Insertion sort:

```
#include<stdio.h>
void InsertionSort(int b[], int n)
{
    int x,y;
    int tmp;
    for(y = 1; y < n; y++)
    {
        tmp = b[y];
        for(x = y; x > 0 && b[x-1] > tmp; x--)
            b[x] = b[x-1];
        b[x] = tmp;
    }
}

int main()
{
    int i, n, b[10];
    printf("Enter the number of elements :: ");
    scanf("%d",&n);
    printf("Enter the elements :: ");
    for(i = 0; i < n; i++)
    {
        scanf("%d",&b[i]);
    }
    InsertionSort(b,n);
    printf("The sorted elements are ::  ");
    for(i = 0; i < n; i++)
        printf("%d  ",b[i]);
    printf("\n");
    return 0;
}
```

Selection sort:

```
#include <stdio.h>
int main()
{
  int b[50], m, e, f, position, s;
  printf("Enter number of elements\n");
  scanf("%d", &m);

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

  for (e = 0; e < m; e++)
    scanf("%d", &b[e]);

  for (e = 0; e < (m - 1); e++)
  {
    position = e;

    for (f = e + 1; e < m; f++)
    {
      if (b[position] > b[f])
        position = f;
    }
    if (position != e)
    {
      s = b[e];
      b[e] = b[position];
      b[position] = s;
    }
  }

  printf("Sorted list in ascending order:\n");

  for (e = 0; e < m; e++)
    printf("%d\n", b[e]);

  return 0;
}
```

Bubble sort:

```
#include <stdio.h>

int main()
{
  int a[50], m, x, y, swap;
  printf("Enter number of elements\n");
  scanf("%d", &m);

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

  for (x = 0; x < m; x++)
    scanf("%d", &a[x]);

  for (x = 0 ; x < m - 1; x++)
  {
    for (y = 0 ; y < m - x - 1; y++)
    {
      if (a[y] > a[y+1])
      {
        swap       = a[y];
        a[y]   = a[y+1];
        a[y+1] = swap;
      }
    }
  }

  printf("Sorted list in ascending order:\n");

  for (x = 0; x < m; x++)
     printf("%d\n", a[x]);

  return 0;
}
```

Merge sort:

```
#include <stdio.h>

#define max 10

int a[11] = { 10, 14, 19, 26, 27, 31, 33, 35, 42, 44, 0 };
int b[10];

void merging(int low, int mid, int high) {
   int l1, l2, j;
   for(l1 = low, l2 = mid + 1, j = low; l1 <= mid && l2 <= high; j++) {
      if(a[l1] <= a[l2])
         b[j] = a[l1++];
      else
         b[j] = a[l2++];
   }

   while(l1 <= mid)
      b[j++] = a[l1++];

   while(l2 <= high)
      b[j++] = a[l2++];

   for(j = low; j <= high; j++)
      a[j] = b[j];
}

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 j;

   printf("List before sorting\n");

   for(j = 0; j <= max; j++)
      printf("%d ", a[j]);

   sort(0, max);

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

```
  for(j = 0; j <= max; j++)
    printf("%d ", a[j]);
}
```