#include <stdio.h>  
#include <time.h>  
void swap(int \*p, int \*q)  
{  
    int temp = \*p;  
    \*p=\*q;  
    \*q = temp;  
    printf("Swapped %d and %d \n",\*p,\*q);  
}  
int partition(int numb[],int left, int right)  
{ printf("Enters partition\n");  
    int i,j, pivot;  
    printf("l= %d and r= %d\n",left,right);  
    pivot = numb[left];  
    printf("pivot%d\n",pivot);  
    i=left;  
    j= right+1;  
     printf("i= %d and j = %d\n",i,j);  
     while(i<j)  
     {  i=i+1;  
         while(numb[i]<pivot && i<right)  
         {    
          //printf("numb[i] %d\n",numb[i]);  
             i=i+1;  
             printf("i=%d\n",i);  
         }  
         printf("Exitted i\n");  
         j=j-1;  
         while(numb[j]>pivot && j>left)  
         {  
         //printf("numb[j] %d\n",numb[j]);  
             j=j-1;  
             printf("j=%d\n",j);  
               
         }  
          swap(&numb[i],&numb[j]);  
         //  
         //  
     }  
swap(&numb[i],&numb[j]);  
swap(&numb[left],&numb[j]);  
printf("Partition ends\n");  
return j;  
}  
  
void quicksort(int numb[], int left, int right)  
{ printf("Enters quicksort\n");  
    if(left < right)  
    {int p = partition(numb, left, right);  
     printf("returns p%d\n",p);  
    quicksort(numb, left, p-1);  
    quicksort(numb, p+1, right);  
    }  
}  
int main() {

    int n,a[300],k,r;  
  clock\_t start,end;  
  double totaltime;  
   
  printf("\nenter value of n");  
  scanf("%d",&n);  
  for(k=0;k<n;k++)  
  {  
    scanf("%d",&a[k]);  
      }  
   printf("\nunsorted numbers\n");  
for(k=0;k<n;k++)  
  printf("%d\n",a[k]);  
  
  start=clock();  
  quicksort(a,0,n-1);  
  end=clock();  
  
  
  printf("\nsorted numbers\n");  
  for(k=0;k<n;k++)  
  printf("%d\n",a[k]);  
  totaltime=(double)(end-start)/CLOCKS\_PER\_SEC;  
  printf("time taken is %e",totaltime);  
   
    return 0;  
}