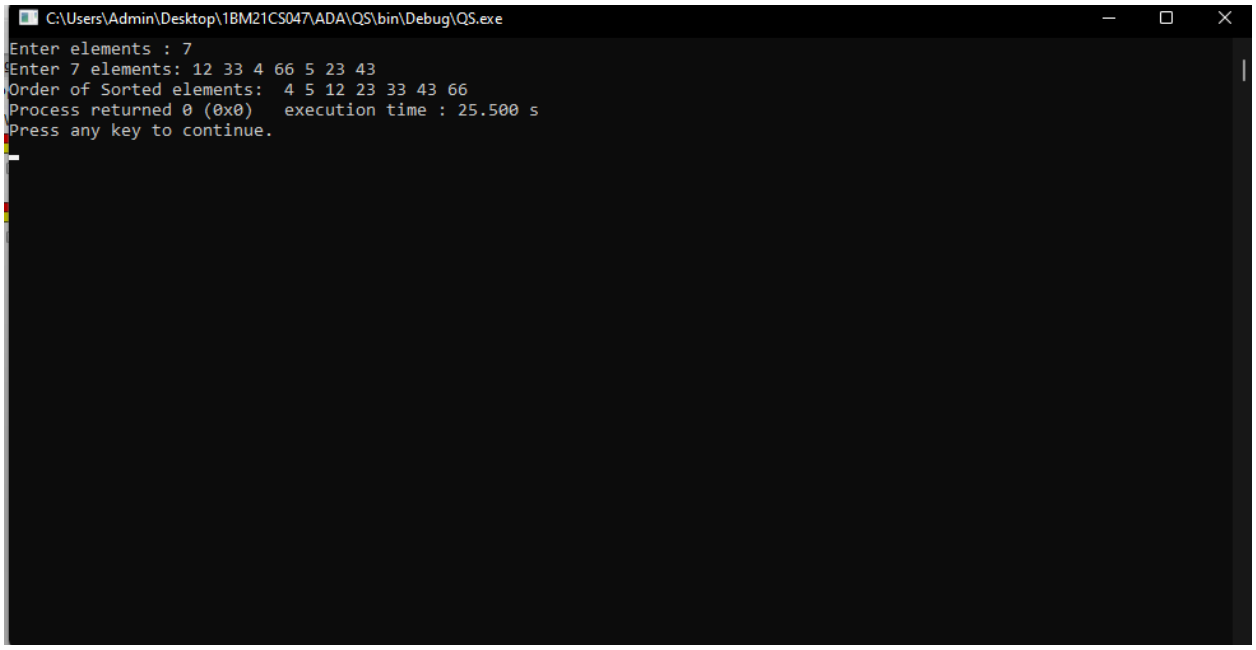
5.Sort a given set of N integer elements using Quick Sort technique and compute its time taken.

Code:
```
#include <stdio.h>
void quicksort(int number[25], int first, int last)
{
  int i, j, pivot, temp;
  if (first < last)
  {
    pivot = first;

    i = first;
    j = last;
    while (i < j)
    {
      while (number[i] <= number[pivot] && i < last)
        i++;
      while (number[j] > number[pivot])
        j--;
      if (i < j)
      {
        temp = number[i];
        number[i] = number[j];
        number[j] = temp;
      }
    }
    temp = number[pivot];
    number[pivot] = number[j];
    number[j] = temp;
    quicksort(number, first, j - 1);
    quicksort(number, j + 1, last);
  }
}
int main()
{
  int i, count, number[25];
  printf("Enter elements : ");
  scanf("%d", &count);
  printf("Enter %d elements: ", count);
  for (i = 0; i < count; i++)
    scanf("%d", &number[i]);
  quicksort(number, 0, count - 1);
  printf("Order of Sorted elements: ");
  for (i = 0; i < count; i++)
    printf(" %d", number[i]);
  return 0;
}
```

Output:



C:\Users\Admin\Desktop\1BM21CS047\ADA\QS\bin\Debug\QS.exe

```
Enter elements : 7
Enter 7 elements: 12 33 4 66 5 23 43
Order of Sorted elements:  4 5 12 23 33 43 66
Process returned 0 (0x0)   execution time : 25.500 s
Press any key to continue.
```

Observation:

20-07-23.

a) Sort a given set of N integers elements using Quick Sort method

```
#include<stdio.h>.
Void quicksort (int number[25], int first, int last)
{
   int i,j, pivot, temp;
   if (first <last)
   {
      pivot = first;
      i=first;
      j=last;
      while (i<j)
      {
         while (number[i] <= number[pivot] && i<last )
            i++;
         while (number [j] > number [pivot])
            j--;
         if (i<j)
         {
            temp = number[i];
            number[~~first~~i] = number[j];
            number [j] = temp;
         }
      }
      temp=number [pivot];
      number [pivot]= number [j];
      number [j] = temp;
      quicksort(number, first ,j-1);
      quicksort(number, j+1 ,last);
   }
}
void main() {
   int i, count, number[25];
   printf(" Enter elements: ");
   scanf("%d", &count );
   printf("\nEnter %d elements : ",count);
   for (i=0; i<count; i++)
      scanf("%d", & number [i]);
   quicksort(number, 0 , count-1);
   printf(" Ordered elements: ");
   for(i=0;i<count; i++)
      printf("%d" , number [i]);
}.
```

Output:

```
Enter elements : 4
Enter 4 elements : 20 11 15 7
Order of Sorted elements : 7 11 15 20.
```