

## LAB PROGRAMS (Sorting)

***/\* A program for the Insertion sort algorithm. \*/***

```
#include <stdio.h>
int main()
{
    int n, a[100], i, j, temp;
    printf("Enter the number of elements : ");
    scanf("%d", &n);
    printf("Enter %d integers:\n", n);
    for (i=0;i<n;i++)
    {
        printf("Enter a[%d] element : ",i);
        scanf("%d",&a[i]);
    }

    for (i=1;i<=n-1;i++)
    {
        j = i;
        while (j>0 && a[j]<a[j-1])
        {
            temp = a[j];
            a[j] = a[j-1];
            a[j-1] = temp;
            j--;
        }
    }
    printf("The sorted list in ascending order:\n");
    for (i = 0; i <= n - 1; i++)
    {
        printf("%d\n", a[i]);
    }
    return 0;
}
```

### **Output:**

Enter the number of elements : 3

Enter 3 integers:

Enter a[0] element : 1

Enter a[1] element : 5

Enter a[2] element : 2

The sorted list in ascending order:

1

2

5

**/\* A program for the Selection sort algorithm. \*/**

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int n, a[100], i, j, temp, po;
```

```
    printf("Enter the number of elements : ");
```

```
    scanf("%d", &n);
```

```
    printf("Enter %d integers:\n", n);
```

```
    for (i=0;i<n;i++)
```

```
    {
```

```
        printf("Enter a[%d] element : ",i);
```

```
        scanf("%d",&a[i]);
```

```
    }
```

```
    for(i = 0; i < n - 1; i++)
```

```
    {
```

```
        po=i;
```

```
        for(j = i + 1; j < n; j++)
```

```
        {
```

```
            if(a[po] > a[j])
```

```
            po=j;
```

```
        }
```

```
        if(po != i)
```

```
        {
```

```
            temp=a[i];
```

```
            a[i]=a[po];
```

```
            a[po]=temp;
```

```
        }
```

```
    }
```

```
    printf("The sorted list in ascending order:\n");
```

```

    for (i = 0; i <= n - 1; i++)
    {
        printf("%d\n", a[i]);
    }
    return 0;
}

```

### **Output:**

```

Enter the number of elements : 3
Enter 3 integers:
Enter a[0] element : 1
Enter a[1] element : 4
Enter a[2] element : 3
The sorted list in ascending order:
1
3
4

```

### **/\*A program for the Bubble sort algorithm.\*/**

```

#include <stdio.h>
int main()
{
    int n, a[100], i, j, temp, po;
    printf("Enter the number of elements : ");
    scanf("%d", &n);
    printf("Enter %d integers:\n", n);
    for (i=0;i<n;i++)
    {
        printf("Enter a[%d] element : ",i);
        scanf("%d",&a[i]);
    }
    for(i = 0 ; i < n - 1; i++)
    {
        for(j = 0 ; j < n-i-1; j++)
        {
            if(a[j] > a[j+1])
            {

```

```

    temp=a[j];
    a[j]=a[j+1];
    a[j+1]=temp;
}
}
}
printf("The sorted list in ascending order:\n");
for (i = 0; i <= n - 1; i++)
{
    printf("%d\n", a[i]);
}
return 0;
}

```

### **Output:**

```

Enter the number of elements : 3
Enter 3 integers:
Enter a[0] element : 1
Enter a[1] element : 7
Enter a[2] element : 3
The sorted list in ascending order:
1
3
7

```

### ***/\*A program for the Merge sort algorithm.\*/***

```

#include<stdio.h>
void mergesort(int a[],int i,int j);
void merge(int a[],int i1,int j1,int i2,int j2);
int main()
{
    int a[100],n,i;
    printf("Enter the number of elements : ");
    scanf("%d",&n);
    printf("Enter array elements: \n");
    for(i=0;i<n;i++)
    {

```

```

        printf("Enter a[%d] element : ",i);
scanf("%d",&a[i]);
    }
    mergesort(a,0,n-1);
    printf("The sorted list in ascending order:\n");
    for(i=0;i<n;i++)
    {
        printf("%d\n",a[i]);
    }
    return 0;
}

```

```

void mergesort(int a[],int i,int j)
{
    int mid;

    if(i<j)
    {
        mid=(i+j)/2;
        mergesort(a,i,mid);
        mergesort(a,mid+1,j);
        merge(a,i,mid,mid+1,j);
    }
}

```

```

void merge(int a[],int i1,int j1,int i2,int j2)
{
    int temp[50];
    int i,j,k;
    i=i1;
    j=i2;
    k=0;
    while(i<=j1 && j<=j2)
    {
        if(a[i]<a[j])
            temp[k++]=a[i++];
        else
            temp[k++]=a[j++];
    }
}

```

```

        while(i<=j1)
            temp[k++]=a[i++];
        while(j<=j2)
            temp[k++]=a[j++];
        for(i=i1,j=0;i<=j2;i++,j++)
            a[i]=temp[j];
    }

```

### **Output:**

```

Enter the number of elements : 3
Enter array elements:
Enter a[0] element : 13
Enter a[1] element : 7
Enter a[2] element : 25
The sorted list in ascending order:
7
13
25

```

### ***/\*A program for the Heapsort algorithm.\*/***

```

#include<stdio.h>
void create(int []);
void down_adjust(int [],int);
void main()
{
    int heap[100],n,i,last,temp;
    printf("Enter the number of elements : ");
    scanf("%d",&n);
    printf("Enter %d integers:\n", n);
    for(i=1;i<=n;i++)
    {
        printf("Enter heap[%d] element : ",i);
        scanf("%d",&heap[i]);
    }
    heap[0]=n;
    create(heap);
    while(heap[0] > 1)

```

```

    {
        last=heap[0];
        temp=heap[1];
        heap[1]=heap[last];
        heap[last]=temp;
        heap[0]--;
        down_adjust(heap,1);
    }
    printf("The sorted list in ascending order:\n");
    for(i=1;i<=n;i++)
        printf("%d\n",heap[i]);
}
void create(int heap[])
{
    int i,n;
    n=heap[0];
    for(i=n/2;i>=1;i--)
        down_adjust(heap,i);
}
void down_adjust(int heap[],int i)
{
    int j,temp,n,flag=1;
    n=heap[0];
    while(2*i<=n && flag==1)
    {
        j=2*i;
        if(j+1<=n && heap[j+1] > heap[j])
            j=j+1;
        if(heap[i] > heap[j])
            flag=0;
        else
        {
            temp=heap[i];
            heap[i]=heap[j];
            heap[j]=temp;
            i=j;
        }
    }
}

```

**Output:**

Enter the number of elements : 3

Enter 3 integers:

Enter heap[1] element : 2

Enter heap[2] element : 8

Enter heap[3] element : 1

The sorted list in ascending order:

1

2

8