#include<iostream>

#include<cstdlib>

#include<sys/time.h>

using namespace std;

void swap(int \*x,int \*y)

{

int temp;

temp=\*x;

\*x=\*y;

\*y=temp;

}

void generate\_random(int a[],int n)

{

int i;

srand(time(0));

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

a[i]=rand()%1000;

}

int Partition(int a[10],int l,int h)

{

int i,j,p;

i=l;j=h+1; p=l;

do{

do{

i=i+1;

}while(a[i]<a[p]);

do{

j=j-1;

}while(a[j]>a[p]);

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

}while(i<=j);

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

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

return j;

}

int Quicksort(int a[10],int m,int n)

{

int s;

if(m<n+1)

{

s=Partition(a,m,n);

Quicksort(a,m,s-1);

Quicksort(a,s+1,n);

return \*a;

}

}

int main()

{

int a[100000],i,ch,n;

struct timeval t;

double start,end;

FILE \*fp;

printf("Enter the type of operation\n");

printf("1-Randomly generate numbers and quicksort\n");

printf("2-Enter the number of values to generate and sort\n");

scanf("%d",&ch);

switch(ch)

{

case 1:

fp=fopen("quicksort.txt","w");

for(i=10000;i<100000;i=i+5000)

{

generate\_random(a,i);

gettimeofday(&t,NULL);

start=t.tv\_sec+(t.tv\_usec/1000000.0);

Quicksort(a,0,i-1);

gettimeofday(&t,NULL);

end=t.tv\_sec+(t.tv\_usec/1000000.0);

printf("%d\t%lf\n",i,end-start);

fprintf(fp,"%d\t%lf\n",i,end-start);

}

fclose(fp); break;

case 2:

printf("Enter the number of values to be generated\n");

scanf("%d",&i);

generate\_random(a,i);

gettimeofday(&t,NULL);

start=t.tv\_sec+(t.tv\_usec/1000000.0);

Quicksort(a,0,i-1);

gettimeofday(&t,NULL);

end=t.tv\_sec+(t.tv\_usec/1000000.0);

printf("%d\t%lf\n",i,end-start);

printf("Sorted numbers are\n");

printf("The sorted array is\n");

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

printf("%d\t",a[n]);

break;

default: printf("Invalid choice\n");

}

return 0;

}

