

# 1. Write a program for the Insertion sort algorithm.

```
#include<stdio.h>

int main() {

    int i, j, count, temp, number[50];

    printf("ENTER THE NUMBER OF ELEMENTS: ");

    scanf("%d",&count);

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


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

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

    for(i=1;i<count;i++){

        temp=number[i];

        j=i-1;

        while((temp<number[j]) && (j>=0)) {

            number[j+1]=number[j];

            j=j-1;

        }

        number[j+1]=temp;

    }

    printf("sorted elements order: ");

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

        printf(" %d",number[i]);
```

```
        return 0;
    }
}
```

## 2. Write a program for the Selection sort algorithm

```
#include<stdio.h>

int main(){

    int i, j, count, temp, number[25];

    printf("enter the number of elements: ");

    scanf("%d",&count);

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

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

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

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

        for(j=i+1;j<count;j++){

            if(number[i]>number[j]){

                temp=number[i];

                number[i]=number[j];

                number[j]=temp;

            }

        }

    }
}
```

```

    }

    printf("elements in sorted order: ");

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

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

    return 0;
}

```

### 3. Write a program for the Bubble sort algorithm.

```

#include<stdio.h>

int main(){

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

    printf("enter the number of elements: ");

    scanf("%d",&count);

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

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

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

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

```

```

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

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

        temp=number[j];

        number[j]=number[j+1];

        number[j+1]=temp;

    }

}

}

printf("order of sorted elements: ");

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

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


return 0;

}

```

#### 4. Write a program for the Merge sort algorithm.

```

#include <stdio.h>

#define max 10

int a[11] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 };

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]);

}
```

