

《计算机程序设计》作业 No-07及第6次上机

作业内容要点： 函数

【姓名_____学号_____】

【要求】

- (一)在计算机上编程序，加上必要的注释。
- (二)上机实验，经助教检查通过后，复制源码并记录实验结果，完成报告。
- (三)实验报告：记录调试及改错过程；知识点或方法技巧的收获心得。



1、合法括号序列

一个字符串里含有一些小括号，包括左括号‘(’和右括号’)’。合法的序列形式中左、右括号的数量是相同的，每一对匹配的括号中总是左括号在前右括号在后，自左向右读取的时候不会出现未匹配的右括号。

请编写一个函数判断字符串中的括号是否是合法的序列，函数原型如下：

```
int goodbrackets( char str[]);
```

当str是合法括号序列时函数返回1，否则返回0；

程序main函数中输入带括号的字符串，调用函数进行判断，根据函数返回值，在main函数中输出true(表示合法) 和false（表示非法）。(只考虑小括号字符构成的序列是否合法，其他字符可以忽略不计)。

测试样例：

输入：a(b) (c)

输出：true

测试样例：

输入：(a()())

输出：false

测试样例：

输入：(x)(y)(z)

输出: **false**

(一) 【源码】

```
//  
// main.c  
// 120701  
//  
// Created by 李佩哲 on 2021/11/10.  
//  
  
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
  
int main() {  
    int goodbrackets(char x[]);  
    char str[65535];  
    gets(str);  
    if(goodbrackets(str) == 1) printf("true\n");  
    else printf("false\n");  
    return 0;  
}  
  
int goodbrackets(char x[]) {  
    int boolean = 0, l = 0;  
    unsigned long n = strlen(x);  
    int num[n + 1];  
    for(int i = 0; i < n; i++) num[i] = x[i];  
    for (unsigned long i = 0; i < n; i++) {  
        if (x[i] == 40) l++;  
        else if (x[i] == 41) l--;  
        if(l<0)break;  
    }  
    if(l == 0) boolean = 1;  
    return boolean;  
}
```

(二) 【运行结果】

```
warning: this program uses gets(), which is unsafe.  
a( (b) (c) )  
true  
Program ended with exit code: 0  
warning: this program uses gets(), which is unsafe.  
(x)( (y)(z)  
false  
Program ended with exit code: 0
```

(三) 【实验报告】

用计数的方法来统计括号个数

2、求转置矩阵（用函数实现）

在主函数中对一个5x5大小的二维数组初始化，表示一个5x5的方阵，如下：

1	2	3	4	5
1	2	3	4	5
1	2	3	4	5
1	2	3	4	5
1	2	3	4	5

定义函数实现矩阵转置， **void transpose(int matrix[][5]);**

并在主函数中调用，将上述矩阵进行转置，输出转置之后的形式。

（一）【源码】

```
//  
// main.c  
// 120702  
//  
// Created by 李佩哲 on 2021/11/10.  
//  
  
#include <stdio.h>  
#include <stdlib.h>  
  
int tpmatrix[5][5];  
  
int main(){  
    void transpose(int pmatrix[5][5]);  
    int pmatrix[5][5] = {  
        {1, 2, 3, 4, 5},  
        {1, 2, 3, 4, 5},  
        {1, 2, 3, 4, 5},  
        {1, 2, 3, 4, 5},  
        {1, 2, 3, 4, 5}  
    };  
    transpose(pmatrix);  
    for(int i = 0; i < 5; i++) {  
        for(int j = 0; j < 5; j++)
```

```

        printf("%d ", tpmatrix[i][j]);
        printf("\n");
    }
    return 0;
}

```

```

void transpose(int pmatrix[5][5]) {
    for(int i = 0; i < 5; i++)
        for(int j = 0; j < 5; j++)
            tpmatrix[i][j] = pmatrix[j][i];
}

```

(二) 【运行结果】

```

1 1 1 1 1
2 2 2 2 2
3 3 3 3 3
4 4 4 4 4
5 5 5 5 5
Program ended with exit code: 0

```

(三) 【实验报告】

与上次求矩阵乘积的方法一样

3. 统计单词个数

我们把单词定义为由连续英文字母构成的字符串。

编写程序输入一行字符串，长度不超过100个字符。编写函数对这个字符串进行统计，返回单词的个数。

函数原型： **int wordcounter(char str[]);**

在main函数中输出单词个数。

样例输入：

I love China!

样例输出： 3

样例输入：

-----hi!!! how are you -----

样例输出： 4

(四) 【源码】

```
//
// main.c
// 120703
//
// Created by 李佩哲 on 2021/11/10.
//

// Code file created by C Code Develop

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main(){
    int wordcounter(char str[]);
    char str[65535];
    gets(str);
    int i = wordcounter(str);
    printf("%d\n", i);
    return 0;
}

int wordcounter(char str[]) {
    int i = 0;
    unsigned long r = 0;
    r = strlen(str);
    int num[r+2];
    for (int t = 0; t < r; t++) num[t+1] = str[t]; //给第一个单词留下判定
    //定点，因为下面的判断是：前一个字符不是字母而后一个是才算一个单词
    for (int t = 0; t < r; t++)
        if(
            !(
                (num[t] > 64 && num[t] < 91)
                ||
                (num[t] > 96 && num[t] < 123) //前一个字符不是字母
            )
            &&
            (
                (num[t + 1] > 64 && num[t + 1] < 91)
                ||
                (num[t + 1] > 96 && num[t + 1] < 123) //后一个字符是字母
            )
        ) i++;
    return i;
}
```

(六) 【运行结果】

```
warning: this program uses gets(), which is unsafe.  
I love China!  
3  
Program ended with exit code: 0  
warning: this program uses gets(), which is unsafe.  
-----hi!!! how are you -----  
4  
Program ended with exit code: 0
```

(七) 【实验报告】

用ascii码判断是否为字母，以前一个不是字母而后一个是字母来判断单词，然后再在字符串的开头加一个识别符用来判断是不是一开始第一个字母就是单词