



- Subject : Data Structures and Algorithms (Lab)
- Submitted To : Sir Muhammad Adil
- Submitted by : Akmal afridi
- Degree : BS(SE)
- ID # 14226
- Date : 03/10/2020

Task#1: Code a program in any computer language of your choice to implement Algorithm for inserting an element in a One-Dimensional Array.

Answer :

### C program

```
#include <stdio.h>

int main()
{
    int array[100], position, c, n, value;
    printf("Enter number of elements in array\n");
    scanf("%d", &n);
    printf("Enter %d elements\n", n);
    for (c = 0; c < n; c++) scanf("%d", &array[c]);
    printf("Enter the location where you wish to insert an element\n");
    scanf("%d", &position);
    printf("Enter the value to insert\n");
    scanf("%d", &value);
    for (c = n - 1; c >= position - 1; c--)
        array[c+1] = array[c];
    array[position-1] = value;
    printf("Resultant array is\n");
    for (c = 0; c <= n; c++)
```

```
    printf("%d\n", array[c]);  
return 0;  
}
```



Task#2:

Code a program in any computer language of your choice to implement Linear Search Algorithm.

Answer:

### **C program**

```
#include<stdio.h>  
  
int main()  
{  
    int arr[10], count, element, search = 0;  
    printf("\nEnter 10 Elements in Array:\t");  
    for(count = 0; count < 10; count++)  
    {  
scanf("%d", &arr[count]);  
    }  
    printf("\nElements in Array are:\n");
```

```
for(count = 0; count < 10; count++)
{
    printf("%d\t", arr[count]);
}

printf("\n");

printf("\nEnter Element To Search:\t");

scanf("%d", &element);

for(count = 0; count < 10; count++)
{
    if(arr[count] == element)
    {
        printf("\nElement %d found at Position %d\n", element, count + 1);
        search++;
    }
}

if(search == 0)
{
    printf("\nElement %d Not Found in Array\n", element);
}

printf("\n");

return 0;
}
```



Task#3: Suppose there is a linear array A[]

A[] = {30, 50, 20, 10, 20, 35, 40}

- i. Design a program to sort the above list using Selection Sort.
- ii. Design a program to sort the above list using Bubble Sort.

i) Answer:

**Java program for implementation of Selection Sort:**

```
class SelectionSort
{
void sort(int arr[])
{
    int n = arr.length;
    for (int i = 0; i < n-1; i++)
    {
        int min_idx = i;
        for (int j = i+1; j < n; j++)
            if (arr[j] < arr[min_idx])
                min_idx = j;
    }
}
```

```
        int temp = arr[min_idx];
        arr[min_idx] = arr[i];
        arr[i] = temp;
    }
}

void printArray(int arr[])
{
    int n = arr.length;
    for (int i=0; i<n; ++i)
        System.out.print(arr[i]+" ");
    System.out.println();
}

public static void main(String args[])
{
    SelectionSort ob = new SelectionSort();
    int arr[] = {30,50,20,10,20,35,40};
    ob.sort(arr);
    System.out.println("Sorted array");
    ob.printArray(arr);
}
}
```



ii) Answer:

### Bubble Sort - C program to sort an Array

```
#include <stdio.h>

int main()
{
    int arr[MAX],limit;
    int i,j,temp;
    printf("Enter total number of elements: ");
    scanf("%d",&limit);
    printf("Enter array elements: \n");
    for(i=0; i<limit; i++)
    {
        printf("Enter element %3d: ",i+1);
        scanf("%d",&arr[i]);
    }
    for(i=0; i<(limit-1); i++)
    {
        for(j=0; j<(limit-i-1); j++)
        {
            if(arr[j]>arr[j+1])
            { temp=arr[j];
```

```
        arr[j]=arr[j+1];
        arr[j+1]=temp;
    }
}
}

printf("Array elements in Ascending Order:\n");

for(i=0; i<limit; i++)
    printf("%d ",arr[i]);

printf("\n");

for(i=0; i<(limit-1); i++)
{
    for(j=0; j<(limit-i-1); j++)
    {
        if(arr[j]<arr[j+1])
        {
            temp=arr[j];
            arr[j]=arr[j+1];
            arr[j+1]=temp;
        }
    }
}
}
```

