# 机考测试报告

姓名：李奇峰 学号：U202411282 题号：C02

【测试代码】

//U202411282 LiQiFeng

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define M 10

#define N 35

typedef struct{

char content[ N ] ;

int length ;

} String ;

void setRandomString( String \*s ) ;

void sortString( String \*sArray ) ;

void exchange( String \*s1 , String \*s2 ) ;

int main()

{

int i, j ;

srand( time( NULL)) ;

String string[ M ] = { 0 } ;

setRandomString( string ) ;

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

printf( " Length Content\n");

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

{

printf( " %-6d ", string[ i ].length ) ;

for( j = 0 ; j < string[ i ].length ; j ++ )

{

printf( "%d ", string[ i ].content[ j ] ) ;

}

printf( "\n") ;

}

sortString( string ) ;

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

printf( " Length Content\n");

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

{

printf( " %-6d ", string[ i ].length ) ;

for( j = 0 ; j < string[ i ].length ; j ++ )

{

printf( "%d ", string[ i ].content[ j ] ) ;

}

printf( "\n") ;

}

return 0;

}

void setRandomString( String \*s )

{

int i, j ;

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

{

s[ i ].length = rand() % ( N + 1 ) ;

for( j = 0 ; j < s[ i ].length ; j ++ )

{

s[ i ].content[ j ] = rand() % 9 + 1 ;

}

}

}

void sortString( String \*sArray )

{

int i, j, ans, k ;

char min;

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

{

k = 0 ;//k用于找出有多少个数小于或等于min

ans = i ;//ans用于判断是否有数小于min

min = sArray[ i ].content[ 0 ] ;

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

{

if( min > sArray[ j ].content[ 0 ] )

{

min = sArray[ j ].content[ 0 ] ;

ans = j ;

k = 1 ;//一旦min改变使k回复初始值

}

else if( min == sArray[ j ].content[ 0 ] )

{

k ++ ;//统计是否有多个

}

}

if( ans != i )//需要进行排序

{

if( k == 1 )//仅一个小于min

{

exchange( &sArray[ ans ] , &sArray[ i ]) ;

}

else//多个小于min，进行进一步排序

{

for( j = ans + 1 ; j < M ; j ++ )

{

//比较length

if( ( min == sArray[ j ].content[ 0 ] ) && ( sArray[ ans ].length > sArray[ j ].length ) )

{

ans = j ;

}

}

if( ans != i )

{

exchange( &sArray[ ans ] , &sArray[ i ]) ;

}

}

}

else//可能存在等于min的数

{

if( k != 0 )//存在等于min的数，进行进一步排序

{

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

{

//比较length

if( ( min == sArray[ j ].content[ 0 ] ) && ( sArray[ ans ].length > sArray[ j ].length ) )

{

ans = j ;

}

}

if( ans != i )

{

exchange( &sArray[ ans ] , &sArray[ i ]) ;

}

}

}

}

}

void exchange( String \*s1 , String \*s2 )

{

String s3 ;

s3 = \*s1 ;

\*s1 = \*s2 ;

\*s2 = s3 ;

}

【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setRandomString函数，考察是否实现随机生成结构体数组 | 运行考察结果 |  | 测试通过 |
| 2 | 测试sortString函数能否正常按照要求进行排序（使用选择排序） | 将由setRandomString函数得到的随机数组进行排序 |  | 测试通过 |
| 3 | 测试sortString函数能否正常按照要求对length为0的字符串进行排序（特例） | 将由setRandomString函数得到的随机数组进行排序 |  | 测试通过 |
|  |  |  |  |  |
|  |  |  |  |  |

【测试结论】

该题所有要求都完成