Week - 1

1. Write a C- program to Reverse an Array.

program

```
#include<stdio.h>
#include<conio.h>
int main()
{
  int  a[10], i, n;
 printf("Enter the size of an Array:\n");
  scanf("%d", &n);
 printf("Enter the elements into an Array\n");
 for(i=0; i<n; i++)
   scanf("%d", &a[i]);
 printf("\n After Reversing an Array elements are:");
 for(i=n-1; i>=0; i--)
   printf("%d\t", a[i]);
  getch();
  return 0;
}
```

output

2. C programs to implement the Searching Techniques - Linear Search & Binary Search.

2.1 C-program to implement Linear Search.

Source Code

```
#include<stdio.h>
int main()
{
   int a[50], search, i, n;
   printf("Enter the no. of elements in array\n");
   scanf("%d", &n);
   printf("Enter elements into an Array\n");
   for(i=0; i<n; i++)
      scanf("%d", &a[i]);
   printf("Enter a number to search:\n");
   scanf("%d", &search);
   for(i=0; i<n; i++)
   {
      if(a[i]==search)
      {
         printf("The search element %d is present at
                 location %d.\n", search, i+1);
         break;
      }
   }
   if(i==n)
      printf("%d is not present in the array.\n", search);
   return 0;
}
```

output

2.2 C- Program to Implement Binary Search.

Tip: Enter data elements in Ascending order only like, 15, 20, 21, 32, 49, 62 ... etc.

Program

```
#include<stdio.h>
int main()
{
  int i, l, r, mid, n, search, a[50];
  printf("Enter the size of the array:\n");
  scanf("%d", &n);
  printf("Enter elements into an Array\n");
  for (i=0; i<n; i++)
   scanf("%d", &a[i]);
  printf("Enter value to find:\n");
  scanf("%d", &search);
  l=0;
  r=n-1;
  mid=(l+r)/2;
  while (l<=r)
   {
    if (a[mid]<search)
     l=mid+1;
    else if (a[mid]==search)
     {
      printf("%d found at location %d.\n", search, mid+1);
      break;
     }
    else
     r=mid-1;
     mid=(l+r)/2;
   }
   if (l>r)
    printf("Not found! %d is not present in the list.\n", search);
    return 0;
}
```

3. C - programs to implement Sorting Techniques. —
Bubble, Selection and Insertion sort

3.1 C- program to implement Bubble Sort.

Source Code:

```
#include<stdio.h>
int main()
{
  int a[50], n, i, j, temp, flag;
  printf("Enter no. of elements\n");
  scanf("%d", &n);
  printf("Enter elements of an Array\n");
  for(i=0; i<n; i++)
   scanf("%d", &a[i]);
  for(i=0; i<n-1; i++)
  {
    flag=0;
   for(j=0; j<n-1-i; j++)
    {
      if(a[j] > a[j+1])
       {
         temp = a[j];
         a[j] = a[j+1];
         a[j+1] = temp;
          flag=1;
       }
    }
   if(flag==0)
    break;
  }
 printf("sorted list in ascending order:\n");
  for(i=0; i<n; i++)
  printf("%d\t", a[i]);
} return 0;
```

3.2 C- Program to implement Selection sort.

Source Code

```
#include<stdio.h>
int main()
{
 int a[100],n,i,j,min,temp;
 printf("Enter number of elements\n");
 scanf("%d",&n);
 printf("Enter elements of an array\n");
 for(i=0;i<n;i++)
  scanf("%d",&a[i]);
 for(i=0;i<n-1;i++)
  {
     min=i;
     for(j=i+1;j<n;j++)
      {
        if(a[j]<a[min])
          {
            min=j;
          }
      }
     if(min!=i)
      {
       temp=a[i];
       a[i]=a[min];
       a[min]=temp;
      }
  }
     printf("sorted list by selection sort is in ascending
                                          order:\n");
     for(i=0;i<n;i++)
     printf("%d\t",a[i]);
     return 0;
}
```

Insertion sort

```
#include<stdio.h>
int main()
{
  int n, a[50], i, j, temp;
  printf("Enter number of elements\n");
  scanf("%d",&n);
  printf("Enter elements of an array:\n");
  for(i=0; i<n; i++)
   scanf("%d",&a[i]);
  for(i=1; i<n; i++)
   {
     temp=a[i];
     j=i-1;
     while(j>=n && a[j]>temp)
     {
       a[j+1]=a[j];
       j--;
     }
     a[j+1]=temp;
   }
  printf("sorted list in ascending oider:\n");
  for(i=0; i<n; i++)
   { printf("%d\t", a[i]);
   }
   return 0;
}
```