Green University of Bangladesh

Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering

Semester: 3rd - Fall semester / Year:2022

BSc in CSE (Day)

LAB REPORT NO - 02

Course Title: Data Structure Lab

Course Code: CSE 106 / Section: DE-221

Lab Experiment Tittle :

Implement bubble sort, Insertion sort and selection sort in a same program using user choice (switch case).

Student Details

	Name	ID
1.	Khondokar Saim	221902353

Lab Date : 26 / 10 / 2022

Submission Date : 01 / 11 / 2022

Course Teacher's Name : Farhana Akter Sunny.

[For Teachers use only: Don't Write Anything inside this box]

Lab Report Status

Marks: ..... Signature: .....

Comments: ..... Date: .....

❑ Tittle of the Lab Experiment :

Implement Bubble sort, Insertion sort and Selection sort in a same program using user choice (switch case).

❑ Objectives :

- Switch case statement evaluates a given expression and based on the evaluated value(matching a certain condition), it executes the statements associated with it. Basically, it is used to perform different actions based on different conditions(cases).
- Bubble sort is a basic algorithm for arranging a string of numbers or other elements in the correct order. The method works by examining each set of adjacent elements in the string, from left to right, switching their positions if they are out of order.
- Insertion Sort is a sorting algorithm where the array is sorted by taking one element at a time. The principle behind insertion sort is to take one element, iterate through the sorted array & find its correct position in the sorted array.
- Selection sort is a sorting algorithm that selects the smallest element from an unsorted list in each iteration and places that element at the beginning of the unsorted list.

1. Implement Bubble sort, Insertion sort and Selection sort in a same program using user choice (switch case).

Algorithm :

step - 1 : Insert all the necessary data types

step - 2: make category option for option selection
1.Bubble sort
2.Insertion Sort
3.Selection Sort

step - 3 : input the number of your desire option
step - 4 : make a Switch Statement with the operation data types .
Because of we have 3 category that's why we make three case option in switch statement

step - 5 : In case-1 we'll run the Bubble sort operation .
-declare the number of elements and enter the elements
- write the logics of Bubble sort code
-for outer loop
for(x=0; x<number; x++)
-for inner loop
for(y=x+1; y<number; y++)
{
if(array[x]>array[y])
{
temp=array[x];
array[x]=array[y];
array[y]=temp;
}
}
-print the Bubble Sort result
printf("\nBubble Sort Result : ");
for(x=0; x<number; x++)
{
printf("%d ",array[x]);
}

step - 6 : insert break option in end of the case-1

step - 7 : In case-2 we'll run the Insertion sort operation .

-declare the number of elements and enter the elements

- write the logics of Insertion sort code

-for outer loop

```
for(x = 1; x <= number - 1; x++)
```

-for inner loop

```
for(y = x; y > 0 && array[y - 1] > array[y]; y--)
{
        temp = array[y];
        array[y] = array[y - 1];
        array[y - 1] = temp;
 }
 }
```

-print the Bubble Sort result

```
printf("\nInsertion Sort Result : ");
for(x = 0; x < number; x++)
{
printf(" %d ", array[x]);
 }
printf("\n");
```

step - 8 : insert break option in end of the case-2

step - 9 : In case-3 we'll run the Selection sort operation .

-declare the number of elements and enter the elements

- write the logics of Insertion sort code

-for outer loop

```
for (x=0; x<number; x++)
{
min = x;
```

-for inner loop

```
for (y=x+1; y<number; y++)
 {
        if (array[y] < array[min])
        min = y;
 }
        temp = array[x];
        array[x] = array[min];
        array[min] = temp;
 }
```

-print the Bubble Sort result

```
printf("\nSelection Sort Result : ");
```

step - 10 : insert break option in end of the case-3

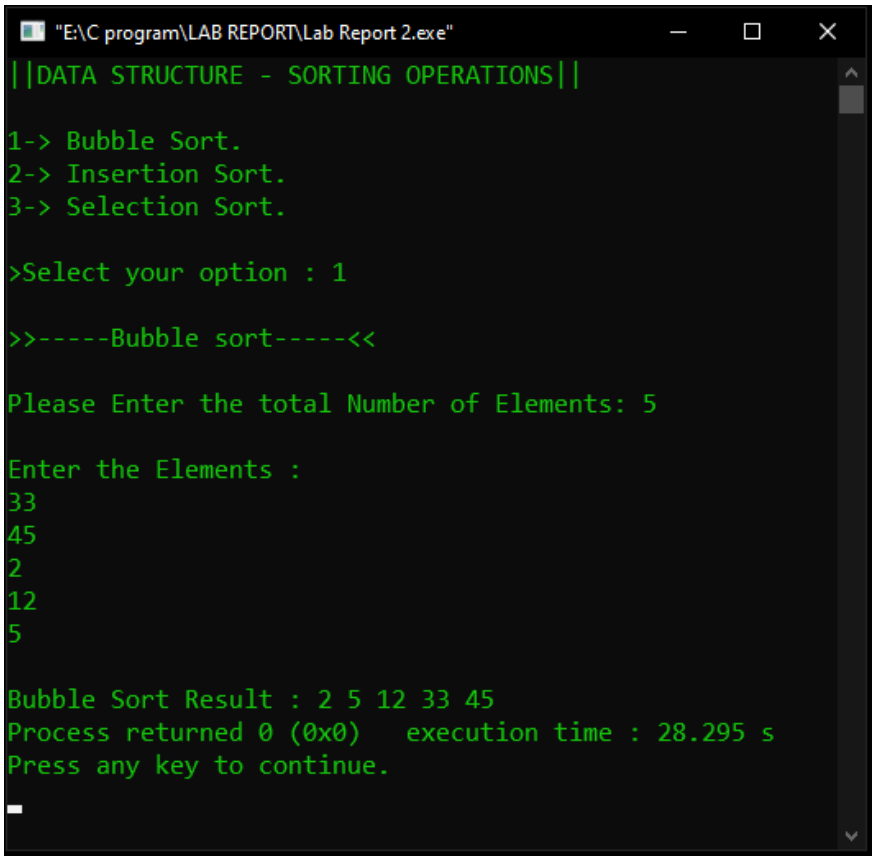
Code:

```
#include<stdio.h>
int main()
{
   int array[100],x,y,temp,number,min,operation;

   /* telling the user to choose any option */
   printf("||DATA STRUCTURE - SORTING OPERATIONS||\n");
   printf("\n1-> Bubble Sort.\n");
   printf("2-> Insertion Sort.\n");
   printf("3-> Selection Sort.\n");

   //input the number of your desire option
   printf("\n>Select your option : ");
   scanf("%d",&operation);

   switch(operation)
   {
   //case 1 for Bubble sort
   case 1:
      printf("\n>>-----Bubble sort-----<<\n");

      printf("\nPlease Enter the total Number of Elements: ");
      scanf("%d",&number);
      printf("\nEnter the Elements : \n");
      for(x=0; x<number; x++)
      {
         scanf("%d",&array[x]);
      }
      //outer loop
      for(x=0; x<number; x++)
      {
         //inner loop
         for(y=x+1; y<number; y++)
         {
            if(array[x]>array[y])
            {
               temp=array[x];
               array[x]=array[y];
               array[y]=temp;
            }
         }
      }
```

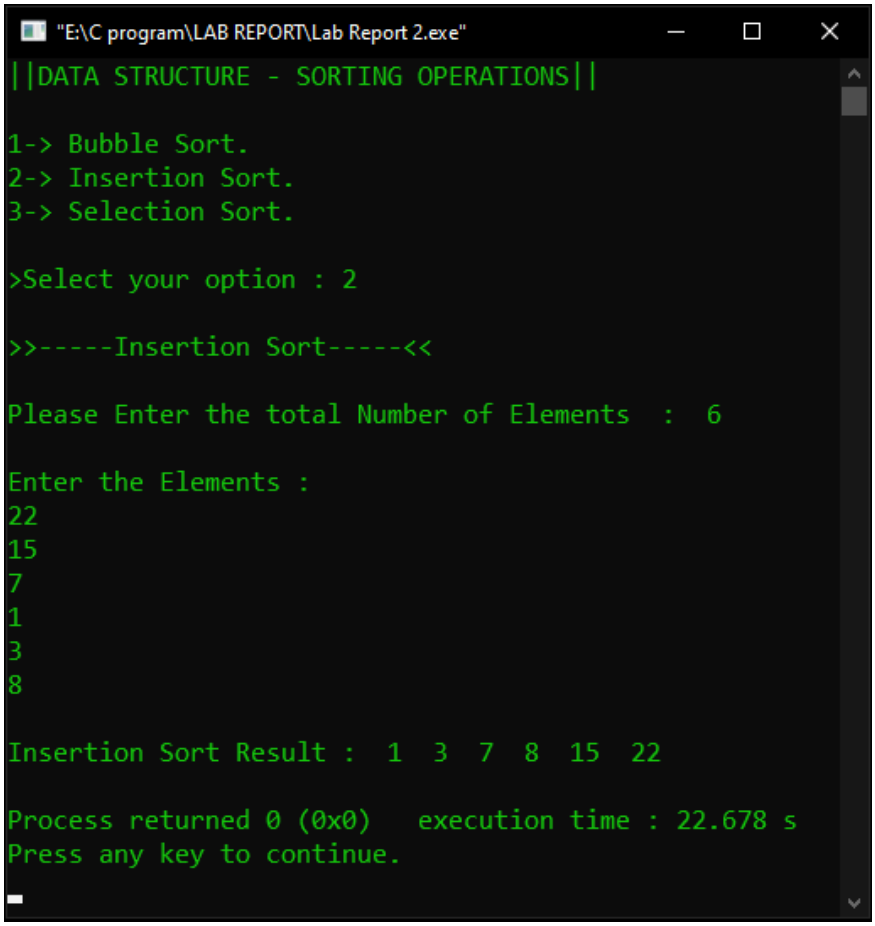
```
printf("\nBubble Sort Result : ");
      for(x=0; x<number; x++)
      {
         printf("%d ",array[x]);
      }

      break;

   //case 2 for Insertion Sort
   case 2:
      printf("\n>>-----Insertion Sort-----<<\n");

      printf("\nPlease Enter the total Number of Elements  :  ");
      scanf("%d", &number);

      printf("\nEnter the Elements : \n");
      for(x = 0; x < number; x++)
      {
         scanf("%d", &array[x]);
      }
      //outer loop
      for(x = 1; x <= number - 1; x++)
      {
         //inner loop
         for(y = x; y > 0 && array[y - 1] > array[y]; y--)
         {
            temp = array[y];
            array[y] = array[y - 1];
            array[y - 1] = temp;
         }
      }
      printf("\nInsertion Sort Result : ");
      for(x = 0; x < number; x++)
      {
         printf(" %d ", array[x]);
      }
      printf("\n");

      break;
```

```
    //case 3 for Selection Sort
    case 3:
        printf("\n>>-----Selection Sort-----<<\n");

        printf("\nPlease Enter the total Number of Elements: ");
        scanf("%d", &number);
        printf("\nEnter the Elements : \n");
        for(x=0; x<number; x++)
        {
            scanf("%d", &array[x]);
        }
        //outer loop
        for (x=0; x<number; x++)
        {
            min = x;
            //inner loop
            for (y=x+1; y<number; y++)
            {
                if (array[y] < array[min])
                    min = y;
            }
            temp = array[x];
            array[x] = array[min];
            array[min] = temp;
        }

        printf("\nSelection Sort Result : ");
        for(x=0; x<number; x++)
        {
            printf("%d ",array[x]);
        }

        break;

    }

}
```

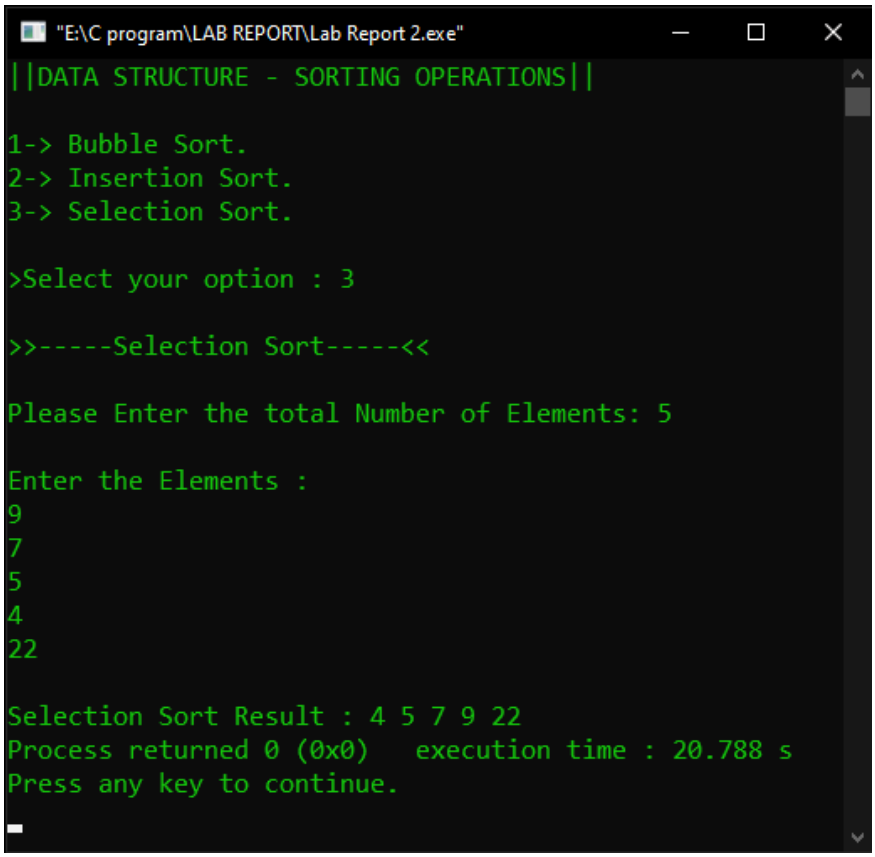
Output:



Case – 1 For Bubble Sort.



Case – 2 For Insertion Sort.



Case – 3 For Selection Sort.

❑ Analysis and Discussion :

- We got the exact result on output. Sometimes the result was wrong but we found the right implementation.
- The problem of displaying anything in output is the easiest implementation. We solve that very easily.
- In this assignment, we faced some problems in this question but with the teacher's help we solve it.
- All program is easy to understand and these helped me a lot to remove my confusion about Bubble sort, Insertion sort and Selection sort of a Data and basic array's operations.
- I learnt display something in program, Sorting operations as like , Bubble sort, Insertion sort and Selection sort array insertion , switch statement on program and many basic things about c programming.