

2017 年《程序设计导论》课程期末测试 A 卷

2017 — 2018 学年第 1 学期

班级 _____ 姓名 _____ 学号 _____

注意事项

- 1. 本次测试的时间为 150 分钟；编程结果采用机器自动评测。
- 2. 共有 6 题，按评测得分从高到低的顺序，在总成绩中实际所占百分比依次为 25、20、15、15、15、10。
- 3. 提交到在线评测系统中的程序均采用标准输入和标准输出（键盘输入和屏幕输出）。
- 4. 程序设计语言选用 C 或 C++。
- 5. 所有题目的时间限制均为 1s。

一、分类统计

得分	评卷人

【问题描述】

现有一 n 个正整数的数列，分别统计其中是整数 m 的整数倍的元素总和与非整数倍元素之和。

【输入格式】

两行，第 1 行有两个整数，分别为 n 和 m；

第 2 行，n 个整数。数据之间均以空格分隔。

【输出格式】

最多两行，若存在 m 的整数倍元素，第 1 行输出它们的和，后一行输出非整数倍的和；没有则不输出。

【输入样例 1】

6 3
1 2 3 8 9 6

【输出样例 1】

18
11

【数据说明】

1≤n≤1000，2≤m≤9；数列元素以及所求总和均在整型范围内。

二、循环链表

得分	评卷人

【问题描述】

给定一个循环链表，请你编写一个函数，返回该链表的节点数目。特别提示：循环链表是一种链式存储结构，它的最后一个结点指向头结点，形成一个环。

本题是完成函数题目，你无需处理输入输出（以下的输入输出格式仅供参考）。

【输入格式】

一行，包含 n+1（0 ≤ n ≤ 20）个数字。第一个数字 n 表示有 n 个数字，后面跟着 n 个用空格隔开的整数。

【输出格式】

一行，一个整数表示链表中的节点数。

【输入样例】

5 1 2 3 4 5

【输出样例】

5

本题为完成函数题，请你完成返回链表节点数目的函数。

```
#include <iostream>
#include <stdio.h>
#include <string.h>
#include <string>
#include <stdlib.h>
#include <memory.h>
using namespace std;
//节点的结构体类型声明
struct node
{
    int data;
    node *next;
};
//该函数实现把 val 值插入链表最后，并返回插入后链表的头指针
node* Insert(node *head, int val)
{ //此函数代码隐藏 }
//读入数据创建循环链表
node* CreateList(node *head)
{ //此函数代码隐藏 }
```

```
//释放循环链表
void release(node *head)
{
    //此函数代码隐藏
}
//这是你需要完成的函数
//给定循环链表的头指针，请返回该循环链表的节点数
int getLength(node* head);

int main()
{
    node *head = NULL;
    head = CreateList(head); //创建链表
    cout << getLength(head) << endl; //输出链表的节点数目
    release(head);
    return 0;
}
//请在下面实现 getLength 函数
```

三、日期识别

得分	评卷人

【问题描述】
按照我国现行的公历，今天是 2018 年 1 月 3 日，而按传统的阴历是丁酉年，鸡年，十一月十七，壬子月乙未日。日期在日常交往中非常常用，且表示形式五花八门。让计算机读懂日期，不是件简单的事情，但是这个重任交给你了，请你编写一个 C/C++程序，读懂输入的各种格式的日期，并将其转换为统一格式。统一日期格式为 4 位数字表示年，2 位数字表示月，2 位数字表示日期，例如 2018 年 1 月 3 日表示：20180103。

【输入格式】
一行，表示日期的数据。

【输出格式】
一行，转换后的统一日期格式。

【输入样例】
2018.1.3

【输出样例】
20180103

【问题提示】
为了简化程序，不处理汉字，只处理英文日期和阿拉伯数字表示的日期，即中文日期格式和美国日期格式。输入数据月份在日期前；年份可能在最前，也可能在最后，并在 4 位数范围内。可能的输入日期格式如下：

- 第一类：按“月、日、年”顺序排列。例如：
(1) January 3, 2018 （注意逗号后有一个空格）
- 第二类（美国）：按“月、日、年”的顺序排列。例如：
(2) 1.3.2018
(3) 1/3/2018
- 第三类：按“年、月、日”顺序排列。例如：
(4) 2018/1/3
(5) 2018-1-3
(6) 2018.1.3

输入数据中可能出现的英文月份单词，依 1 月、2 月、...12 月顺序为：January, February, March, April, May, June, July, August, September, October, November, December。

四、空间位置搜索

得分	评卷人

【问题描述】
三维空间中有可构成三棱锥的四个点。现需要在刚好能容纳此三棱锥的长方体范围内搜索空间点，使得其到这四个顶点的距离总和最小。已知条件：顶点两两之间其坐标相差均不超过 800；只需在整数坐标范围内查找；任意两点间的距离，计算后向下取整（不四舍五入）。

【输入格式】
四行，每行为一个空间点的 x、y、z 坐标数据，**整型**，空格分隔。

【输出格式】
若干行，第一行两个数，分别为最小距离总和，以及能找到的空间点个数；第二行开始，每行三个整数，分别是空间点的坐标 x、y、z。当有多个点时，按三维坐标值的字典顺序输出，输出时均以**空格分隔**。

【数据说明】
80%的输入，其坐标均在[-300, 300]范围内；70%的输入，仅能找到一个点；所有的输入，最多不会超过 1000 个点，它们到三棱锥顶点的距离总和相等。

【输入样例 1】

-83 255 -14

99 255 -83

26 255 54

11 13 11

【输出样例 1】

497 1

11 225 2

【输入样例 2】

12445 180 -133

12559 171 -98

12473 150 29

12487 289 -79

【输出样例 2】

356 2

12496 198 -84

12499 196 -82

【有关输入的提示】

第一行数字输入后的【回车符】可能被当做一个字符串接受，请注意特殊处理。

【解题提示】

针对一个左括号（小括号或大括号）应取搜索与它“匹配”的右括号，匹配的条件应该同时满足：1) 右括号与它类型相同，2) 嵌套在它们内部的子串满足匹配。如果一个字符串全部是由满足上述条件的部分组成，则整个字符串满足匹配；否则，则字符串不匹配。

【数据说明】

字符串的长度不超过 500。测试字符串数据中：50%的数据要么只包含一种类型的括号要么所有的左括号都在所有的右括号的左边；余下 50%的数据对括号出现的类型与位置并无明确限制。

五、括号匹配检查

得分	评卷人

【问题描述】

编写一段程序判断一段字符串中的大括号"{}"及小括号"()"是否匹配，注意括号间会存在嵌套。例如：

1. 字符串"()"的括号匹配；
2. 字符串"(", ")", ")(("的括号不匹配；
3. 字符串"(abcd{e})"的括号匹配；
4. "(a(b }"的括号不匹配
5. "(a)x{b{cd}ef(g)}"的括号匹配

【输入格式】

共 n+1 行。

第一行：一个数字 n

第 2-(n+1)行：每行一个字符串。【注意：可能包含空格。】

【输出格式】

共 n 行：每行输出相应字符串括号是否匹配:匹配输出 true，不匹配输出 false。

【输入样例 1】

1

(a)x{b{cd}ef(g)}

【输出样例 1】

true

【输入样例 2】

2

(a)(b }

if (a == b) { printf("true"); }

【输出样例 2】

false

true

六、排列问题

得分	评卷人

【问题描述】

按照字典序输出的 n 小写英字符对应的所有排列，注意 n 个字符中可能存在相同的字符。

【输入格式】

两行，分别为 1 个整数 n，和一个长度为 n 字符串（字符串中仅包含小写英文字母），表示需要排列 n 个字符。

【输出格式】

为 m+1 行，前面 m 行依次为按字典序输出的排列结果，最后一行为一个整数，表示不同排列的总数 m。

【输入样例】

4

egge

【输出样例】

eegg

egeg

egge

geeg

gege

gg ee

6

【数据规模说明】

1≤n≤500