

Lab programs

1. Write a program for the Insertion sort algorithm.

```
#include <math.h>
```

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

```
void insertionSort(int arr[], int n)
```

```
{
```

```
    int i, key, j;
```

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

```
        key = arr[i];
```

```
        j = i - 1;
```

```
        while (j >= 0 && arr[j] > key) {
```

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

```
            j = j - 1;
```

```
        }
```

```
        arr[j + 1] = key;
```

```
    }
```

```
}
```

```
void printArray(int arr[], int n)
```

```
{
```

```
    int i;
```

```

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

        printf("%d ", arr[i]);

    printf("\n");
}

int main()

{

    int arr[] = { 20,12,65,45,46 };

    int n = sizeof(arr) / sizeof(arr[0]);

    insertionSort(arr, n);

    printArray(arr, n);

    return 0;

}

```

2. Write a program for the Selection sort algorithm.

```

#include <stdio.h>

void swap(int *xp, int *yp)

{

    int temp = *xp;

    *xp = *yp;

    *yp = temp;

}

void selectionSort(int arr[], int n)

{

    int i, j, min_idx;

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

```

```
{

    min_idx = i;

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

        if (arr[j] < arr[min_idx])

            min_idx = j;

    swap(&arr[min_idx], &arr[i]);

}

}

void printArray(int arr[], int size)

{

    int i;

    for (i=0; i < size; i++)

        printf("%d ", arr[i]);

    printf("\n");

}

int main()

{

    int arr[] = {20,12,65,45,46};

    int n = sizeof(arr)/sizeof(arr[0]);

    selectionSort(arr, n);

    printf("Sorted array: \n");

    printArray(arr, n);

    return 0; }
```

3. Write a program for the Bubble sort algorithm.

```
#include<stdio.h>

int main(){

    int count, temp, i, j, a[30];

    printf("How many numbers are u going to enter?: ");

    scanf("%d",&count);

    printf("Enter %d numbers: ",count);

    for(i=0;i<count;i++)

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

    for(i=count-2;i>=0;i--){

        for(j=0;j<=i;j++){

            if(a[j]>a[j+1]){

                temp=a[j];

                a[j]=a[j+1];

                a[j+1]=temp;

            }

        }

    }

    printf("Sorted elements: ");

    for(i=0;i<count;i++)

        printf(" %d",a[i]);

    return 0;

}
```

4. Write a program for the Merge sort algorithm.

```
#include <stdio.h>

#define max 10

int a[11] = {20,12,65,45,46,6,28,56,38,14,98 };

int b[10];

void merging(int low, int mid, int high) {

    int l1, l2, i;

    for(l1 = low, l2 = mid + 1, i = low; l1 <= mid && l2 <= high; i++) {

        if(a[l1] <= a[l2])

            b[i] = a[l1++];

        else

            b[i] = a[l2++];

    }

    while(l1 <= mid)

        b[i++] = a[l1++];

    while(l2 <= high)

        b[i++] = a[l2++];

    for(i = low; i <= high; i++)

        a[i] = b[i];

}

void sort(int low, int high) {

    int mid;

    if(low < high) {

        mid = (low + high) / 2;
```

```
    sort(low, mid);

    sort(mid+1, high);

    merging(low, mid, high);

} else {

    return;

}

}

int main() {

    int i;

    printf("List before sorting\n");

    for(i = 0; i <= max; i++)

        printf("%d ", a[i]);

    sort(0, max);

    printf("\nList after sorting\n");

    for(i = 0; i <= max; i++)

        printf("%d ", a[i]);

}
```