<code> c program

First approach

// below we have a simple C program for bubble sort

```
#include <stdio.h>

void bubbleSort(int arr[], int n)
{
   int i, j, temp;
   for(i = 0; i < n; i++)
   {
      for(j = 0; j < n-i-1; j++)
      {
         if( arr[j] > arr[j+1])
         {
            // swap the elements
            temp = arr[j];
            arr[j] = arr[j+1];
            arr[j+1] = temp;
         }
      }
   }

   // print the sorted array
   printf("Sorted Array: ");
   for(i = 0; i < n; i++)
   {
      printf("%d  ", arr[i]);
   }
}

int main()
```

```
{
   int arr[100], i, n, step, temp;
   // ask user for number of elements to be sorted
   printf("Enter the number of elements to be sorted: ");
   scanf("%d", &n);
   // input elements if the array
   for(i = 0; i < n; i++)
   {
      printf("Enter element no. %d: ", i+1);
      scanf("%d", &arr[i]);
   }
   // call the function bubbleSort
   bubbleSort(arr, n);

   return 0;
}
```

Optimize approach <code>

```
// below we have a simple C program for bubble sort
#include <stdio.h>

void bubbleSort(int arr[], int n)
{
   int i, j, temp, flag=0;
   for(i = 0; i < n; i++)
   {
      for(j = 0; j < n-i-1; j++)
      {
         // introducing a flag to monitor swapping
         if( arr[j] > arr[j+1])
         {
```

```
            // swap the elements
            temp = arr[j];
            arr[j] = arr[j+1];
            arr[j+1] = temp;
            // if swapping happens update flag to 1
            flag = 1;
          }
        }
        // if value of flag is zero after all the iterations of inner loop
        // then break out
        if(flag==0)
        {
           break;
        }
    }

    // print the sorted array
    printf("Sorted Array: ");
    for(i = 0; i < n; i++)
    {
        printf("%d  ", arr[i]);
    }
}

int main()
{
    int arr[100], i, n, step, temp;
    // ask user for number of elements to be sorted
    printf("Enter the number of elements to be sorted: ");
    scanf("%d", &n);
    // input elements if the array
```

```
    for(i = 0; i < n; i++)
  {
     printf("Enter element no. %d: ", i+1);
     scanf("%d", &arr[i]);
  }
  // call the function bubbleSort
  bubbleSort(arr, n);

  return 0;
}
```