

## 第六届蓝桥杯大赛个人赛省赛（软件类）

### C/C++ 大学 A 组

#### 考生须知：

- 考试开始后，选手首先下载题目，并使用考场现场公布的解压密码解压试题。
  - 考试时间为 4 小时，时间截止后，提交答案无效。
  - 在考试强制结束前，选手可以主动结束考试（需要身份验证），结束考试后将无法继续提交或浏览答案。
  - 选手可浏览自己已经提交的答案，被浏览的答案允许拷贝。
  - 对同一题目，选手可多次提交答案，以最后一次提交的答案为准。
  - 选手切勿在提交的代码中书写“姓名”、“考号”、“院校名”等与身份有关的信息或其它与竞赛题目无关的内容，否则成绩无效。
  - 选手必须通过浏览器方式提交自己的答案，选手在其它位置的作答或其他方式提交的答案无效。
  - 试题包含三种类型：“结果填空”、“代码填空”与“程序设计”。
- 结果填空题：**要求选手根据题目描述直接填写结果，**求解方式不限**，不要求源代码，把结果填空的答案直接通过网页提交即可，不要书写多余的内容。
- 代码填空题：**要求选手在弄清给定代码工作原理的基础上填写缺失的部分，使得程序逻辑正确、完整。
- 把代码填空的答案（仅填空处的答案，**不包括题目已存在的代码或符号**）直接通过网页提交即可，不要书写多余的内容。
- 使用 ANSI C/ANSI C++ 标准，不要依赖操作系统或编译器提供的特殊函数。
- 程序设计题目：**要求选手设计的程序对于给定的输入能给出正确的输出结果，**考生的程序只有能运行出正确结果才有机会得分**。
- 注意：在评卷时使用的输入数据与试卷中给出的示例数据可能是不同的，**选手的程序必须是通用的**，不能只对试卷中给定的数据有效。
- 对于编程题目，要求选手给出的解答完全符合 ANSI C++ 标准，不能使用诸如绘图、Win32 API、中断调用、硬件操作或与操作系统相关的 API。
- 代码中允许使用 STL 类库。
- 注意：main 函数结束必须返回 0**
- 注意：所有依赖的函数必须明确地在源文件中 `#include <xxx>`，不能通过工程设置而省略常用头文件。
- 所有源码必须在同一文件中，调试通过后，拷贝提交。
- 提交时，注意选择所期望的编译器类型。

#### 1. 结果填空（满分 3 分）

问题的描述在考生文件夹下对应题号的“题目.txt”中，相关的参考文件在同一目录中，请先阅读题目，不限解决问题的方式，只要求提交结果。

必须通过浏览器提交答案。

#### 2. 结果填空（满分 5 分）

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，不限解决问题的方式，只要求提交结果。  
必须通过浏览器提交答案。

### 3. 结果填空 (满分 9 分)

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，不限解决问题的方式，只要求提交结果。  
必须通过浏览器提交答案。

### 4. 代码填空 (满分 11 分)

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，不限解决问题的方式。  
只要求填写缺失的代码部分，千万不要画蛇添足，填写多余的已有代码或符号。  
必须通过浏览器提交答案。

### 5. 代码填空 (满分 15 分)

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，不限解决问题的方式。  
只要求填写缺失的代码部分，千万不要画蛇添足，填写多余的已有代码或符号。  
必须通过浏览器提交答案。

### 6. 结果填空 (满分 17 分)

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，不限解决问题的方式，只要求提交结果。  
必须通过浏览器提交答案。

### 7. 结果填空 (满分 21 分)

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，不限解决问题的方式，只要求提交结果。  
必须通过浏览器提交答案。

### 8. 程序设计 (满分 13 分)

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，必须通过编程的方式解决问题。

注意：在评卷时使用的输入数据与试卷中给出的示例数据可能是不同的。**选手的程序必须是通用的**，不能只对试卷中给定的数据有效。

仔细阅读程序的输入、输出要求，千万不要输出没有要求的、多余的内容，例如：“请

您输入 xx 数据: ”。

建议仔细阅读示例，不要想当然!

程序处理完一个用例的数据后，立即退出 (return 0)，千万不要循环等待下一个用例的输入。

程序必须使用标准输入、标准输出，以便于机器评卷时重定向。

对于编程题目，要求选手给出的解答完全符合 ANSI C++ 标准，不能使用诸如绘图、Win32API、中断调用、硬件操作或与操作系统相关的 API。

代码中允许使用 STL 类库。

注意: main 函数结尾需要 return 0

注意: 所有依赖的函数必须明确地在源文件中 #include <xxx>， 不能通过工程设置而省略常用头文件。

所有代码放在同一个源文件中，调试通过后，拷贝提交该源码。

提交时，注意选择所期望的编译器类型。

## 9. 程序设计 (满分 25 分)

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，必须通过编程的方式解决问题。

注意: 在评卷时使用的输入数据与试卷中给出的示例数据可能是不同的。**选手的程序必须是通用的**，不能只对试卷中给定的数据有效。

仔细阅读程序的输入、输出要求，千万不要输出没有要求的、多余的内容，例如: “请您输入 xx 数据: ”。

建议仔细阅读示例，不要想当然!

程序处理完一个用例的数据后，立即退出 (return 0)，千万不要循环等待下一个用例的输入。

程序必须使用标准输入、标准输出，以便于机器评卷时重定向。

对于编程题目，要求选手给出的解答完全符合 ANSI C++ 标准，不能使用诸如绘图、Win32API、中断调用、硬件操作或与操作系统相关的 API。

代码中允许使用 STL 类库。

注意: main 函数结尾需要 return 0

注意: 所有依赖的函数必须明确地在源文件中 #include <xxx>， 不能通过工程设置而省略常用头文件。

所有代码放在同一个源文件中，调试通过后，拷贝提交该源码。

提交时，注意选择所期望的编译器类型。

## 10. 程序设计 (满分 31 分)

问题的描述在考生文件夹下对应题号的“题目.txt”中。相关的参考文件在同一目录中。请先阅读题目，必须通过编程的方式解决问题。

注意: 在评卷时使用的输入数据与试卷中给出的示例数据可能是不同的。**选手的程序必须是通用的**，不能只对试卷中给定的数据有效。

仔细阅读程序的输入、输出要求，千万不要输出没有要求的、多余的内容，例如: “请您输入 xx 数据: ”。

建议仔细阅读示例，不要想当然!

程序处理完一个用例的数据后，立即退出（return 0），千万不要循环等待下一个用例的输入。

程序必须使用标准输入、标准输出，以便于机器评卷时重定向。

对于编程题目，要求选手给出的解答完全符合 ANSI C++ 标准，不能使用诸如绘图、Win32 API、中断调用、硬件操作或与操作系统相关的 API。

代码中允许使用 STL 类库。

注意：main 函数结尾需要 return 0

注意：所有依赖的函数必须明确地在源文件中 #include <xxx>， 不能通过工程设置而省略常用头文件。

所有代码放在同一个源文件中，调试通过后，拷贝提交该源码。

提交时，注意选择所期望的编译器类型。

### 方程整数解

方程:  $a^2 + b^2 + c^2 = 1000$

(或参见【图 1.jpg】)

这个方程有整数解吗？有：a,b,c=6,8,30 就是一组解。

你能算出另一组合适的解吗？

请填写该解中最小的数字。

注意：你提交的应该是一个整数，不要填写任何多余的内容或说明性文字。

$$a^2 + b^2 + c^2 = 1000$$

其中，a,b,c 都是整数

### 星系炸弹

在 X 星系的广袤空间中漂浮着许多 X 星人造“炸弹”，用来作为宇宙中的路标。

每个炸弹都可以设定多少天之后爆炸。

比如：阿尔法炸弹 2015 年 1 月 1 日放置，定时为 15 天，则它在 2015 年 1 月 16 日爆炸。

有一个贝塔炸弹，2014 年 11 月 9 日放置，定时为 1000 天，请你计算它爆炸的准确日期。

请填写该日期，格式为 yyyy-mm-dd 即 4 位年份 2 位月份 2 位日期。比如：2015-02-19

请严格按照格式书写。不能出现其它文字或符号。

### 奇妙的数字

小明发现了一个奇妙的数字,它的平方和立方正好把 0~9 的 10 个数字每个用且只用了一次,你能猜出这个数字是多少吗?

请填写该数字,不要填写任何多余的内容。

### 格子中输出

StringInGrid 函数会在一个指定大小的格子中打印指定的字符串。  
要求字符串在水平、垂直两个方向上都居中。  
如果字符串太长,就截断。  
如果不能恰好居中,可以稍稍偏左或者偏上一点。

下面的程序实现这个逻辑,请填写划线部分缺少的代码。

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

void StringInGrid(int width, int height, const char* s)
{
    int i,k;
    char buf[1000];
    strcpy(buf, s);
    if(strlen(s)>width-2) buf[width-2]=0;

    printf("+");
    for(i=0;i<width-2;i++) printf("-");
    printf("\n");

    for(k=1; k<=(height-1)/2;k++){
        printf("|");
        for(i=0;i<width-2;i++) printf(" ");
        printf("\n");
    }

    printf("|");

    printf("%*s%*s%*s", _____); //填空

    printf("\n");
}
```



```

        for(k=(height-1)/2+1; k<height-1; k++){
            printf("|");
            for(i=0;i<width-2;i++) printf(" ");
            printf("|\\n");
        }

        printf("+");
        for(i=0;i<width-2;i++) printf("-");
        printf("+\\n");
    }

int main()
{
    StringInGrid(20,6,"abcd1234");
    return 0;
}

```

对于题目中数据，应该输出：

```

+-----+
|          |
|  abcd1234  |
|          |
|          |
+-----+

```

(如果出现对齐问题，参看【图 1.jpg】)

注意：只填写缺少的内容，不要书写任何题面已有代码或说明性文字。



九数组分数

1,2,3...9 这九个数字组成一个分数，其值恰好为 1/3，如何组法？

下面的程序实现了该功能，请填写划线部分缺失的代码。

```

#include <stdio.h>

void test(int x[])
{
    int a = x[0]*1000 + x[1]*100 + x[2]*10 + x[3];
    int b = x[4]*10000 + x[5]*1000 + x[6]*100 + x[7]*10 + x[8];

    if(a*3==b) printf("%d / %d\n", a, b);
}

void f(int x[], int k)
{
    int i,t;
    if(k>=9){
        test(x);
        return;
    }

    for(i=k; i<9; i++){
        {t=x[k]; x[k]=x[i]; x[i]=t;}
        f(x,k+1);
        _____ // 填空处
    }
}

int main()
{
    int x[] = {1,2,3,4,5,6,7,8,9};
    f(x,0);
    return 0;
}

```

注意：只填写缺少的内容，不要书写任何题面已有代码或说明性文字。

#### 牌型种数

小明被劫持到 X 赌城，被迫与其他 3 人玩牌。  
 一副扑克牌（去掉大小王牌，共 52 张），均匀发给 4 个人，每个人 13 张。  
 这时，小明脑子里突然冒出一个问题：  
 如果不考虑花色，只考虑点数，也不考虑自己得到的牌的先后顺序，自己手里能拿到的初始

牌型组合一共有多少种呢?

请填写该整数，不要填写任何多余的内容或说明文字。

手链样式

小明有 3 颗红珊瑚，4 颗白珊瑚，5 颗黄玛瑙。  
他想用它们串成一圈作为手链，送给女朋友。  
现在小明想知道：如果考虑手链可以随意转动或翻转，一共可以有多少不同的组合样式呢？

请你提交该整数，不要填写任何多余的内容或说明性的文字。

饮料换购

乐羊羊饮料厂正在举办一次促销优惠活动。乐羊羊 C 型饮料，凭 3 个瓶盖可以再换一瓶 C 型饮料，并且可以一直循环下去(但不允许暂借或赊账)。

请你计算一下，如果小明不浪费瓶盖，尽量地参加活动，那么，对于他初始买入的 n 瓶饮料，最后他一共能喝到多少瓶饮料。

输入：一个整数 n，表示开始购买的饮料数量 (0<n<10000)

输出：一个整数，表示实际得到的饮料数

例如：

用户输入：

100

程序应该输出：

149

用户输入：

101

程序应该输出：

151

资源约定：

峰值内存消耗 < 256M

CPU 消耗 < 1000ms

请严格按照要求输出，不要画蛇添足地打印类似：“请您输入...” 的多余内容。



所有代码放在同一个源文件中，调试通过后，拷贝提交该源码。

注意: main 函数需要返回 0

注意: 只使用 ANSI C/ANSI C++ 标准，不要调用依赖于编译环境或操作系统的特殊函数。

注意: 所有依赖的函数必须明确地在源文件中 `#include <xxx>`， 不能通过工程设置而省略常用头文件。

提交时，注意选择所期望的编译器类型。

### 垒骰子

赌圣 atm 晚年迷恋上了垒骰子，就是把骰子一个垒在另一个上边，不能歪歪扭扭，要垒成方柱体。

经过长期观察，atm 发现了稳定骰子的奥秘：有些数字的面贴着会互相排斥！

我们先来规范一下骰子：1 的对面是 4，2 的对面是 5，3 的对面是 6。

假设有  $m$  组互斥现象，每组中的那两个数字的面紧贴在一起，骰子就不能稳定的垒起来。

atm 想计算一下有多少种不同的可能的垒骰子方式。

两种垒骰子方式相同，当且仅当这两种方式中对应高度的骰子的对应数字的朝向都相同。

由于方案数可能过多，请输出模  $10^9 + 7$  的结果。

不要小看了 atm 的骰子数量哦~

「输入格式」

第一行两个整数  $n\ m$

$n$  表示骰子数目

接下来  $m$  行，每行两个整数  $a\ b$ ，表示  $a$  和  $b$  数字不能紧贴在一起。

「输出格式」

一行一个数，表示答案模  $10^9 + 7$  的结果。

「样例输入」

2 1

1 2

「样例输出」

544

「数据范围」

对于 30% 的数据:  $n \leq 5$

对于 60% 的数据:  $n \leq 100$

对于 100% 的数据:  $0 < n \leq 10^9, m \leq 36$

资源约定:  
峰值内存消耗 < 256M  
CPU 消耗 < 2000ms

请严格按照要求输出, 不要画蛇添足地打印类似: “请输入...” 的多余内容。

所有代码放在同一个源文件中, 调试通过后, 拷贝提交该源码。

注意: main 函数需要返回 0

注意: 只使用 ANSI C/ANSI C++ 标准, 不要调用依赖于编译环境或操作系统的特殊函数。

注意: 所有依赖的函数必须明确地在源文件中 `#include <xxx>`, 不能通过工程设置而省略常用头文件。

提交时, 注意选择所期望的编译器类型。

## 灾后重建

Pear 市一共有  $N$  ( $\leq 50000$ ) 个居民点, 居民点之间有  $M$  ( $\leq 200000$ ) 条双向道路相连。这些居民点两两之间都可以通过双向道路到达。这种情况一直持续到最近, 一次严重的地震毁坏了全部  $M$  条道路。

震后, Pear 打算修复其中一些道路, 修理第  $i$  条道路需要  $P_i$  的时间。不过, Pear 并不打算让全部的点连通, 而是选择一些标号特殊的点让他们连通。

Pear 有  $Q$  ( $\leq 50000$ ) 次询问, 每次询问, 他会选择所有编号在  $[l, r]$  之间, 并且 编号  $\bmod K = C$  的点, 修理一些路使得它们连通。由于所有道路的修理可以同时开工, 所以完成修理的时间取决于花费时间最长的一条路, 即涉及到的道路中  $P_i$  的最大值。

你能帮助 Pear 计算出每次询问时需要花费的最少时间么? 这里询问是独立的, 也就是上一个询问里的修理计划并没有付诸行动。

### 【输入格式】

第一行三个正整数  $N$ 、 $M$ 、 $Q$ , 含义如题面所述。

接下来  $M$  行, 每行三个正整数  $X_i$ 、 $Y_i$ 、 $P_i$ , 表示一条连接  $X_i$  和  $Y_i$  的双向道路, 修复需要  $P_i$  的时间。可能有自环, 可能有重边。  $1 \leq P_i \leq 1000000$ 。

接下来  $Q$  行, 每行四个正整数  $L_i$ 、 $R_i$ 、 $K_i$ 、 $C_i$ , 表示这次询问的点是  $[L_i, R_i]$  区间中所有编号  $\bmod K_i = C_i$  的点。保证参与询问的点至少有两个。

### 【输出格式】

输出  $Q$  行, 每行一个正整数表示对应询问的答案。

**【样例输入】**

```
7 10 4
1 3 10
2 6 9
4 1 5
3 7 4
3 6 9
1 5 8
2 7 4
3 2 10
1 7 6
7 6 9
1 7 1 0
1 7 3 1
2 5 1 0
3 7 2 1
```

**【样例输出】**

```
9
6
8
8
```

**【数据范围】**

对于 20%的数据,  $N, M, Q \leq 30$

对于 40%的数据,  $N, M, Q \leq 2000$

对于 100%的数据,  $N \leq 50000, M \leq 2 \cdot 10^5, Q \leq 50000$ .  $P_i \leq 10^6$ .  $L_i, R_i, K_i$  均在  $[1, N]$  范围内,  
 $C_i$  在  $[0, \text{对应询问的 } K_i]$  范围内.

资源约定:

峰值内存消耗  $< 256\text{M}$

CPU 消耗  $< 5000\text{ms}$

请严格按照要求输出, 不要画蛇添足地打印类似: “请您输入...” 的多余内容.

所有代码放在同一个源文件中, 调试通过后, 拷贝提交该源码.

注意: `main` 函数需要返回 0

注意: 只使用 ANSI C/ANSI C++ 标准, 不要调用依赖于编译环境或操作系统的特殊函数.

注意: 所有依赖的函数必须明确地在源文件中 `#include <xxx>`, 不能通过工程设置而省略常用头文件。

提交时, 注意选择所期望的编译器类型。