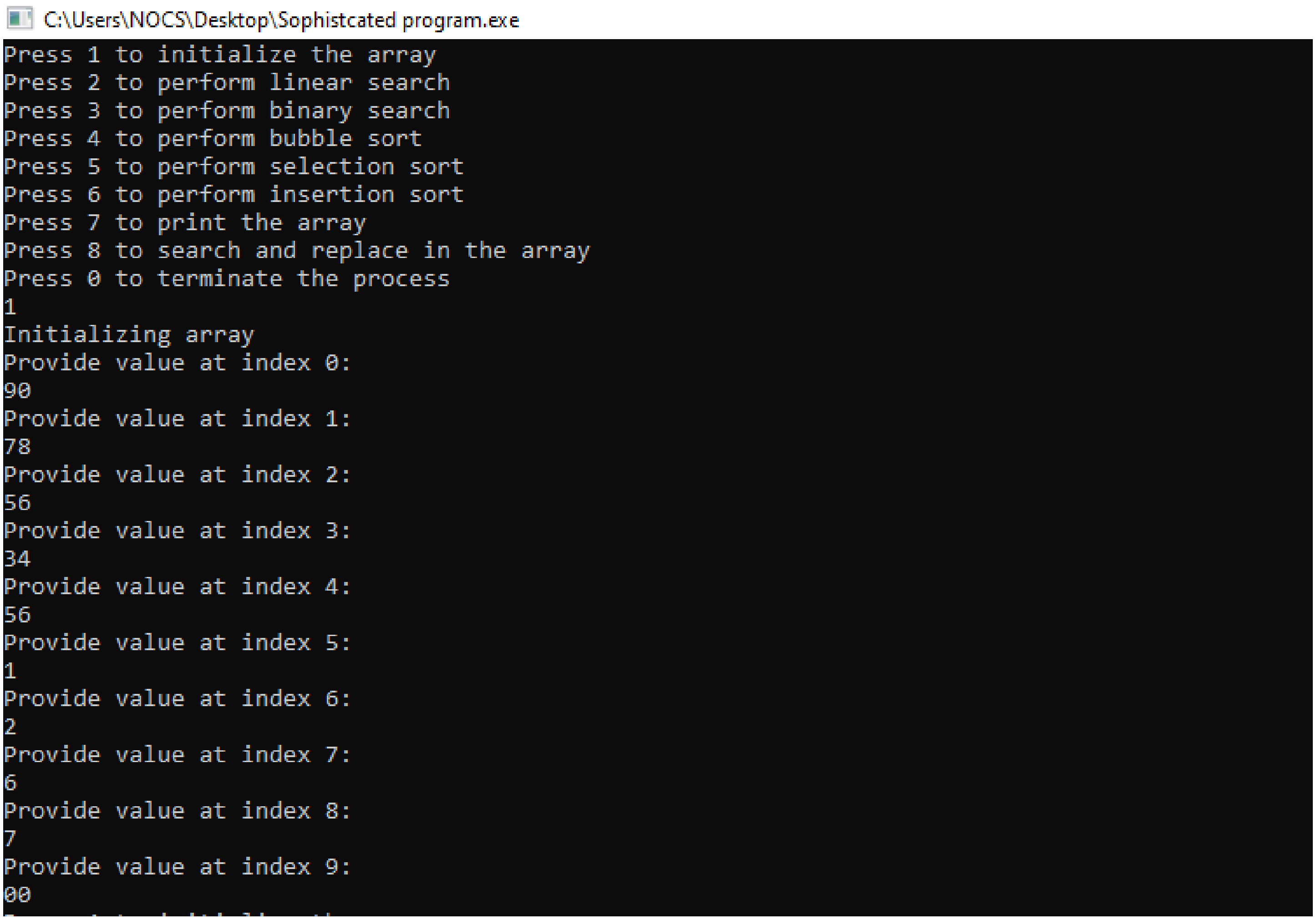
```
cout<<"Provide the item you want to search \n";
cin>> item;
int loc=-1;
for (int i=0;i<n;i++)
{
 if (item==arr[i])
 {
 loc=i;
 break;
 }
}//ending value assigning
 if (loc==-1)
 {
 cout<<"Value not found \n";
 }
 else
 {
 cout<<"Value found at index: "<<loc;
cout<<endl;
 }
 return loc;
}//end of linearSearch(int arr[], int n)

void bubbleSort(int arr[], int n) //start of bubble sort
{
```

```
for (int marker=n-1;marker>0;marker--)
{
for (int i=0; i<marker; i++)
{
if (arr[i]>arr[i+1])
{
int temp=arr[i];
arr[i]=arr[i+1];
arr[i+1]=temp;
}
}//for i
}//for marker
}//end of bubble sort

int binarySearch(int arr[], int n)  // start of binary Search
{
int item;
cout<<"Provide the item you want to search \n";
cin>> item;
int loc=-1;
int beg=0;
int end=n-1;
int mid=(beg+end)/2;
while (beg<=end && arr[mid]!=item)
{
```

```
if (item<arr[mid])//need to search backward direction
{
end=mid-1;
}
else//need to search in forward direction
{
beg=mid+1;
}
mid=(beg+end)/2;
}//while
if (beg>end)
{
cout<<"Value not found \n";
}
else
{
cout<<"Value found at index: "<<mid<<endl;
}
return mid;
}// end of binary Search
void insertionSort(int arr[],int n) //start of insertionsort
{for (int i=1;i<n;i++)
{ int j;
int key=arr[i];
 for (int j=i-1; j>=0 && arr[j]>key;j--)
 {
 arr[j+1]=arr[j];
 }//inner for loop
 arr[j+1]=key;

}//outer for loop
}//end of insertionsort
int largest (int arr[], int n)
{
 int max=0;
 for (int i=0;i<n;i++) //could also be for (int i=0;i<=n;i++)
 {
 if (arr[max]<arr[i])
 {
 max=i;
 }//comparison for largest value

 } //loop for largest value

 return max;
}
void selectionSort(int arr[], int n) //  start of selectionSort
{
 for (int marker=n; marker>0; marker--)
 {
```

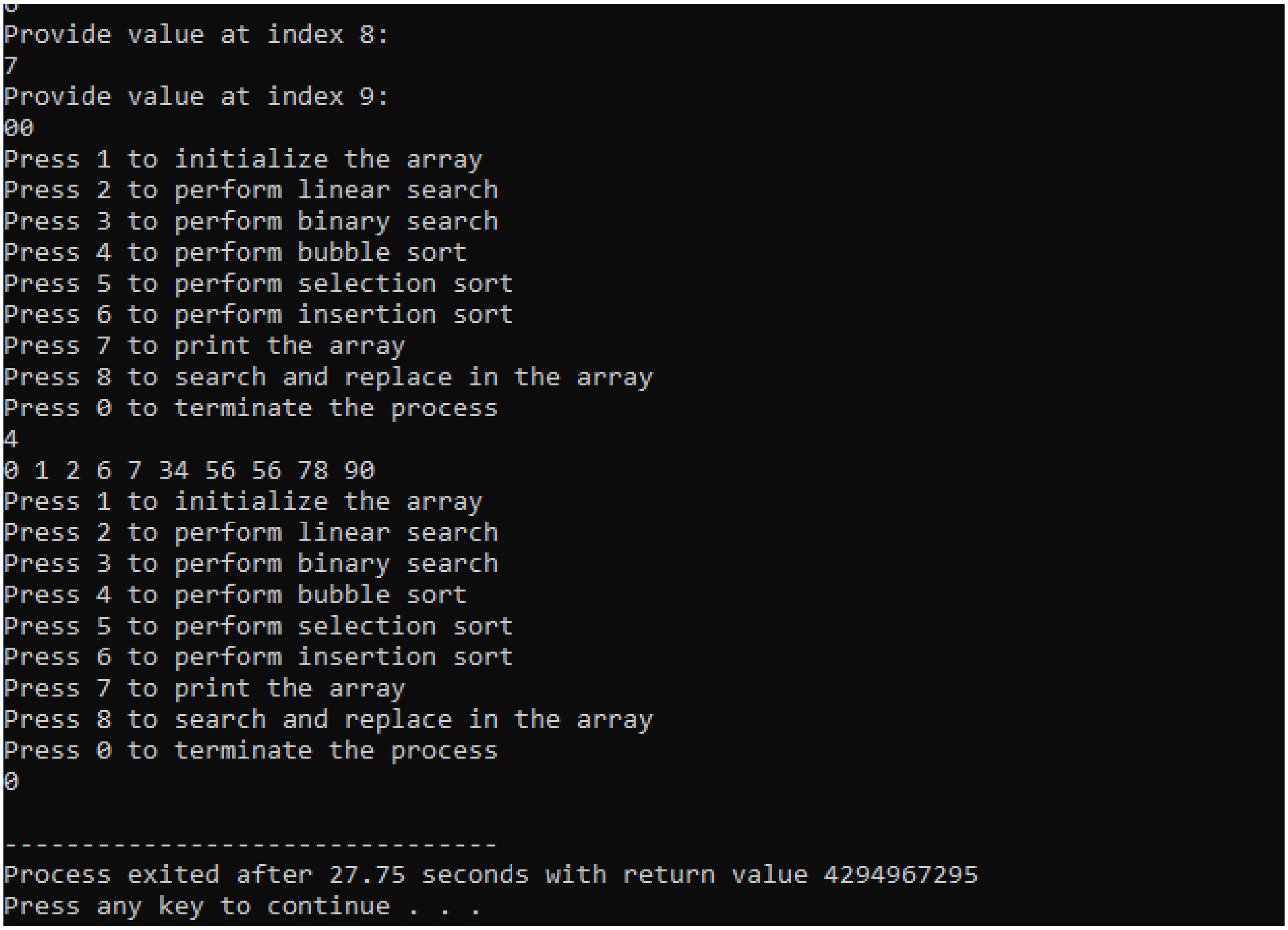
```
 int max= largest(arr,marker+1); //only int max=largest(arr,marker) when i<=n, in int largest.


 int temp=arr[max];
 arr[max]=arr[marker];
 arr[marker]=temp;
 } //marker loop


}//  end of selectionSort
int replace (int arr[], int n)
{
Int find=binarySearch(arr,n);
//To replace an array,we will  firstly  use any search method to search our value.
int newvalue;//initialize a new value to replace with existing
cout<<"Enter the value you want to replace the found value with \n";
cin>>newvalue;
cout<<endl;
arr[find]=newvalue;//replace our value
}
```

Output:



C:\Users\NOCS\Desktop\Sophistcated program.exe

```
Press 1 to initialize the array
Press 2 to perform linear search
Press 3 to perform binary search
Press 4 to perform bubble sort
Press 5 to perform selection sort
Press 6 to perform insertion sort
Press 7 to print the array
Press 8 to search and replace in the array
Press 0 to terminate the process
1
Initializing array
Provide value at index 0:
90
Provide value at index 1:
78
Provide value at index 2:
56
Provide value at index 3:
34
Provide value at index 4:
56
Provide value at index 5:
1
Provide value at index 6:
2
Provide value at index 7:
6
Provide value at index 8:
7
Provide value at index 9:
00
```



```
Provide value at index 8:
7
Provide value at index 9:
00
Press 1 to initialize the array
Press 2 to perform linear search
Press 3 to perform binary search
Press 4 to perform bubble sort
Press 5 to perform selection sort
Press 6 to perform insertion sort
Press 7 to print the array
Press 8 to search and replace in the array
Press 0 to terminate the process
4
0 1 2 6 7 34 56 56 78 90
Press 1 to initialize the array
Press 2 to perform linear search
Press 3 to perform binary search
Press 4 to perform bubble sort
Press 5 to perform selection sort
Press 6 to perform insertion sort
Press 7 to print the array
Press 8 to search and replace in the array
Press 0 to terminate the process
0


--------------------------------
Process exited after 27.75 seconds with return value 4294967295
Press any key to continue . . .
```

THANKS