

2019“信息与未来”小学生夏令营 程序设计小能手活动题集

共6题；全自动评测(键盘输入、屏幕输出)，请严格遵照题目指定格式输入输出。

测试点运行超过 0.5 秒或使用超过 128 MiB 内存将被判错。

1. 幸运素数 (15分，文件名：p1.cpp, p1.pas, p1.bas)

素数，又称质数，是指除 1 和其自身之外，没有其他约数的正整数。例如 2、3、5、13 都是素数，而 4、9、12、18 则不是。特别地，规定 1 不是素数 (因此自然数的质因数分解就是唯一的)。

如果一个数本身是素数，并且把最低位删除后得到的数仍是素数、再把最低位删除后得到的数仍是素数.....如此往复，直到得到一个一位素数，我们就称它是“幸运素数”。以 233 为例：

- 233 本身是素数；
- $23 = \lfloor 233/10 \rfloor$ 是素数；
- $2 = \lfloor 23/10 \rfloor$ 是素数，

因此 233 是“幸运”素数。而 211 则不是幸运素数：虽然 211 是素数，但 21 不是素数。请编程求出一定范围内的所有幸运数字。

输入格式

输入一行两个用空格分隔的正整数 m, n ，表示我们希望求幸运素数的范围。

输出格式

从小到大输出 $m, m+1, m+2, \dots, n$ 中所有的幸运素数，每行输出一个。

样例数据

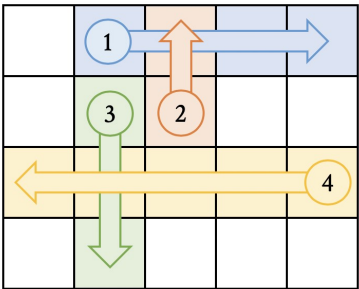
样例输入	样例输出
6 30	7 23 29

数据规模

对于 100% 的测试数据，有 $1 \leq m \leq n \leq 9999$ 。

2. 粉刷矩形 (15分，文件名： p2.cpp, p2.pas, p2.bas)

在一个 $n \times m$ 的矩形上进行粉刷工作 (初始时，矩形的格子无色)，每次选择一个起点和一个方向，将连续的格子刷成同样的颜色，直到碰到矩形边界后停止；已经有颜色的格子中的颜色将会被覆盖，例如右图展示了依次在 (1, 2)、(2, 3)、(2, 2)、(3, 5) 格子进行向右、向上、向下、向左粉刷后每个格子的颜色。



请你编程模拟粉刷矩形的过程，并输出每个格子的颜色。

输入格式

输入第一行三个整数 n, m, k 表示矩形的大小是 $n \times m$ 、共有 k 次粉刷操作。

接下来 k 行，每行都是空格分隔的两个整数和两个字符，分别表示粉刷起点的行号、列号、粉刷的颜色 (用小写字母表示)、粉刷的方向 (R, U, L, D 分别表示向右、向上、向左、向下)。

输出格式

输出 n 行，每行 m 个字符 (字符之间无空格)，表示粉刷后每个格子的颜色。无色的格子输出“.”(点)。

样例数据

样例输入	样例输出
4 5 4	
1 2 a R	.abaa
2 3 b U	.xb..
2 2 x D	ttttt
3 5 t L	.x...

数据规模

对于 100% 的测试数据，有 $1 \leq n, m, k \leq 50$ 。

3. 新斐波那契数列 (15分, 文件名: p3.cpp, p3.pas, p3.bas)

给定正整数 a ($a \geq 1$), 新斐波那契数列 f_a 按如下方式定义:

- $f_a(1) = 1$;
- $f_a(2) = a$;
- $f_a(n) = f_a(n-1) + f_a(n-2)$ ($n > 2$);

例如, 给定 $a = 4$, 有 $f_4(1) = 1, f_4(2) = 4, f_4(3) = 5, f_4(4) = 9, f_4(5) = 14, \dots$. 现在已知新斐波那契数列中的一项 x , 但并不知道 n 和 a 的值是多少。请你求出所有可能的 n, a ($n \geq 2$) 满足 $f_a(n) = x$ 。

输入格式

你需要在一个测试数据中处理多个新斐波那契数列问题。输入第一行 T 表示问题的数量。

接下来 T 行, 每行一个整数: 待求解的 x 。

输出格式

对于每个新斐波那契数列问题, 按照 n 从小到大的顺序, 输出所有可能的 n, a 满足 $f_a(n) = x$ 。每行输出一对 n 和 a , 由一个空格分隔。

样例数据

样例输入	样例输出
2 9 123	2 9
	3 8
	4 4
	2 123
	3 122
	4 61
	6 24
	10 3

数据规模

对于 60% 的测试数据, 有 $x \leq 10^6$ 。

对于 100% 的测试数据, 有 $2 \leq x \leq 10^9$ 、 $T \leq 20$ 。

4. 堆栈计算机 (15分，文件名： p4.cpp, p4.pas, p4.bas)

有一种新型的堆栈计算机，计算机的内存是一个初始为空的数列，计算机支持三种操作：

- 1 - 将整数 1 放入数列的尾部。任意时刻都可执行此操作。
- dup - 将数列尾部的数字复制一份，放入数列尾部。只有数列非空时才能执行此操作。
- add - 取出 (并删除) 数列尾部的两个数字，相加后放回数列的尾部。只有数列中至少有两个数字时才能执行此操作。

给定一个正整数 n ，你需要输出一个长度不超过 200 的程序 (操作的序列)，它在执行结束后，恰好得到一个长度为 1 的数列，并且数列中的数字恰好为 n 。右图展示了一个最终得到 8 的程序。

数列	执行的操作					
<table><tr><td></td><td></td><td></td><td></td><td></td></tr></table>						1
<table><tr><td>1</td><td></td><td></td><td></td><td></td></tr></table>	1					1
1						
<table><tr><td>1</td><td>1</td><td></td><td></td><td></td></tr></table>	1	1				add
1	1					
<table><tr><td>2</td><td></td><td></td><td></td><td></td></tr></table>	2					dup
2						
<table><tr><td>2</td><td>2</td><td></td><td></td><td></td></tr></table>	2	2				add
2	2					
<table><tr><td>4</td><td></td><td></td><td></td><td></td></tr></table>	4					dup
4						
<table><tr><td>4</td><td>4</td><td></td><td></td><td></td></tr></table>	4	4				add
4	4					
<table><tr><td>8</td><td></td><td></td><td></td><td></td></tr></table>	8					程序结束
8						

输入格式

输入一行一个正整数 n ，表示希望输出的数字。

输出格式

输出一个若干行 (不超过 200 行，否则判为不正确) 的满足上述要求的程序。如有多种方案，输出任意一种即可。注意“dup”和“add”均为小写。

样例数据

样例输入	样例输出
1	1
8	1 1 add dup add dup add

数据规模

- 对于 60% 的测试数据，有 $n \leq 10^4$ 。
- 对于 100% 的测试数据，有 $1 \leq n \leq 10^9$ 。

5. 鸡兔同笼 (20分, 文件名: p5.cpp, p5.pas, p5.bas)

有三种动物被关在笼子里, 它们分别是:

- 鸡, 每只鸡有一个头、两只脚;
- 三脚猫, 每只三脚猫有一个头, 三只脚;
- 兔子, 每只兔子有一个头, 四只脚。

现在知道笼子中一共有 x 个头、 y 只脚, 但这个信息并不足以推导出鸡、兔和三脚猫的准确数量。这时候就需要编程来帮忙了——请你求出笼子里分别至少、至多有几只兔子。

输入格式

你需要在一个测试数据中处理多个鸡兔同笼问题。输入第一行 T 表示问题的数量。

接下来 T 行, 每行两个用空格分隔的正整数 x, y , 表示头和脚的数量。

输出格式

对于每个问题, 输出一行两个空格分开的整数, 表示笼子里至少和至多有几只兔子。输入数据保证每个问题至少有一种鸡、三脚猫和兔子的合法组合。

样例数据

样例输入	样例输出
4	0 0
6 12	3 4
6 21	6 6
6 24	2965 7655
12345 40000	

数据规模

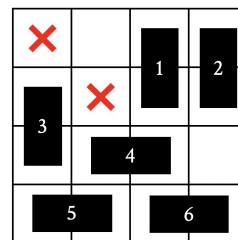
对于 50% 的测试数据, 有 $1 \leq x \leq 10^6$ 。

对于 100% 的测试数据, 有 $1 \leq x, y \leq 10^9$ 、 $1 \leq T \leq 20$ 。

6. 方格覆盖 (20分, 文件名: p6.cpp, p6.pas, p6.bas)

给定一个 $n \times n$ 的矩形, 其中从左上角开始, 对角线上连续的 k 个格子中有障碍物。你可以把若干 1×2 的小矩形放置到该大矩形中, 要求是放置的两个小矩形不能占据相同的格子, 且不能碰到障碍物。例如右图是 $n = 4, k = 2$ 的例子, 我们放置了 6 个 1×2 的小矩形。

给定 n, k , 请你输出一个方案, 使得放置的 1×2 小矩形尽可能多。可以证明, $n = 4, k = 2$ 时, 至多只能放置 6 个小矩形。



输入格式

输入一行两个用空格分隔的正整数 n, k , 表示矩形的大小和障碍物的数量。

输出格式

输出 n 行, 每行 n 个整数 (用任意数量的空格分隔)。放置的小矩形分别用 1, 2, ... 编号。不放置小矩形的格子输出 0。如有多种最优方案 (放置最多数量的矩形), 输出任意一种即可。

样例数据

样例输入	样例输出
4 2	0 0 1 2 3 0 1 2 3 4 4 0 5 5 6 6
5 3	0 8 8 9 10 1 0 0 9 10 1 3 0 0 7 2 3 5 5 7 2 4 4 6 6

数据规模

对于 50% 的测试数据, 有 $1 \leq n \leq 10$;

对于 100% 的测试数据, 有 $1 \leq k \leq n \leq 50$ 。