买铅笔（pencil）

【问题描述】

P老师需要去商店买*n*支铅笔作为小朋友们参加NOIP的礼物。她发现商店一共有 3种包装的铅笔，不同包装内的铅笔数量有可能不同，价格也有可能不同。为了公平起 见，P老师决定只买同一种包装的铅笔。

商店不允许将铅笔的包装拆开，因此P老师可能需要购买超过*n*支铅笔才够给小朋 友们发礼物。

现在P老师想知道，在商店每种包装的数量都足够的情况下，要买够至少*n*支铅 笔最少需要花费多少钱。

【输入格式】

从文件*pencil.in*中读入数据。

输入的第一行包含一个正整数*n*，表示需要的铅笔数量。

接下来三行，每行用两个正整数描述一种包装的铅笔：其中第一个整数表示这种包装内铅笔的数量，第二个整数表示这种包装的价格。

保证所有的7个数都是不超过10000的正整数。

【输出格式】

输出到文件*pencil.out*中。

输出一行一个整数，表示P老师最少需要花费的钱。

【样例1输入】

```
57
2 2
50 30
30 27
```

【样例1输出】

```
54
```

【样例1说明】

铅笔的三种包装分别是：

- 2支装，价格为2 ；
- 50支装，价格为30 ；
- 30支装，价格为27 。

P老师需要购买至少57支铅笔。

如果她选择购买第一种包装，那么她需要购买29份，共计2 x 29 = 58支，需要花费的钱为2 x 29 = 58

实际上，P老师会选择购买第三种包装，这样需要买2份。虽然最后买到的铅笔数 量更多了，为30 x 2 = 60支，但花费却减少为27 x 2 = 54，比第一种少。

对于第二种包装，虽然每支铅笔的价格是最低的，但要够发必须买2份，实际的花费达到了 30 x 2 = 60，因此P老师也不会选择。

所以最后输出的答案是54

【样例2输入】

```
9998
128 233
128 2333
128 666
```

【样例2输出】

```
18407
```

【样例3输入】

```
9999
101 1111
1 9999
1111 9999
```

【样例3输出】

```
89991
```

回文日期（date)

【问题描述】

在日常生活中，通过年、月、日这三个要素可以表示出一个唯一确定的日期。

牛牛习惯用8位数字表示一个日期，其中，前4位代表年份，接下来2位代表月 份，最后2位代表日期。显然：一个日期只有一种表示方法，而两个不同的日期的表示方法不会相同。

牛牛认为，一个日期是回文的，当且仅当表示这个日期的8位数字是回文的。现 在，牛牛想知道：在他指定的两个日期之间（包含这两个日期本身），有多少个真实存 在的日期是回文的。

【提示】

一个8位数字是回文的，当且仅当对于所有的i（ $1 < i < 8$ ）从左向右数的第i个 数字和第9 - i个数字（即从右向左数的第i个数字）是相同的。

例如：

•对于2016年11月19日，用8位数字20161119表示，它不是回文的。

•对于2010年1月2日，用8位数字20100102表示，它是回文的。

•对于2010年10月2日，用8位数字20101002表示，它不是回文的。

每一年中都有12个月份：

其中，1、3、5、7、8、10、12月每个月有31天；4、6、9、11月每个 月有30天；而对于2月，闰年时有29天，平年时有28天。

一个年份是闰年当且仅当它满足下列两种情况其中的一种：

1. 这个年份是4的整数倍，但不是100的整数倍；
2. 这个年份是400的整数倍。

例如：

•以下几个年份都是闰年：2000、2012、2016。

•以下几个年份是平年：1900、2011、2014 "

【输入格式】

从文件*date.in*中读入数据。

输入包括两行，每行包括一个8位数字。

第一行表示牛牛指定的起始日期$date_1$ "

第二行表示牛牛指定的终止日期$date_2$ "

保证$date_1$和$date_2$都是真实存在的日期，且年份部分一定为4位数字，且首位数 字不为0 "

保证$date_1$ 一定不晚于$date_2$

【输出格式】

输出到文件*date.out*中。

输出一行，包含一个整数，表示在$date_1$和$date_2$之间，有多少个日期是回文的。

【样例1输入】

20110101
20111231

【样例1输出】

1

【样例2输入】

20000101
20101231

【样例2输出】

2

【样例说明】

对于样例1,符合条件的日期是20111102
对于样例2,符合条件的日期是20011002和20100102。

【子任务】

对于60%的数据，满足$date_1 = date$

金币

(coin.cpp/c/pas)

【问题描述】

国王将金币作为工资，发放给忠诚的骑士。第一天，骑士收到一枚金币；之后两天（第二天和第三天）每天收到两枚金币；之后三天（第四、五、六天）每天收到三枚金币；之后四天（第七、八、九、十天）每天收到四枚金币……；这种工资发放模式会一直这样延续下去：当连续 N 天每天收到 N 枚金币后，骑士会在之后的连续 N+1 天里，每天收到 N+1 枚金币。

请计算在前K 天里，骑士一共获得了多少金币。

【输入格式】

输入文件名为 coin.in。

输入文件只有 1 行，包含一个正整数K，表示发放金币的天数。

【输出格式】

输出文件名为 coin.out。

输出文件只有 1 行，包含一个正整数，即骑士收到的金币数。

【输入输出样例 1】

coin.in	**coin.out**
6	14

见选手目录下的coin/coin1.in 和 coin/coin1.ans。

【输入输出样例 1 说明】

骑士第一天收到一枚金币；第二天和第三天，每天收到两枚金币；第四、五、六天，每天收到三枚金币。因此一共收到 1+2+2+3+3+3=14 枚金币。

【输入输出样例 2】

coin.in	**coin.out**
1000	29820

见选手目录下的coin/coin2.in 和 coin/coin2.ans。

【数据说明】

对于 100%的数据，1 ≤ K ≤ 10,000。

扫雷游戏

(mine.cpp/c/pas)

【问题描述】

扫雷游戏是一款十分经典的单机小游戏。在 n 行 m 列的雷区中有一些格子含有地雷（称之为地雷格），其他格子不含地雷（称之为非地雷格）。玩家翻开一个非地雷格时，该格将会出现一个数字——提示周围格子中有多少个是地雷格。游戏的目标是在不翻出任何地雷格的条件下，找出所有的非地雷格。

现在给出n行m列的雷区中的地雷分布，要求计算出每个非地雷格周围的地雷格数。

注：一个格子的周围格子包括其上、下、左、右、左上、右上、左下、右下八个方向上与之直接相邻的格子。

【输入格式】

输入文件名为 mine.in。

输入文件第一行是用一个空格隔开的两个整数n和m，分别表示雷区的行数和列数。

接下来 n 行，每行 m 个字符，描述了雷区中的地雷分布情况。字符’*’表示相应格子是地雷格，字符’?’表示相应格子是非地雷格。相邻字符之间无分隔符。

【输出格式】

输出文件名为 mine.out。

输出文件包含 n 行，每行 m 个字符，描述整个雷区。用’*’表示地雷格，用周围的地雷个数表示非地雷格。相邻字符之间无分隔符。

【输入输出样例 1】

mine.in	**mine.out**
3 3 *?? ??? ?*?	*10 221 1*1

见选手目录下的 mine/mine1.in 和 mine/mine1.ans。

【输入输出样例 2】

mine.in	**mine.out**
2 3 ?*? *??	2*1 *21

见选手目录下的 mine/mine2.in 和 mine/mine2.ans。

【输入输出样例 3】

见选手目录下的 mine/mine3.in 和 mine/mine3.ans。

【数据说明】

对于 100%的数据，$1 \leqslant n \leqslant 100$，$1 \leqslant m \leqslant 100$。

珠心算测验(count.cpp)

【问题描述】

珠心算是一种通过在脑中模拟算盘变化来完成快速运算的一种计算技术。珠心算训练， 既能够开发智力，又能够为日常生活带来很多便利，因而在很多学校得到普及。

某学校的珠心算老师采用一种快速考察珠心算加法能力的测验方法。他随机生成一个正 整数集合，集合中的数各不相同，然后要求学生回答：其中有多少个数，恰好等于集合中另 外两个（不同的）数之和?

最近老师出了一些测验题，请你帮忙求出答案。

【输入】

输入文件名为 count.in。

输入共两行，第一行包含一个整数 *n*，表示测试题中给出的正整数个数。

第二行有 *n* 个正整数，每两个正整数之间用一个空格隔开，表示测试题中给出的正整数。

【输出】

输出文件名为 count.out。

输出共一行，包含一个整数，表示测验题答案。

【输入输出样例】

count.in	count.out
4 1 2 3 4	2

【样例说明】

由 1+2=3，1+3=4，故满足测试要求的答案为 2。注意，加数和被加数必须是集合中的两个不同的数。

【数据说明】

对于 100%的数据，$3 \leqslant n \leqslant 100$，测验题给出的正整数大小不超过 10,000。

比例简化（ratio.cpp/c/pas）

在社交媒体上，经常会看到针对某一个观点同意与否的民意调查以及结果。例如，对某 一观点表示支持的有 1498 人，反对的有 902 人，那么赞同与反对的比例可以简单的记为1498:902。

不过，如果把调查结果就以这种方式呈现出来，大多数人肯定不会满意。因为这个比例的数值太大，难以一眼看出它们的关系。对于上面这个例子，如果把比例记为 5:3，虽然与真实结果有一定的误差，但依然能够较为准确地反映调查结果，同时也显得比较直观。

现给出支持人数 A，反对人数 B，以及一个上限 L，请你将 A 比 B 化简为 A'比 B'，要求在 A'和 B'均不大于 L 且 A'和 B'互质(两个整数的最大公约数是1) 的前提下，$A'/B' \geq A/B$ 且 $A'/B' - A/B$ 的值尽可能小。

【输入】

输入文件名为 ratio.in。

输入共一行，包含三个整数 A，B，L，每两个整数之间用一个空格隔开，分别表示支持人数、反对人数以及上限。

【输出】

输出文件名为 ratio.out。

输出共一行，包含两个整数 A'，B'，中间用一个空格隔开，表示化简后的比例。

【输入输出样例】

ratio.in	ratio.out
1498 902 10	5 3

【数据说明】

对于 100%的数据，$1 \leqslant A \leqslant 1{,}000{,}000$，$1 \leqslant B \leqslant 1{,}000{,}000$，$1 \leqslant L \leqslant 100$，$A/B \leqslant L$ 。

记数问题

(countt.cpp/c/pas)

【问题描述】

试计算在区间 1 到 n 的所有整数中，数字x（0 ≤ x ≤ 9）共出现了多少次？例如，在 1到 11 中，即在 1、2、3、4、5、6、7、8、9、10、11 中，数字 1 出现了 4 次。

【输入】

输入文件名为 countt.in。

输入共 1 行，包含 2 个整数 n、x，之间用一个空格隔开。

【输出】

输出文件名为 countt.out。

输出共 1 行，包含一个整数，表示 x 出现的次数。

【输入输出样例】

countt.in	countt.out
11 1	4

【数据说明】

对于 100%的数据，1≤ n ≤ 1,000,000，0 ≤ x ≤ 9。

- # 表达式求值

(expr.cpp/c/pas)

【问题描述】

给定一个只包含加法和乘法的算术表达式，请你编程计算表达式的值。

【输入】

输入文件为 expr.in。

输入仅有一行，为需要你计算的表达式，表达式中只包含数字、加法运算符“+”和乘法运算符“*”，且没有括号，所有参与运算的数字均为 0 到 2^{31}-1 之间的整数。输入数据保证这一行只有 0~9、+、*这12 种字符。

【输出】

输出文件名为 expr.out。

输出只有一行，包含一个整数，表示这个表达式的值。**注意：当答案长度多于 4 位时，**请只输出最后 4 位，前导 0 不输出。

【输入输出样例 1】

expr.in	**expr.out**
1+1*3+4	8

【输入输出样例 2】

expr.in	**expr.out**
1+1234567890*1	7891

【输入输出样例 3】

expr.in	**expr.out**
1+1000000003*1	4

【输入输出样例说明】

样例 1 计算的结果为 8，直接输出 8。

样例 2 计算的结果为 1234567891，输出后 4 位，即 7891。

样例 3 计算的结果为 1000000004，输出后 4 位，即 4。

【数据范围】

对于 30%的数据，0≤表达式中加法运算符和乘法运算符的总数≤100；
对于 80%的数据，0≤表达式中加法运算符和乘法运算符的总数≤1000；
对于 100%的数据，0≤表达式中加法运算符和乘法运算符的总数≤100000。