QUICK SORT :

#include <stdio.h>

#include<time.h>

int a[20],n;

int partition(int [],int, int);

void quick\_sort(int [],int,int);

void swap(int\*,int\*);

int main()

{

int i;

clock\_t start, end;

double time\_taken;

printf("Enter the no. of elements:");

scanf("%d", &n);

printf("Enter the array elements:");

for (i = 0; i < n; i++) {

scanf("%d", &a[i]);

}

start = clock();

quick\_sort(a, 0, n - 1);

end = clock();

time\_taken = (double)(end - start) / CLOCKS\_PER\_SEC;

printf("Sorted array:");

for (i = 0; i < n; i++) {

printf("%d ", a[i]);

}

printf("\n");

printf("Time taken to sort: %f seconds\n", time\_taken);

return 0;

}

void swap(int \*a,int \*b){

int temp=\*a;

\*a=\*b;

\*b=temp;

}

void quick\_sort(int a[],int low,int high){

if(low<high){

int mid=partition(a,low,high);

quick\_sort(a,low,mid-1);

quick\_sort(a,mid+1,high);

}

}

int partition(int a[],int low,int high){

int pivot=a[low];

int i=low;

int j=high+1;

while(i<=j){

do{

i=i+1;

}while(a[i]<pivot && i<=high);

do{

j=j-1;

}while(a[j]>pivot && j>=low);

if(i<j){

swap(&a[i],&a[j]);

}

}

swap(&a[j],&a[low]);

return j;

}

OUTPUT:

Enter the no. of elements:10

Enter the array elements:96 53 26 78 12 63 85 12 06 95

Sorted array:6 12 12 26 53 63 78 85 95 96

Time taken to sort: 0.000002 seconds