LAB PROGRAM 5

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

SOURCE CODE

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>
void quicksort(int number[100000],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[100000];
  clock_t start,end;
  double timetaken;
  printf("Enter the size of the array: ");
  scanf("%d",&count);
  printf("\n");

  for(i=0;i<count;i++)
  {
     number[i]=rand();
  }
  printf("The given array is: ");
  for(i=0;i<count;i++)
  {
      printf(" %d",number[i]);
  }
  printf("\n");
  printf("\n");
  start=clock();
  quicksort(number,0,count-1);
  end=clock();
  printf("Sorted Array is: ");
  for(i=0;i<count;i++)
  printf(" %d",number[i]);
  timetaken=(double)(end-start);
  timetaken=(timetaken/CLOCKS_PER_SEC);
  printf("\n");
  printf("\n");
```

```
  printf("Time taken to sort an array of size %d is: %f seconds",count,timetaken);
  printf("\n");
  printf("\n");
  return 0;
}
```

OUTPUT SCREENSHOT