Practical 1

19BCE248

DAA

Aim: To implement iterative and full recursive version of following sorting algorithms: Selection Sort, Insertion Sort and Bubble Sort.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| N(input size) | Bubble  Iterative | Bubble  Recursive | Selection  Iterative | Selection  Recursive | Insertion  Iterative | Insertion  Recursive | Quick  Sort |
| 20 | 0.000004 | 0.000003 | 0.000005 | 0.000004 | 0.000002 | 0.000003 | 0.000004 |
| 50 | 0.000010 | 0.000009 | 0.000008 | 0.000006 | 0.000001 | 0.000004 | 0. 000009 |
| 100 | 0.000037 | 0.000048 | 0.000022 | 0.000032 | 0.000003 | 0.000017 | 0.000014 |
| 1000 | 0.003091 | 0.003561 | 0.001456 | 0.001883 | 0.000013 | 0.001417 | 0.000176 |
| 5000 | 0.003091 | 0.088618 | 0.035715 | 0.043083 | 0.000060 | 0.028479 | 0.000701 |
| 10000 | 0.407882 | 0.320480 | 0.137629 | 0.156866 | 0.000123 | 0.110927 | 0.001676 |
| 50000 | 9.424185 | 9.388515 | 3.181516 | 3.630031 | 0.000627 | 2.102296 | 0.012376 |
| 100000 | 34.589463 | 36.853157 | 12.287979 | 14.455557 | 0.001081 | 8.041283 | 0.022479 |

Code:

Bubble Iterative:

#include <stdio.h>

#include <time.h>

#include<stdlib.h>

void main()

{

int x[]={100,1000,5000,10000,50000};

for (int k = 0;k < 5; k++) {

// int n;

// scanf("%d",&n);

int n=x[k];

int a[n];

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

a[i]=rand();

}

clock\_t begin = clock();

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

int ind=-1;

for (int j = 0; j < n-i-1; j++) {

if(a[j]>a[j+1]){

int temp = a[j];

a[j] = a[j+1];

a[j+1] = temp;

}

}

}

clock\_t end = clock();

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

printf("%d-->Time Spent:%lf\n",x[k],time\_spent);

}

}

Bubble Recursive:

#include <stdio.h>

#include <time.h>

#include<stdlib.h>

void rec(int \*a,int n);

void main()

{

int x[]={100,1000,5000,10000,50000};

for (int k = 0;k < 5; k++) {

// int n;

// scanf("%d",&n);

int n=x[k];

int a[n];

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

a[i]=rand();

}

clock\_t begin = clock();

rec(a,n);

clock\_t end = clock();

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

printf("%d-->Time Spent:%lf\n",x[k],time\_spent);

}

}

void rec(int \*a,int n){

if(n==1)

return;

for (int i = 0; i < n-1; i++) {

if(a[i]>a[i+1]){

int t=a[i];

a[i]=a[i+1];

a[i+1]=t;

}

}

rec(a,n-1);

}

Selection Iterative:

#include <stdio.h>

#include <time.h>

#include<stdlib.h>

void main()

{

int x[]={100,1000,5000,10000,50000};

for (int k = 0;k < 5; k++) {

// int n;

// scanf("%d",&n);

int n=x[k];

int a[n];

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

a[i]=rand();

}

clock\_t begin = clock();

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

int min=a[i],ind=i;

for (int j = i; j < n; j++) {

if(a[j]<min){

ind=j;

min=a[ind];

}

}

if(ind!=i){

int t=a[ind];

a[ind]=a[i];

a[i]=t;

}

}

clock\_t end = clock();

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

printf("%d-->Time Spent:%lf\n",x[k],time\_spent);

}

}

Selection Recursive:

#include <stdio.h>

#include <time.h>

#include<stdlib.h>

void rec(int \*a,int n);

void main()

{

int x[]={100,1000,5000,10000,50000};

for (int k = 0;k < 5; k++) {

// int n;

// scanf("%d",&n);

int n=x[k];

int a[n];

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

a[i]=rand();

}

clock\_t begin = clock();

rec(a,n);

clock\_t end = clock();

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

printf("%d-->Time Spent:%lf\n",x[k],time\_spent);

}

}

void rec(int \*a,int n){

if(n<=1)

return;

int mx=a[n-1],ind=n-1;

for (int i = 0; i < n-1; i++) {

if(a[i]>mx)

{

mx=a[i];

ind=i;

}

}

int t=a[n-1];

a[n-1]=a[ind];

a[ind]=t;

rec(a,n-1);

}

Insertion Iterative:

#include <stdio.h>

#include <time.h>

#include<stdlib.h>

void main()

{

int x[]={100,1000,5000,10000,50000};

for (int k = 0;k < 5; k++) {

// int n;

// scanf("%d",&n);

int n=x[k];

int a[n];

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

a[i]=rand();

}

clock\_t begin = clock();

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

int j=i-1;

while(j>=0 && a[j]>a[i]){

a[j+1]=a[j];

j--;

}

a[j+1]=a[i];

}

clock\_t end = clock();

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

printf("%d-->Time Spent:%lf\n",x[k],time\_spent);

}

}

Insertion Recursive:

#include <stdio.h>

#include <time.h>

#include<stdlib.h>

void rec(int \*a,int n);

void main()

{

int x[]={100,1000,5000,10000,50000};

for (int k = 0;k < 5; k++) {

// int n;

// scanf("%d",&n);

int n=x[k];

int a[n];

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

a[i]=rand();

}

clock\_t begin = clock();

rec(a,n);

clock\_t end = clock();

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

printf("%d-->Time Spent:%lf\n",x[k],time\_spent);

}

}

void rec(int \*a,int n){

if(n<=1)

return;

rec(a,n-1);

int t=a[n-1],j=n-2;

while(j>=0 && a[j]>t){

a[j+1]=a[j];

j--;

}

a[j+1]=t;

}