机考测试报告

姓名：杨清含 学号：U202413716 题号：

【测试代码】

//杨清含 U202413716

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define SIZE 20

#define VASIZE 99

void setRandomArray(int \*a, const int size);

void insertSort(int a[], const int size);

void displayResults(int \*a, const int size, int \*foundNum);

void printArray(int a[], const int size);

int main()

{

srand(time(NULL));

int a[SIZE];

int foundNum = 0;

setRandomArray(a, SIZE);

printf("The random array is:\n");

/\*a[0] = 1;

a[2] = 3;

a[3] = 9;\*/

printArray(a, SIZE);

insertSort(a, SIZE);

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

printArray(a, SIZE);

displayResults(a, SIZE, &foundNum);

return 0;

}

void setRandomArray(int \*a, const int size)

{

int i;

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

{

a[i] = rand() % VASIZE + 1;

}

return;

}

void insertSort(int a[], const int size)

{

int insertValue, i, j, hold;

for(i = 1; i < size; i ++)

{

insertValue = a[i];

for(j = i - 1; j >= 0 && a[j] > insertValue; j --)

{

hold = a[j];

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

a[j + 1] = hold;

}

}

return;

}

void displayResults(int \*a, const int size, int \*foundNum)

{

int mem[50];

int current, i, j, k;

if(size > 50)

{

printf("Can't input such a big array!\n");

\*foundNum = -1;

return;

}

for(i = 0; i < size - 2; i++)

{

mem[0] = a[i];

for(j = i + 1; j < size - 1; j ++)

{

mem[1] = a[j];

current = 1;

for(k = j + 1; k < size; k ++)

{

if(a[k] \* mem[current - 1] == mem[current] \* mem[current])

{

current ++;

mem[current] = a[k];

}

}

if(current > 1)

{

//printf("In");

\*foundNum ++;

printf("%d\n", \*foundNum);

printArray(mem, current + 1);

printf("\n");

}

}

}

printf("Total find %d arrays.\n", \*foundNum);

return;

}

void printArray(int a[], const int size)

{

int i ;

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

{

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

if(i % 5 == 4)

{

printf("\n");

}

}

return;

}

【测试过程】

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 测试任务 | 测试方法 | 测试结果 | 测试结论 |
| 1 | 测试setRandomArray能否正常设置随机值 | 打印设置数组 |  | 测试通过 |
| 2 | 测试insertSort能否正常排序 | 打印排序后的随机数组，并于原数组对比 |  | 测试通过 |
| 3 | 测试displayResult能否正常运行 | 通过随机值 |  | 测试通过：考虑到多元素的情况，并对大于二的数组进行打印 |

【测试结论】

该题所有要求都完成