

【注意:】

- 1、每题均有知识点使用限制, 认真阅读, 注意合规
- 2、不考虑输入错误
- 3、首行及格式缩进要求同之前作业(首行-10%, 缩进-10%)
- 4、要求 Dev 下 “0 errors, 0 warnings”
- 5、不要忘记首行信息

【本作业特别要求:】

- 1、所有题目都必须用指针(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
 - 整个函数, 不能直接用 *(数组名+int)/(指针+int) 来规避不允许出现 [] 形式的要求, 至少要有指针的++/--
 - 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!!!

§ 6. 指针基础



6.4. 字符串与指针

6.4.4. 下标法与指针法处理字符串

例: 将str1的内容复制给str2

```
int main()
{ char str1[]="I Love CHINA!", str2[20];
  char *p1, *p2;
  p1=str1;
  p2=str2;
  for(; *p1!='\0'; p1++, p2++)
    *p2=*p1;
  *p2='\0';
  p1=str1;
  p2=str2;
  cout << "str1:" << p1 << endl
        << "str2:" << p2 << endl;
  return 0;
}
```



例: 将str1的内容复制给str2

```
int main()
{ char str1[]="I Love CHINA!", str2[20];
  int i;
  for(i=0; str1[i]!='\0'; i++)
    str2[i] = str1[i];
  str2[i]='\0';
  cout << str1 << endl;
  cout << str2 << endl;
  return 0;
}
```



数组法实现

例: 将str1的内容复制给str2

```
int main()
{ char str1[]="I Love CHINA!", str2[20];
  int i;
  for(i=0; *(str1+i]!='\0'; i++)
    *(str2+i) = *(str1+i);
  *(str2+i]='\0';
  cout << str1 << endl;
  cout << str2 << endl;
  return 0;
}
```



等价指针法

1、一维数组与指针

题目描述：键盘输入若干非负整数到一个数组中（个数不定，以-1 结束，超过 2000 个数据则只取前 2000 个），将其调整为按个位数大小（0-9）的顺序排列。

输入输出格式要求：多行

Line1：输入提示：“请输入数组各元素,以-1 结束(不含-1 最多 2000 个数)”

Line2：键盘输入的若干数字（-1 结束，**不考虑输入错误**，允许加回车的多行形式）

Line3：输出提示：“数组的内容为”

Line4-x：若干行输出，每行 15 个元素

Linex+1：输出提示：“调整后的数组内容为”

Linex+2：若干行输出，每行 10 个元素

```
Microsoft Visual Studio 调试控制台
请输入数组各元素,以-1结束(不含-1最多2000个数)
3 103 123 97 101 87 72 88 74 61 65 2 30 17 19 16 9 9 5 43 2 9 -1
数组的内容为
3 103 123 97 101 87 72 88 74 61 65 2 30 17 19
16 9 9 5 43 2 9
调整后数组的内容为
30 61 101 2 2 72 43 3 123 103
74 65 5 16 97 17 87 88 9 9
19 9
```

- ① **不要忘记首行信息，检查是否正确的方法见⑥，不是 txt_compare**
- ② 本题允许使用的知识点：一维数组与指针（含之前的程序基本结构、函数、数组等知识）
- ③ 给出 w14-cl-s1-demo.exe 供参考，如果题目要求、截图、demo.exe 不一致，以 demo 为准
- ④ 给出基准程序 w14-cl-s1.cpp，按基准程序的限制要求完成即可（**违规 0 分**）
- ⑤ 生成测试数据的方法：w14-cl-s1-demo.exe --makedat 数量[1..2100]，会生成 data.txt（注意：makedat 前是两个减号）
- ⑥ 检查是否正确的方法（**不是 txt_compare**）：
w14-cl-s1.exe(你的) <data.txt >myout.txt（从 data.txt 中读，输出写入 myout.txt 中）
w14-cl-s1-demo.exe --check myout.txt（两个减号，myout.txt 为输出重定向文件）

```
D:\VS-Debug>w14-cl-s1-demo.exe --makedat 1234
已生成文件data.txt，共1234个有效数字。

D:\VS-Debug>w14-cl-s1-demo.exe < data.txt > myout.txt

D:\VS-Debug>w14-cl-s1-demo.exe --check myout.txt
检查原始数组：通过
检查调整后数组：通过
检查结束
```

生成数据和检查都用demo
红框内替换为你的exe

注：必测有效数据超过 2000 的情况（即--makedat 2001~2100 等情况下要正确）

2、字符串与指针

题目描述：键盘输入长度为 n 的字符串，再输入 m 值，表示将每个字符依次后移 m 个位置，后面的字符则回绕到前面

输入输出格式要求：多行

Line1: 输入提示：“请输入长度不超过 80 的字符串”

Line2: 键盘输入的字符串（图形字符、汉字、空格为有效，回车结束，**超过 80 只取前 80**）

Line3: 输出：“原始：*** ”

Line4: 输入提示：“请输入向后移动的位置 m $[1..2^{31}-1]$ ”

Line5: 键盘输入的 m 值（不考虑输入错误，允许超过串长）

Line6: 输出提示：“移动：*** ”

```
Microsoft Visual Studio 调试控制台
请输入长度不超过80的字符串
Welcome to Tongji University!
原始: Welcome to Tongji University!*
请输入向后移动的位置m[1..2^31-1]
11
移动: University!Welcome to Tongji *
```

```
Microsoft Visual Studio 调试控制台
请输入长度不超过80的字符串
同济大学
原始: 同济大学*
请输入向后移动的位置m[1..2^31-1]
8000006
移动: 济大学同*
```

```
Microsoft Visual Studio 调试控制台
请输入长度不超过80的字符串
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ
原始: ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHI*
请输入向后移动的位置m[1..2^31-1]
8
移动: ABCDEFGHABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz0123456789*
```

```
Microsoft Visual Studio 调试控制台
请输入长度不超过80的字符串
同济大学
原始: 同济大学*
请输入向后移动的位置m[1..2^31-1]
3
移动: 苟    么*  没毛病，是对的
```

思考：汉字移动奇数位置为什么会乱？

- ① **不要忘记首行信息，检查是否正确的方法为 txt_compare 与 demo 做到完全一致**
- ② 本题允许使用的知识点：一维字符数组与指针（含之前的程序基本结构、函数、数组等知识）
- ③ 给出 w14-c1-s2-demo.exe 供参考，如果题目要求、截图、demo.exe 不一致，以 demo 为准
- ④ 本题要求两种方法完成

方法 1：移动时用数组做临时交换变量（对应基准程序为 w14-c1-s2-1.cpp，**C++方式**）

方法 2：移动时用一个简单变量做临时交换变量（对应基准程序为 w14-c1-s2-1.c，**C方式**）

要求：1、按基准程序的限制要求完成即可（**违规 0 分**）

2、不允许添加头文件，则 C 方式下意味着不能用 strlen 等函数

3、基准程序中提供了标尺供大家调试时使用，提交时记得**注释掉或者删掉!!!**

【编译器要求：】

		编译器VS	编译器Dev
w14-c1-s1.cpp	数组元素调整	/	Y
w14-c1-s2-1.cpp	字符串循环移动(C++)	/	Y
w14-c1-s2-2.c	字符串循环移动(C)	/	Y

【提交要求:】

- 1、5月31日11:34:59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

【重要提醒:】

实验课作业考察快速完成度，提交开放后，**先将**已完成的题目提交，**不要**最后集中提交，避免各种问题导致本次分数全部丢失。