

清华大学本科生考试试题专用纸

考试课程 计算机语言与程序设计 ( 卷)      2014 年 1 月 14 日

姓名 \_\_\_\_\_ 班级 \_\_\_\_\_ 学号 \_\_\_\_\_

大题	题号	一	二	三	四	总分
	分数	30 分	30 分	30 分	10 分	
成绩						

试卷提交说明

1. 在考卷上填写你的姓名、学号；
2. 按要求编程，调试运行，将每道题的结果考屏，粘贴在考卷对应的题后；
3. 交卷时，提交一个压缩文件包，包括：
  - a) 文件包名  
学号+姓名，如：2013123456ZhangSan.rar
  - b) 文件包内容
    - ✧ 每道题的 c 工程文件夹；
    - ✧ 考卷。

**一、字符串插入 (30 分)**

编制一个程序，要求实现如下功能：

1. 主函数循环运行，当输入"quit"或"exit"时，程序结束运行。(5 分)

2. 函数 input() (10 分)

从键盘输入字符串 s 和字符串 t (假设，字符串长度都不超过 100 个字符)，从键盘输入整数 n，要求  $0 \leq n < \text{字符串 } s \text{ 的长度}$ ，否则提示重新输入 n，将 s、t、n 传回给主函数。

3. 函数 insert() (10 分)

(1) 接收主函数传递来的字符串 s、t 和整数 n，并将字符串 t 按照如下的要求插入到字符串 s 中去。

(2) 插入的具体要求只考虑下面的情况：

a) 将字符串 t 插入到字符串 s 第 n 个元素 s[n] 之后首次出现的空格元素 s[m] 之后；

b) 若字符串 s 的第 n 个元素 s[n] 之后无空格元素 s[m] 出现，请将字符串 t 插入到字符串 s 的最后；

例如：输入字符串 s 为 "I love apples", 字符串 t 为 "round\_" (注：输入的 round 后面包含了一个空格)，整数 n 为 3，则插入后的字符串 s 为 "I love round apples"

再如：输入字符串 s 为 "I love apples", 字符串 t 为 "\_very much" (注：输入的 very much 前面包含了一个空格)，整数 n 为 9，则插入后的字符串 s 为 "I love apples very much"

4. 在主函数中输出插入以后的字符串 s (5 分)

程序示例输出：

```
"E:\LLC-WS\教学目录\2013年秋季C语言程序设计\final-exam\T1\Debug\1.exe"
程序是否继续执行? 按quit或者exit程序退出: g

Please input string s:I love apples
Please input string t:round
Please input integer n:20
Please input integer n again:3

The new string s is:I love round apples
程序是否继续执行? 按quit或者exit程序退出: g

Please input string s:I love apples
Please input string t: very much
Please input integer n:9

The new string s is:I love apples very much
程序是否继续执行? 按quit或者exit程序退出: quit
Press any key to continue.
```

## 二、链表节点交换 (30 分)

编制一个程序, 要求实现如下功能:

### 1. 函数 create() (10 分)

从键盘循环输入整型数字, 一次一个, 按照输入顺序存储成无重复元素的单向链表, 输入的整数为 0 时认为输入结束, 并将头指针返回给主函数。每次输入一个整数之后, 都要显示出当前链表的全部记录数据。如果用户输入了重复的数据, 应当提示重新输入。

### 2. 函数 exchange() (10 分)

用户输入两个节点的序号 (起始位置为 1), 在链表中交换对应位置的两个节点 (注: 是交换节点而不是交换节点的内容)。如果用户输入的序号不合法, 应当提示重新输入。

### 3. 函数 print() (5 分)

按顺序输出当前链表的全部记录数据。

### 4. 编写主函数 (5 分)

主函数循环运行, 当且仅当输入 '@' 时, 程序结束运行。每次循环, 实现创建链表、交换节点的功能。交换节点前后各输出一次链表的结构。

程序示例输出:

```
"E:\LLC-WS\教学目录\2013年秋季C语言程序设计\final-exam\T2\Debug\2.exe"
请输入新节点的值: 2
当前链表内容为:
2->NULL
请输入新节点的值: 5
当前链表内容为:
2->5->NULL
请输入新节点的值: 8
当前链表内容为:
2->5->8->NULL
请输入新节点的值: 5
该节点值已经存在于链表中! 请输入新节点的值: 10
当前链表内容为:
2->5->8->10->NULL
请输入新节点的值: 12
当前链表内容为:
2->5->8->10->12->NULL
请输入新节点的值: 0

交换前链表内容为:
2->5->8->10->12->NULL

请输入要交换的两个节点的位置 (空格分隔): 2 10
输入位置不合法, 请重新输入: 2 5
交换完成!

交换后链表内容为:
2->12->8->10->5->NULL

按任意键继续, 退出请按'@':
```

### 三、动态矩阵处理。(30 分)

编制程序，实现如下功能：

#### 1. 函数 input() (6 分)

该函数首先从键盘输入整数矩阵的行数  $m$  与列数  $n$ ，根据输入的  $m$  与  $n$ ，动态生成整数矩阵  $a$ ，并依次输入矩阵  $a$  的各元素值。该函数要求能够将整数矩阵  $a$  返回主函数，并能将  $m$  与  $n$  的值传回。

#### 2. 函数 ComputeIndex() (8 分)

该函数能够计算矩阵  $a$  的魅力指数矩阵  $b$ ，并能返回给调用函数。对于矩阵  $a$  的每一个元素  $a_{ij}$ ，定义它的“魅力指数”  $b_{ij} = a_{i-1,j} + a_{i+1,j} + a_{i,j-1} + a_{i,j+1}$ ，即它的上、下、左、右四个邻居之和，如果某个位置不存在邻居，则该位置不参与求和计算。例如，输入的矩阵  $a$  如下图左边的矩阵所示，其对应的“魅力指数”矩阵如下图中间所示。

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

$$\begin{bmatrix} 6 & 9 & 8 \\ 13 & 20 & 17 \\ 12 & 21 & 14 \end{bmatrix}$$

$$\begin{bmatrix} 8 & 5 & 6 \\ 9 & 4 & 7 \\ 2 & 3 & 1 \end{bmatrix}$$

#### 3. 函数 MatrixSort() (8 分)

先调用 ComputeIndex() 函数计算矩阵  $a$  的每个元素的“魅力指数”，然后按“魅力指数”对矩阵  $a$  的所有元素进行排序。排序规则为，按“魅力指数”从左到右，从上到下的顺序降序排列，新得到的排序后的矩阵如上图右侧矩阵所示。

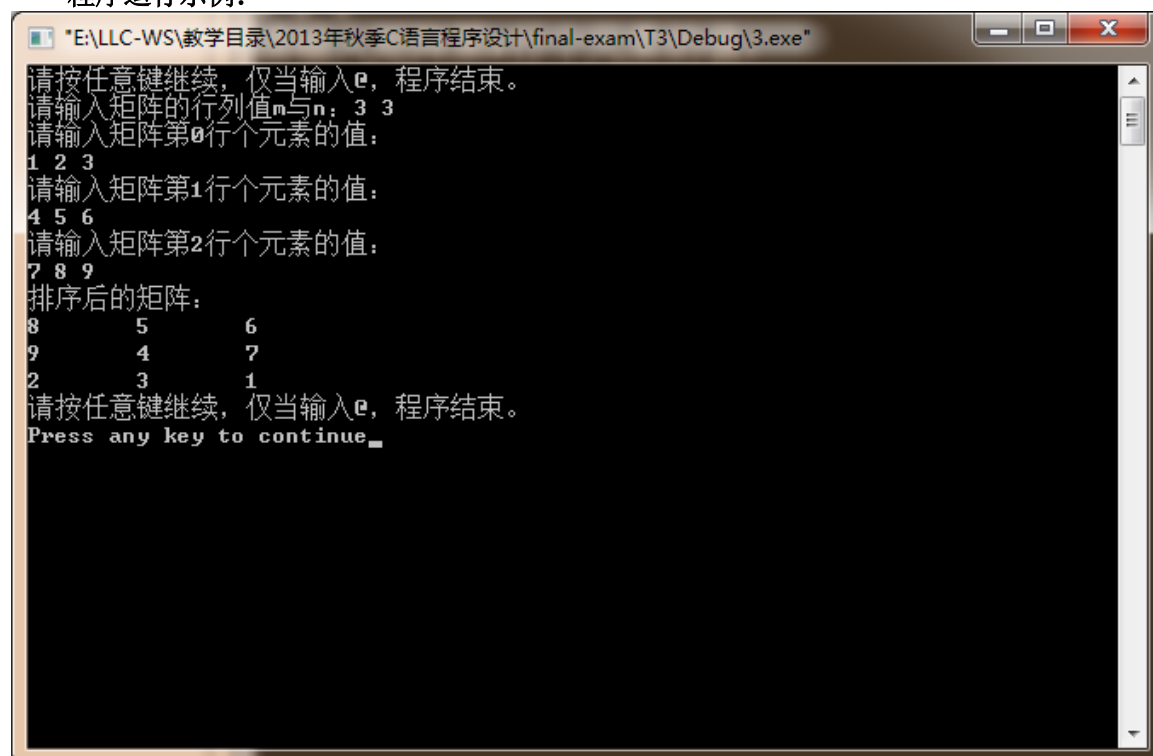
#### 4. 函数 print() (3 分)

按行依次输出  $m \times n$  的矩阵  $a$  的各个元素。

#### 5. 编写主函数 main() (5 分)

按照程序示例，主函数循环运行，当且仅当输入 '@' 时，程序结束运行。

程序运行示例：



```
"E:\LLC-WS\教学目录\2013年秋季C语言程序设计\final-exam\T3\Debug\3.exe"
请按任意键继续，仅当输入@，程序结束。
请输入矩阵的行列值m与n: 3 3
请输入矩阵第0行个元素的值:
1 2 3
请输入矩阵第1行个元素的值:
4 5 6
请输入矩阵第2行个元素的值:
7 8 9
排序后的矩阵:
8      5      6
9      4      7
2      3      1
请按任意键继续，仅当输入@，程序结束。
Press any key to continue_
```

#### 四、递归编程：字符消除。(10 分)

请按照要求编写程序，实现字符串消除。

字符消除，即对连续出现 3 次(含 3 次)以上的字符进行递归消除，例如，字符串“123332211”消除后的字符串为空字符串“”，字符串“123332221”消除后得到的字符串为“121”，它们的消除过程分别如下（其中，红色表示当前被消除的字符）：

(1) “123332211” -> “122211” -> “111” -> “”

(2) “123332221” -> “121”

按照要求，编写程序。

**1. 函数 input() (3 分)**

键盘输入待消除的字符串 str（长度不超过 20），并返回给主函数

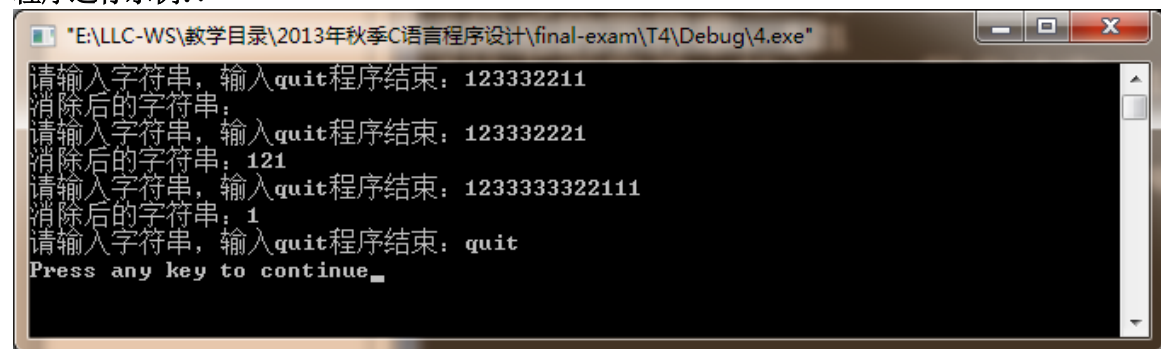
**2. 递归消除函数 Eliminate() (5 分)**

使用递归的方法，对字符串进行消除，并将消除的结果传回给主函数。

**3. 主函数 main() (2 分)**

主函数循环运行。调用 input() 输入字符串 str；若输入的字符串为“quit”时，程序执行结束；否则调用递归函数 Eliminate()，并在主程序中输出消除后的字符串。

程序运行示例：



```
"E:\LLC-WS\教学目录\2013年秋季C语言程序设计\final-exam\T4\Debug\4.exe"
请输入字符串，输入quit程序结束：123332211
消除后的字符串：
请输入字符串，输入quit程序结束：123332221
消除后的字符串：121
请输入字符串，输入quit程序结束：123333322111
消除后的字符串：1
请输入字符串，输入quit程序结束：quit
Press any key to continue_
```

//以上为全部题目。