

第九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 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。

2. 结果填空（满分 7 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. 结果填空 (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，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。

5. 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。

6. 程序设计 (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，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。

7. 程序设计 (满分 19 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。
注意事项同上题

8. 程序设计 (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，必须通过编程的方式解决问题。
注意事项同上题

9. 程序设计 (满分 23 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。
注意事项同上题

10. 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。
注意事项同上题

标题: 分数

$1/1 + 1/2 + 1/4 + 1/8 + 1/16 + \dots$

每项是前一项的一半，如果一共有 20 项，
求这个和是多少，结果用分数表示出来。

类似:

$3/2$

当然，这只是加了前 2 项而已。分子分母要求互质。

注意:

需要提交的是已经约分过的分数，中间任何位置不能含有空格。
请不要填写任何多余的文字或符号。

标题: 星期一

整个 20 世纪 (1901 年 1 月 1 日至 2000 年 12 月 31 日之间), 一共有多少个星期一?
(不要告诉我你不知道今天是星期几)

注意: 需要提交的只是一个整数, 不要填写任何多余的内容或说明文字。

标题: 乘积尾零

如下的 10 行数据, 每行有 10 个整数, 请你求出它们的乘积的末尾有多少个零?

5650 4542 3554 473 946 4114 3871 9073 90 4329
2758 7949 6113 5659 5245 7432 3051 4434 6704 3594
9937 1173 6866 3397 4759 7557 3070 2287 1453 9899
1486 5722 3135 1170 4014 5510 5120 729 2880 9019
2049 698 4582 4346 4427 646 9742 7340 1230 7683
5693 7015 6887 7381 4172 4341 2909 2027 7355 5649
6701 6645 1671 5978 2704 9926 295 3125 3878 6785
2066 4247 4800 1578 6652 4616 1113 6205 3264 2915
3966 5291 2904 1285 2193 1428 2265 8730 9436 7074
689 5510 8243 6114 337 4096 8199 7313 3685 211

注意: 需要提交的是一个整数, 表示末尾零的个数, 不要填写任何多余内容。

标题: 第几个幸运数

到 x 星球旅行的游客都被发给一个整数, 作为游客编号。
x 星的国王有个怪癖, 他只喜欢数字 3,5 和 7。
国王规定, 游客的编号如果只含有因子: 3,5,7,就可以获得一份奖品。

我们来看前 10 个幸运数字是:
3 5 7 9 15 21 25 27 35 45
因而第 11 个幸运数字是: 49

小明领到了一个幸运数字 59084709587505, 他去领奖的时候, 人家要求他准确地说出这是第几个幸运数字, 否则领不到奖品。

请你帮小明计算一下, 59084709587505 是第几个幸运数字。

需要提交的是一个整数, 请不要填写任何多余内容。

标题: 打印图形

如下的程序会在控制台绘制分形图 (就是整体与局部自相似的图形)。

当 n=1,2,3 的时候, 输出如下:

请仔细分析程序, 并填写划线部分缺少的代码。

n=1 时:

```
o
ooo
o
```

n=2 时:

```
o
  ooo
  o
o o o
oooooooo
o o o
o
  ooo
  o
```

n=3 时:

```
o
  ooo
  o
    o o o
    ooooooooo
    o o o
    o
      ooo
      o
o      o      o
ooo    ooo    ooo
o      o      o
o o o o o o o o o o
oooooooooooooooooooooooooooo
o o o o o o o o o o
o      o      o
ooo    ooo    ooo
o      o      o
      o
```

```

    ooo
    o
  o o o
oooooooo
  o o o
    o
    ooo
    o

```

源程序:

```

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

void show(char* buf, int w){
    int i,j;
    for(i=0; i<w; i++){
        for(j=0; j<w; j++){
            printf("%c", buf[i*w+j]==0? ' ': 'o');
        }
        printf("\n");
    }
}

void draw(char* buf, int w, int x, int y, int size){
    if(size==1){
        buf[y*w+x] = 1;
        return;
    }

    int n = _____; //填空
    draw(buf, w, x, y, n);
    draw(buf, w, x-n, y ,n);
    draw(buf, w, x+n, y ,n);
    draw(buf, w, x, y-n ,n);
    draw(buf, w, x, y+n ,n);
}

int main()
{
    int N = 3;
    int t = 1;
    int i;

```

```

for(i=0; i<N; i++) t *= 3;

char* buf = (char*)malloc(t*t);
for(i=0; i<t*t; i++) buf[i] = 0;

draw(buf, t, t/2, t/2, t);
show(buf, t);
free(buf);

return 0;
}

```

注意：只提交划线部分缺少的代码，不要抄写任何已经存在的代码或符号。

标题：航班时间

【问题背景】

小 h 前往美国参加了蓝桥杯国际赛，小 h 的女朋友发现小 h 上午十点出发，上午十二点到达美国，于是感叹到“现在飞机飞得真快，两小时就能到美国了”。

小 h 对超音速飞行感到十分恐惧，仔细观察后发现飞机的起降时间都是当地时间。由于北京和美国东部有 12 小时时差，故飞机总共需要 14 小时的飞行时间。

不久后小 h 的女朋友去中东交换，小 h 并不知道中东与北京的时差。但是小 h 得到了女朋友来回航班的起降时间，小 h 想知道女朋友的航班飞行时间是多少。

【问题描述】

对于一个可能跨时区的航班，给定来回程的起降时间。假设飞机来回飞行时间相同，求飞机的飞行时间。

【输入格式】

从标准输入读入数据。
一个输入包含多组数据。

输入第一行为一个正整数 T，表示输入数据组数。

每组数据包含两行，第一行为去程的 起降 时间，第二行为回程的 起降 时间。
起降时间的格式如下

h1:m1:s1 h2:m2:s2
或
h1:m1:s1 h3:m3:s3 (+1)

或

h1:m1:s1 h4:m4:s4 (+2)

表示该航班在当地时间 h1 时 m1 分 s1 秒起飞,

第一种格式表示在当地时间 当日 h2 时 m2 分 s2 秒降落

第二种格式表示在当地时间 次日 h3 时 m3 分 s3 秒降落.

第三种格式表示在当地时间 第三天 h4 时 m4 分 s4 秒降落.

对于此题目中的所有以 h:m:s 形式给出的时间, 保证 ($0 \leq h \leq 23, 0 \leq m, s \leq 59$).

【输出格式】

输出到标准输出.

对于每一组数据输出一行一个时间 hh:mm:ss, 表示飞行时间为 hh 小时 mm 分 ss 秒.

注意, 当时间为一位数时, 要补齐前导零. 如三小时四分五秒应写为 03:04:05.

【样例输入】

3

17:48:19 21:57:24

11:05:18 15:14:23

17:21:07 00:31:46 (+1)

23:02:41 16:13:20 (+1)

10:19:19 20:41:24

22:19:04 16:41:09 (+1)

【样例输出】

04:09:05

12:10:39

14:22:05

【限制与约定】

保证输入时间合法, 飞行时间不超过 24 小时.

资源约定:

峰值内存消耗 (含虚拟机) $< 256\text{M}$

CPU 消耗 $< 1000\text{ms}$

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.

注意:

main 函数需要返回 0;

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。
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。

提交程序时，注意选择所期望的语言类型和编译器类型。

标题：三体攻击

【题目描述】

三体人将对地球发起攻击。为了抵御攻击，地球人派出了 $A \times B \times C$ 艘战舰，在太空中排成一个 A 层 B 行 C 列的立方体。其中，第 i 层第 j 行第 k 列的战舰（记为战舰 (i, j, k) ）的生命值为 $d(i, j, k)$ 。

三体人将会对地球发起 m 轮“立方体攻击”，每次攻击会对一个小立方体中的所有战舰都造成相同的伤害。具体地，第 t 轮攻击用 7 个参数 $lat, rat, lbt, rbt, lct, rct, ht$ 描述：所有满足 $i \in [lat, rat], j \in [lbt, rbt], k \in [lct, rct]$ 的战舰 (i, j, k) 会受到 ht 的伤害。如果一个战舰累计受到的总伤害超过其防御力，那么这个战舰会爆炸。

地球指挥官希望你能告诉他，第一艘爆炸的战舰是在哪一轮攻击后爆炸的。

【输入格式】

从标准输入读入数据。

第一行包括 4 个正整数 A, B, C, m ；

第二行包含 $A \times B \times C$ 个整数，其中第 $((i - 1) \times B + (j - 1)) \times C + (k - 1) + 1$ 个数为 $d(i, j, k)$ ；

第 3 到第 $m + 2$ 行中，第 $(t - 2)$ 行包含 7 个正整数 $lat, rat, lbt, rbt, lct, rct, ht$ 。

【输出格式】

输出到标准输出。

输出第一个爆炸的战舰是在哪一轮攻击后爆炸的。保证一定存在这样的战舰。

【样例输入】

```
2 2 2 3
1 1 1 1 1 1 1
1 2 1 2 1 1 1
1 1 1 2 1 2 1
1 1 1 1 1 1 2
```

【样例输出】

```
2
```

【样例解释】

在第 2 轮攻击后，战舰 (1,1,1) 总共受到了 2 点伤害，超出其防御力导致爆炸。

【数据约定】

对于 10% 的数据， $B = C = 1$ ；

对于 20% 的数据， $C = 1$ ；

对于 40% 的数据， $A \times B \times C, m \leq 10,000$ ；

对于 70% 的数据， $A, B, C \leq 200$ ；

对于所有数据， $A \times B \times C \leq 10^6, m \leq 10^6, 0 \leq d(i, j, k), ht \leq 10^9$ 。

资源约定：

峰值内存消耗（含虚拟机） < 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：“请您输入...” 的多余内容。

注意：

main 函数需要返回 0；

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。

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。

提交程序时，注意选择所期望的语言类型和编译器类型。

标题：全球变暖

【题目描述】

你有一张某海域 $N \times N$ 像素的照片，“.”表示海洋、“#”表示陆地，如下所示：

```
.....
.##....
.##....
....##.
.####.
...###.
.....
```

其中"上下左右"四个方向上连在一起的一片陆地组成一座岛屿。例如上图就有 2 座岛屿。

由于全球变暖导致了海面上升，科学家预测未来几十年，岛屿边缘一个像素的范围会被海水淹没。具体来说如果一块陆地像素与海洋相邻(上下左右四个相邻像素中有海洋)，它就会被淹没。

例如上图中的海域未来会变成如下样子：

```
.....
.....
.....
.....
....#.
.....
.....
```

请你计算：依照科学家的预测，照片中有多少岛屿会被完全淹没。

【输入格式】

第一行包含一个整数 N 。 ($1 \leq N \leq 1000$)

以下 N 行 N 列代表一张海域照片。

照片保证第 1 行、第 1 列、第 N 行、第 N 列的像素都是海洋。

【输出格式】

一个整数表示答案。

【样例输入】

```
7
.....
.##....
.##....
....##.
.####.
...###.
.....
```

【样例输出】

```
1
```

资源约定：

峰值内存消耗（含虚拟机） $< 256\text{M}$

CPU 消耗 $< 1000\text{ms}$

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。

注意：

main 函数需要返回 0；

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。

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。

提交程序时，注意选择所期望的语言类型和编译器类型。

标题：倍数问题

【题目描述】

众所周知，小葱同学擅长计算，尤其擅长计算一个数是否是另外一个数的倍数。但小葱只擅长两个数的情况，当有很多个数之后就会比较苦恼。现在小葱给了你 n 个数，希望你从这 n 个数中找到三个数，使得这三个数的和是 K 的倍数，且这个和最大。数据保证一定有解。

【输入格式】

从标准输入读入数据。

第一行包括 2 个正整数 n, K 。

第二行 n 个正整数，代表给定的 n 个数。

【输出格式】

输出到标准输出。

输出一行一个整数代表所求的和。

【样例入】

4 3

1 2 3 4

【样例输出】

9

【样例解释】

选择 2、3、4。

【数据约定】

对于 30% 的数据， $n \leq 100$ 。

对于 60% 的数据, $n \leq 1000$.
对于另外 20% 的数据, $K \leq 10$.
对于 100% 的数据, $1 \leq n \leq 10^5, 1 \leq K \leq 10^3$, 给定的 n 个数均不超过 10^8 .

资源约定:
峰值内存消耗 (含虚拟机) $< 256\text{M}$
CPU 消耗 $< 1000\text{ms}$

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.

注意:
main 函数需要返回 0;
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.
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.

提交程序时, 注意选择所期望的语言类型和编译器类型.

标题: 付账问题

【题目描述】
几个人一起出去吃饭是常有的事。但在结帐的时候，常常会出现一些争执。

现在有 n 个人出去吃饭，他们总共消费了 S 元。其中第 i 个人带了 a_i 元。幸运的是，所有人带的钱的总数是足够付账的，但现在问题来了：每个人分别要出多少钱呢？

为了公平起见，我们希望在总付钱量恰好为 S 的前提下，最后每个人付的钱的标准差最小。这里我们约定，每个人支付的钱数可以是任意非负实数，即可以不是 1 分钱的整数倍。你需要输出最小的标准差是多少。

标准差的介绍：标准差是多个数与它们平均数差值的平方平均数，一般用于刻画这些数之间的“偏差有多大”。形式化地说，设第 i 个人付的钱为 b_i 元，那么标准差为：[参见 p1.png]

【输入格式】
从标准输入读入数据。

第一行包含两个整数 n, S ;
第二行包含 n 个非负整数 a_1, \dots, a_n .

【输出格式】

输出到标准输出。

输出最小的标准差，四舍五入保留 4 位小数。

保证正确答案在加上或减去 10^{-9} 后不会导致四舍五入的结果发生变化。

【样例 1 输入】

5 2333

666 666 666 666 666

【样例输出】

0.0000

【样例解释】

每个人都出 2333/5 元，标准差为 0。

再比如：

【样例输入】

10 30

2 1 4 7 4 8 3 6 4 7

【样例输出】

0.7928

【数据说明】

对于 10% 的数据，所有 a_i 相等；

对于 30% 的数据，所有非 0 的 a_i 相等；

对于 60% 的数据， $n \leq 1000$ ；

对于 80% 的数据， $n \leq 10^5$ ；

对于所有数据， $n \leq 5 \times 10^5, 0 \leq a_i \leq 10^9$ 。

资源约定：

峰值内存消耗（含虚拟机） $< 256M$

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。

注意：

main 函数需要返回 0；

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。

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。

提交程序时，注意选择所期望的语言类型和编译器类型。

$$s = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(b_i - \frac{1}{n} \sum_{i=1}^n b_i \right)^2}$$