

§ 6. 指针基础

【注意:】

- 1、本次作业不允许使用尚未讲授过的任何后续课程中的知识点, 包括但不限于结构体、类等相关概念!!!
- 2、除明确要求外, 已学过的知识中, **不允许**使用 goto、**不允许**使用全局变量 (不包括宏定义和只读变量), **不允许**使用 C++ 的 string 变量
- 3、cstdio、cmath、cstring 中的系统函数**可以**直接使用, 包括课上未介绍过的, 具体可自行查阅相关资料
- 4、除明确要求外, 所有 cpp 源程序不允许使用 scanf/printf 进行输入/输出
- 5、多编译器下均要做到 “0 errors, 0 warnings”
- 6、部分题目要求 C 和 C++ 两种方式实现, 具体见网页要求
- 7、输出为浮点数且未指定格式的, 均要求为 double 型, C++ 为 cout 缺省输出, C 为 %lf 的缺省输出
- 8、认真阅读格式要求及扣分说明!!!

【本次作业特别要求:】

- 1、本章所有作业都必须用指针 (即使出现数组, 也必须是指针法访问)
- 2、一维数组, 除定义变量外, 函数的形参及程序的执行语句不允许出现 [] 形式
- 3、指针在一维数组的元素间移动时, 要体现出指针变量自身的变化 (具体要求参考 060001 课件, 要求为类似 P. 61 的左侧形式, 如果写成右上或右下则得分为 0!!!)

【输出格式要求:】

- 1、为方便机器自动判断正确性, 作业有一定的输入输出格式要求 (但不同于竞赛的无任何提示)
 - 2、每个题目见具体说明, 必须按要求输入和输出, 不允许有偏差
 - 3、没有特别说明的情况下, 最后一行有效输出的最后有一个 endl
 - 4、本次作业的**比对要求**为 txt_compare 在 --trim right 下与 demo 做到**完全一致**
-
- 1、键盘输入一个字符串, 包括数字、非数字字符、空格等, 将其中连续的数字做为一个整数, 依次存放到一维整型数组中, 统计字符串中共有多少个整数, 并输出这些数
- 【要求:】
- 1、在给出的附件 6-b1.cpp 的基础上完成程序, 具体要求见附件
 - 2、读入含空格的字符串的方式不能是 cin (不能读空格), 不能是 gets_s (Dev 不支持), 方法去选做作业 050105 中找
 - 3、测试时字符串长度不超过 256, 不考虑负号, 不考虑包含的整数超过 int 上限
 - 4、如果字符串中包含的整数超过 10 个, 则只统计并输出前 10 个
 - 5、给出 6-b1-demo.exe 供参考

输出格式要求：多行

Line1: 输入提示

Line2: 键盘输入含数字字符串（允许带空格）

Line3: 共有**个整数

Line4: 依次输出的**个整数（空格分隔）

```
Microsoft Visual Studio 调试控制台
请输入间隔含有若干正负数字的字符串
hello -123w xyz k456.78 kjd 98
共有4个整数
123 456 78 98

D:\Workspace\高级语言程序设计\部分作业
按任意键关闭此窗口...
```

```
Microsoft Visual Studio 调试控制台
请输入间隔含有若干正负数字的字符串
a1b2c3d4e5f6g7h8i9j10k11
共有10个整数
1 2 3 4 5 6 7 8 9 10

D:\Workspace\高级语言程序设计\部分作业
按任意键关闭此窗口...
```

2、写一个函数，判断一个字符串是否是回文串（回文串：正反序输出一样），主程序要求从键盘输入字符串，再调用该函数判断。

例如：键盘输入 1234554321，输出：yes

键盘输入 abcdedcba，输出：yes

键盘输入 abcdeabcd，输出：no

【要求：】1、空串及长度为1的串都是回文串

2、字符串允许含空格

3、字符串允许含汉字，但按单字节比较即可（例：“大蟠” / “破” / “烫” / “铺唐”都是回文串，为什么？自己思考并研究）

4、读入含空格字符串的方式，本题要求 VS2022 和 DevC++均使用 fgets（要处理掉最后一个回车）

5、测试时人为保证输入字符串长度不超过 80

6、给出 6-b2-demo.exe 供参考

7、本题测试时如果需要重定向，仅使用输入重定向即可（不要用 get_input_data 方式的管道重定向，因为会过滤前后空格、单回车等）

8、**如果不加判断而直接输出 yes/no 则视为打表**

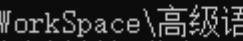
（例如：程序仅一句 `cout << "yes" << endl;` ），期末总分-20

输出格式要求：多行

Line1: 输入提示

Line2: 键盘输入字符串（允许带空格）

Line3: yes/no



Microsoft Visual Studio 调试控制台

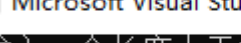
请输入一个长度小于80的字符串（回文串）

a

yes

D:\WorkSpace\高级语言程序设计\部分作业

按任意键关闭此窗口. . .



Microsoft Visual Studio 调试控制台

请输入一个长度小于80的字符串（回文串）

123 45 54 321

yes

D:\Workspace\高级语言程序设计\部分作业

按任意键关闭此窗口。

- 3、写一个函数将 2 进制整数转换为 10 进制，其中 2 进制在 main 中用从键盘上以字符串方式输入一串 0 和 1 的方式实现，转换后的 10 进制在 main 中以无符号整数方式输出

【要求:】 1、假设表示 2 进制的字符串输入长度不超过 32，不考虑输入错误

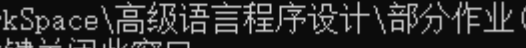
2、给出 6-b3-demo.exe 供参考

输出格式要求：多行

Line1: 输入提示

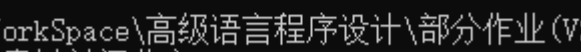
Line2: 键盘输入 0/1 字符串

Line3: 转换后的无符号十进制数



The screenshot shows the 'Microsoft Visual Studio 调试控制台' (Debug Console) window. It contains the following text:

```
请输入一个0/1组成的字符串，长度不超过32  
10010110010111  
9623  
  
D:\Workspace\高级语言程序设计\部分作业(VS2019)  
按任意键关闭此窗口. . .
```



Microsoft Visual Studio 调试控制台

请输入一个0/1组成的字符串，长度不超过32

11111111111111111111111111111111

4294967295

D:\Workspace\高级语言程序设计\部分作业(VS2019)

按任意键关闭此窗口. . .

4、用指针访问一维字符数组的方式实现下列函数

函数原型	功能说明	返回值
int tj_strlen(const char *str);	求字符串 str 的长度	字符串长度
char *tj_strcat(char *s1, const char *s2);	将字符串 s2 追加到 s1 后面, 含\0	新的 s1
Char *tj_strncat(char *s1, const char *s2, const int len);	将字符串 s2 的前 len 个字符追加到 s1 后面, 并添加\0 ★ 若 len 比 s2 的长度大, 则追加整个 s2 即可(含\0)	新的 s1
char *tj_strcpy(char *s1, const char *s2);	将字符串 s2 复制到 s1 中, 覆盖 s1 中原内容, 复制时包含\0	新的 s1
char *tj_strncpy(char *s1, const char *s2, const int len);	将字符串 s2 的前 len 个字符复制到 s1 中, 复制时不含\0 ★ 若 len 比 s2 的长度大, 复制 s2 长度个字符即可(不含\0)	新的 s1
int tj_strcmp(const char *s1, const char *s2);	比较字符串 s1 和 s2 的大小, 英文字母要区分大小写	相等为 0, 不等则为第 1 个不相等字符的 ASCII 差值
int tj_strcasecmp(const char *s1, const char *s2);	比较字符串 s1 和 s2 的大小, 英文字母不分大小写 ★ 例: tj_strcasecmp("abc", "ABZ"); tj_strcasecmp("ABC", "abz"); 均返回 -23 ★ 例: tj_strcasecmp("abZ", "AB["); tj_strcasecmp("abz", "AB["); 均返回 31	相等为 0, 不等则为第 1 个不相等字符的 ASCII 差值 ★ 若不相等处字符是不同的大小写字母, 则统一转换为小写后比较 ★ 若不相等处是大写字母和其它字符, 则返回对应小写字母和其它字符的差值
int tj_strncmp(const char *s1, const char *s2, const int len);	比较字符串 s1 和 s2 的前 len 个字符的大小, 英文字母要区分大小写 ★ 若 len 大于 s1/s2 中长度短的串, 则比较到短串的\0 即结束	相等为 0, 不等则为第 1 个不相等字符的 ASCII 差值
int tj_strcasencmp(const char *s1, const char *s2, const int len);	比较字符串 s1 和 s2 的前 len 个字符的大小, 英文字母不分大小写 ★ 长度要求同 tj_strncmp ★ 大小写要求同 tj_strcasecmp	相等为 0, 不等则为第 1 个不相等字符的 ASCII 差值

<code>char *tj_strupr(char *str);</code>	将字符串 str 中所有小写字母均转为大写，其它字符不变，转换后放在原串中	转换后的串
<code>char *tj_strlwr(char *str);</code>	将字符串 str 中所有大写字母均转为小写，其它字符不变，转换后放在原串中	转换后的串
<code>int tj_strchr(const char *str, const char ch);</code>	在字符串 str 中寻找字符 ch 第 1 次出现的位置，顺序是从左到右	找到：返回 1-n(位置从 1 开始)，未找到则返回 0
<code>int tj_strstr(const char *str, const char *substr);</code>	在字符串 str 中寻找字符串 substr 第 1 次出现的位置，顺序是从左到右	找到：返回 1-n(位置从 1 开始)，未找到则返回 0
<code>int tj_strrchr(const char *str, const char ch);</code>	在字符串 str 中寻找字符 ch 第 1 次出现的位置，顺序是从右到左 ★ 例：tj_strrchr("abcdab", 'a') 返回 5	找到：返回 1-n(位置从 1 开始)，未找到则返回 0
<code>int tj_strrstr(const char *str, const char *substr);</code>	在字符串 str 中寻找字符串 substr 第 1 次出现的位置，顺序是从右到左 ★ 例：tj_strrstr("abcdab", "ab") 返回 5	找到：返回 1-n(位置从 1 开始)，未找到则返回 0
<code>char *tj_strrev(char *str);</code>	字符串反转，放入原串中	反转后的串

- 本题要求：**
- 1、**不允许**使用任何系统函数（strlen、strcpy 等），**不允许**使用 C++ 的 string 类
 - 2、可以用自己定义的函数（例如在其它 tj_**函数中调用 tj_strlen）
 - 3、函数实现时不必考虑空间不够的情况（空间由调用函数保证）
 - 4、给出 6-b4.h、6-b4-main.cpp、6-b4-sub.cpp 三个文件共同形成一个可执行文件，6-b4.h 用于函数声明，6-b4-main.cpp 是测试用例，这两个文件**不准改动，无需提交**；列表中所有函数的具体实现均在 6-b4-sub.cpp 中，每个函数实现时有具体要求，必须按要求实现，提交时只需提交此文件即可
 - 5、要求使用指针的自增/相减/比较等运算，**不能**简单地将 str[i] 改为 *(str+i) 的形式，否则**本题得分为 0**
（具体要求参考 060001 课件，要求为 P. 61 的左侧形式，如果写成右上或右下则得分为 0!!!）
 - 6、整个程序中**不准**用任何形式的全局变量及静态局部变量；在上表列出的所有函数中，均**不准**定义任何数组形式的自动变量
 - 7、**再次提醒，指针为 NULL 和空字符串是两个完全不同的概念，因此对 NULL 有特殊要求（非标），详见 main 函数**
 - 8、针对 tj_strcmp/tj_strcasecmp/tj_strncmp/tj_strcasncmp 这四个比较函数，如果 s1/s2 中一个为 NULL 一个为非 NULL，则返回 1 或 -1 即可

【编译器要求:】

		编译器VS	编译器Dev
6-b1. cpp	字符串中取整数	Y	Y
6-b2. cpp	判断回文串	Y	Y
6-b3. cpp	二进制转十进制	Y	Y
6-b4-sub. cpp	一组字符串处理函数	Y	Y

【作业要求:】

- 1、**5月28日前**网上提交本次作业
- 2、每题所占平时成绩的具体分值见网页
- 3、超过截止时间提交作业会自动扣除相应的分数，具体见网页上的说明