

【注意:】

- 1、本次作业不允许使用后续课程的知识点，包括但不限于指针、引用、结构体、类等概念!!!
- 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，不允许使用 C++ 的 string 类
- 3、cstdio 及 cmath 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!!!

【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

补充:

11、用函数+数组方式重新完成 3-b6/4-b1(人民币转大写)，要求如下：

- 【要求:】
- 1、所有的大写数字均放在全局一维只读字符数组 chnstr 中，具体形式为 const char chnstr[]="零壹贰叁肆伍陆柒捌玖";
凡需输出“零-玖”的地方，只允许从此数组中取值
 - 2、其它大写内容（拾佰仟万亿圆角分整）可自行取值
 - 3、转换后的内容不允许逐次输出，最后的输出只允许用一句输出语句来完成，具体分为两个小题
 3. 1 5-b11-1.c : 用全局 char result[256]; 存放转换结果，用 printf("%s\n", result); 输出结果
 3. 2 5-b11-2.cpp : 用全局 string result; 存放转换结果，用 cout << result << endl; 输出结果（本小题允许 string）
- (注：输入提示、错误提示等个性化输出允许自行按需组织，但输出大写转换结果的语句只能是一句)

- 【提示:】
- 1、根据分解的各位数字从 chnstr 中取部分内容
 - 2、各位数字要输出的内容依次放入 result 中，最后输出这个字符串即可
 - 3、保证 3-b12 中的所有测试数据均通过，输入输出格式要求同 3-b12（不考虑输入错误）
 - 4、如果在 Dev C++ 中有某个警告消除不掉，允许将全局一维只读数组改为 const char chnstr[]="零壹贰叁肆伍陆柒捌玖拾"（未碰到则忽略此提示即可）

12、用一维字符数组方式实现下列函数

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
int tj_strcat(char s1[], const char s2[]);	将字符串 s2 追加到 s1 后面, 含\0	0
int 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)	0
int 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	0
int 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)	0
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 tjstrcasecmp(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: tjstrcasecmp("abc", "ABZ"); tjstrcasecmp("ABC", "abz"); 均返回 -23 ★ 例: tjstrcasecmp("abZ", "AB["); tjstrcasecmp("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 tjstrcasecmp	相等为 0, 不等则为第 1 个不 相等字符的 ASCII 差值

int tj_strupr(char str[]);	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	0
int tj_strlwr(char str[]);	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	0
int tj_strchr(const char str[], const char ch);	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
int tj strstr(const char str[], const char substr[]);	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
int tj strrchr(const char str[], const char ch);	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
int tj strrstr(const char str[], const char substr[]);	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
int tj_strrev(char str[]);	字符串反转, 放入原串中	0

- 【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, 不允许借助指针, 不允许定义全局变量
 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 5-b12.h、5-b12-main.cpp、5-b12-sub.cpp 三个文件共同形成一个可执行文件, 5-b12.h 用于函数声明, 5-b12-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 5-b12-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、受限于目前所学知识, 部分函数的返回类型、实现的具体要求与系统的 str*** 函数不同, 注意按文档要求实现

【编译器要求:】

		编译器VS	编译器Dev
5-b11-1.c	人民币大写(char数组, 一句输出, C方式)	Y	Y
5-b11-2.cpp	人民币大写(string类, 一句输出)	Y	Y
5-b12-sub.cpp	一组字符串处理函数	Y	Y

【作业要求:】

- 1、**11月22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
- 4、另：本周还有**综合题**下发，请自行合理安排时间