ANAGHA ACHARYA

1BM19CS224

Sort a given set of elements using heap sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.

#include<stdio.h>

#include<stdlib.h>

#include<time.h>

int temp;

void heapify(int a[],int n,int i)

{

int largest,left,right;

largest = i;

left = 2\*i + 1;

right = 2\*i + 2;

if(left<n && a[left]>a[largest])

largest=left;

if(right<n && a[right]>a[largest])

largest = right;

if(largest != i)

{

temp = a[i];

a[i]= a[largest];

a[largest] = temp;

heapify(a,n, largest);

}

}

void heapsort(int a[],int n)

{

int i;

for(i=n/2-1;i>=0;i--)

heapify(a,n,i);

for(i=n-1;i>=0;i--)

{

temp=a[0];

a[0]=a[i];

a[i]=temp;

heapify(a, i, 0);

}

}

void main(){

int a[1000],i,n;

clock\_t start,end;

double time;

printf("Enter the number of elements: ");

scanf("%d",&n);

printf("The numbers are:\n");

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

{

a[i]=(int)rand()%10000;

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

}

start=clock();

heapsort(a,n);

end=clock();

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

printf("\nSorted array:\n");

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

{

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

}

printf("\nTime taken=%1f\n",time);

}



