ASSIGNMENT NO:5 **DATE:26/09/2014**

PROGRAM TITLE: Sort an integer array of 10 elements in ascending order using a) bubble sort b) selection sort

PROGRAM ALGORITHM:

```
algo main()
{
      /*a and b are collection of elements*/
      initialise ch to zero
      input elements of a
      copy a to b
      do-while(ch not equal to four)
      {
            switch(ch)
            {
                  case one: display original array
                              break
                  case two: call selection_sort(arguments: b)
                              break
                  case three: call bubble_sort(arguments: b)
                              break
                  case four: stop program
                              break
                  default: print "Invalid Choice"
            }
      }
}
algo bubble_sort(parameters: b)
{
      initialise flag to one
      while(flag not equal to zero)
      {
            for(i=1 to 9)
            {
                  if(present element of b greater than next element of b)
                  {
                        swap both elements
                  }
            }
      }
      print b
}
algo selection_sort(parameters: b)
{
      initialise pos to zero
      for(i=1 to 10)
      {
            set min to ith element of b
            set pos equal to i
            for(j=i+1 to 10)
            {
                  if(present element of a less than min)
                  {
                    set min to present element
                    set pos to i
               }
          }
          swap element at pos with element at i
     }
     print b
}
```

PROGRAM CODE:

```
/*C Program to Sort an Array*/
#include <stdio.h>
#define ARRSIZE 10
int selection_sort(int a[]);
int bubble_sort(int a[]);
int main()
{
     int i,a[ARRSIZE]={0},b[ARRSIZE]={0},ch=0;

     /*Read Input*/
     printf("Enter 10 elements of the array\n");
     for(i=0;i<ARRSIZE;i++)
     {
          scanf("%d",&a[i]);
          b[i]=a[i];
     }

     /*Display Menu to the user*/
     do
     {
          printf("Menu:\n1)Display Original Array\n2)Perform Selection 
Sort\n3)Perform Bubble Sort\n4)Exit\n");
          scanf("%d",&ch);
          getchar();
          switch(ch)
          {
               case 1: printf("Original Array is:\n");
                         for(i=0;i<ARRSIZE;i++)
                         {
                              printf("%d\t",a[i]);
                         }
                         printf("\n");
                         break;
               case 2: selection_sort(b);
                         break;
               case 3: bubble_sort(b);
                         break;
               case 4: return 1;
                         break;
               default:printf("Invalid Choice\n");
          }
     }
     while(ch!=4);
     return 0;
```

```
}

/*Function to perform Selection Sort*/
int selection_sort(int a[])
{
     int i,j,min,tmp,pos=0;
     for(i=0;i<ARRSIZE;i++)
     {

          /*Initialise min to the subscript of the current element*/
          min=a[i];
          pos=i;
          for(j=i+1;j<ARRSIZE;j++)
          {

               /*Find smallest element between the positions i and
ARRSIZE*/
               if(a[j]<min)
               {
                    min=a[j];
                    pos=j;
               }
          }

          /*Swap smallest element with one in position i*/
          tmp=a[pos];
          a[pos]=a[i];
          a[i]=tmp;
     }
     printf("Array after Selection sort is:\n");
     for(i=0;i<ARRSIZE;i++)
     {
          printf("%d\t",a[i]);
     }
     printf("\n");
     return 0;
}

/*Function to perform Bubble Sort*/
int bubble_sort(int a[])
{

     /*Set flag to 1 to begin initial pass*/
     int i,flag=1,tmp;
     while(flag)
     {

          /*Set flag to 0 awaiting a possible swap*/
          flag=0;
          for(i=0;i<ARRSIZE-1;i++)
          {
               if(a[i]>a[i+1])
               {

                    /*Swap elements and then set flag to 1 to indicate
that a swap occured*/
                      tmp=a[i];
                      a[i]=a[i+1];
                      a[i+1]=tmp;
                      flag=1;
                }
           }
     }
     printf("Array after Bubble sort is:\n");
     for(i=0;i<ARRSIZE;i++)
     {
           printf("%d\t",a[i]);
     }
     printf("\n");
     return 0;
}
```

OUTPUT:

```
Enter 10 elements of the array
17 25 -9 0 -15 255 -985 48 241 10
Menu:
1)Display Original Array
2)Perform Selection Sort
3)Perform Bubble Sort
4)Exit
2
Array after Selection sort is:
-985 -15   -9    0     10   17   25    48   241  255
Menu:
1)Display Original Array
2)Perform Selection Sort
3)Perform Bubble Sort
4)Exit
1
Original Array is:
17   25    -9    0     -15  255  -985  48   241  10
Menu:
1)Display Original Array
2)Perform Selection Sort
3)Perform Bubble Sort
4)Exit
3
Array after Bubble sort is:
-985 -15   -9    0     10   17   25    48   241  255
Menu:
1)Display Original Array
2)Perform Selection Sort
3)Perform Bubble Sort
4)Exit
4
```

DISCUSSION:

This Program sorts integer arrays.