# 机考测试报告

|  |  |  |
| --- | --- | --- |
| 姓名：徐诚 | 学号：U202413383 | 题号：B03 |

【测试代码】

//U202413383

//Author:Xu Cheng

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

//生成随机数组

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

//打印排序前后数组

void displayArray ( int \*x, const int size );

//使用选择排序进行排序

void sortArraySelect ( int \*x, const int size );

//查找是否存在两数之和关系的组合并打印出来

void displaySumRelation ( int \*x, const int size, int \*foundNum );

int main( void )

{

srand(time(NULL));

const int size = 16;

int myArray[size];

int \*iptr = &myArray[0];

//设置初始数组

printf("Setting random array:\n");

setRandomArray( iptr, size );

displayArray( iptr, size );

//打印升序排列后数组

printf("\nAfter sorting by selection:\n");

sortArraySelect( iptr, size );

displayArray( iptr, size );

int total = 0;

int \*num = &total;

//查找符合两数之和关系的组合

printf("\nAll pairs meet the Sum Relation:\n");

displaySumRelation( iptr, size, num );

printf("Find total %d pairs",\*num);

return 0;

}

void displayArray ( int \*x, const int size )

{

int i;

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

{

printf("%4d",\*x);

x ++;

}

}

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

{

int i;

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

{

\*x = 1 + rand() % 99;

x ++;

}

}

void sortArraySelect ( int \*x, const int size )

{

int i, j;

int min\_index, min\_value;

for ( i = 0; i < size - 1; i ++ )

{

min\_index = i;

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

{

if ( \*(x + min\_index) > \*(x + j) )

{

min\_index = j;

}

}

if( min\_index != i )

{

min\_value = \*(x + min\_index);

\*(x + min\_index) = \*(x + i);

\*(x + i) = min\_value;

}

}

}

void displaySumRelation ( int \*x, const int size, int \*foundNum )

{

//定义能符合要求的变量组合

int i, j, k;

//i, j, k 分别遍历到最后三个数的对应位置

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

{

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

{

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

{

if ( (\*(x+i)) + (\*(x+j)) == (\*(x+k)) )

{

printf("{%d,%d,%d}\n", \*(x+i), \*(x+j), \*(x+k));

(\*foundNum) ++;

}

}

}

}

}

【测试过程】

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 测试任务 | 测试方法 | 测试结果（截图） | 测试结论 |
| 1 | 测试setRandomArray函数，考察其是否正确生成范围内随机数 |  |  | 测试通过 |
| 2 | 测试sortArraySelection函数，考察其是否正确升序排列 |  |  | 测试通过 |
| 3 | 测试displaySumRelation函数，考察其是否打印出所有符合要求的组合（未考虑数组中两数相等的情况，可能会重复） |  |  | 测试通过 |
|  |  |  |  |  |

【测试结论】

该题所有要求都完成