姓名：刘宏远 学号：U202413662 题号：C01

测试代码：

//name：刘宏远 U202413662 C01

#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 strCopy( char s1[], char s2[] );

int main()

{

srand( time(NULL) );

String s[M];

puts("Before sorting:");

setRandomString( s );

puts("");

puts("After sorting:");

sortString( s );

int i, j;

int counter = 1;

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

{

printf("[%02d]", counter );

for ( j = 0; j < ( \*( s + i ) ).length; j ++ )

{

printf("%c", ( \*( s + i ) ).content[j] );

}

counter ++;

puts("");

}

return 0;

}

void setRandomString( String \*s )

{

int i, j;

int counter = 1;

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

{

printf("[%02d]", counter );

( \*( s + i ) ).length = rand() % ( N + 1 );//set random length

for ( j = 0; j < ( \*( s + i ) ).length; j ++ )

{

( \*( s + i ) ).content[j] = rand() % ( 'z' - 'a' + 1 ) + 'a';//set random character(a-z)

printf("%c", ( \*( s + i ) ).content[j] );

}

counter ++;

printf("\n");

}

}

void sortString( String \*sArray )

{

int i, j;

int isSwapped;

String hold;

//sort by the first character value

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

{

isSwapped = 0;

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

{

if ( ( \*( sArray + j ) ).content[0] > ( \*( sArray + j + 1 ) ).content[0] )

{

hold.length = ( \*( sArray + j ) ).length;

strCopy( hold.content, ( \*( sArray + j ) ).content );

( \*( sArray + j ) ).length = ( \*( sArray + j + 1 ) ).length;

strCopy( ( sArray + j ) -> content, ( sArray + j + 1 ) -> content );

( \*( sArray + j + 1 ) ).length = hold.length;

strCopy( ( sArray + j + 1 ) -> content, hold.content );

isSwapped = 1;

}

}

if ( isSwapped == 0 )

{

break;

}

}

//sort two element by their "length" if their first character are the same

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

{

if ( ( sArray + i ) -> content[0] == ( sArray + i + 1 ) -> content[0] )

{

if ( ( sArray + i ) -> length > ( sArray + i + 1 ) -> length )

{

hold.length = ( \*( sArray + i ) ).length;

strCopy( hold.content, ( \*( sArray + i ) ).content );

( \*( sArray + i ) ).length = ( \*( sArray + i + 1 ) ).length;

strCopy( ( sArray + i ) -> content, ( sArray + i + 1 ) -> content );

( \*( sArray + i + 1 ) ).length = hold.length;

strCopy( ( sArray + i + 1 ) -> content, hold.content );

}

}

}

}

void strCopy( char s1[], char s2[] )

{

int i;

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

{

\*s1++ = \*s2++;

}

}

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void setRandomString( String \*s )函数 | 随机生成结构体数组并打印 |  | 测试通过 |
| 2 | 测试void sortString( String \*sArray )函数 | 对生成的结构体数组排序后打印看是否正确 |  | 测试通过 |
| 3 | 测试main函数 | 打印排序前后的结构体数组看是否正确 |  | 测试通过 |

测试结论：

该题所有要求都完成。