

Problem 1. equation

Input file: `equation.in`
Output file: `equation.out`
Time limit: 1 second
Memory limit: 256 MB

Mr. Hu 又来让你帮忙解方程了。

方程是这样的：

$$x_1 + x_1 + x_3 + \cdots + x_n = m \quad (x_i \geq 0 \ \forall 1 \leq i \leq n)$$

Mr. Hu 希望你求出这个 n 元一次方程的整数解有多少个，因为解的个数有可能变得很大，所以 Mr. Hu 只需要你输出解的个数取模于 mod 。

Input

第 1 行，包含一个整数： T ，表示询问个数

接下来 T 行，每行包含三个整数： $n \ m \ mod$

Output

输出 T 行，每行输出解的个数模对应 mod

Sample

<code>equation.in</code>	<code>equation.out</code>
1 2 3 13	4

Note

样例中，解分别是： $(3, 0), (2, 1), (1, 2), (0, 3)$

- 对于 30% 的数据， $1 \leq n, m \leq 6$ ， $mod = 10^8 + 7$ ， $T = 1$
- 对于 70% 的数据， $1 \leq n, m \leq 10^3$ ， $n + m \leq mod \leq 10^8 + 7$ ， mod 是一个素数， $1 \leq T \leq 100$
- 对于余下 30% 的数据， $1 \leq n, m \leq 10^3$ ， $n + m \leq p, q \leq 10^4$ ， $mod = pq$ ， p, q 是素数， $1 \leq T \leq 10^3$

Problem 2. power

Input file: `power.in`
Output file: `power.out`
Time limit: 1 second
Memory limit: 256 MB

Mr. Hu 打算考一道比较显然的题目，低头一想，就有了这道题。

Mr. Hu 需要你计算：

$$3^n \bmod 10^9 + 8$$

是不是很简单啊。^_^

Input

只有一行，一个数 n 。

Output

输出结果。

Sample

<code>power.in</code>	<code>power.out</code>
3	27

Note

- 对于 10% 的数据， $1 \leq n \leq 10^6$
- 对于 30% 的数据， $1 \leq n \leq 10^{18}$
- 对于 70% 的数据， $1 \leq n \leq 10^{1000}$
- 对于 100% 的数据， $1 \leq n \leq 10^{100000}$

Problem 3. comb

Input file: `comb.in`
Output file: `comb.out`
Time limit: 1 second
Memory limit: 256 MB

一天, Mr. Hu 对组合数产生了兴趣, 他想要知道满足下面条件的数 i 有多少个:

$$\gcd\left(\binom{n}{i}, p\right) = p \quad (0 \leq i \leq n)$$

其中 p 是素数。

Input

第 1 行, 2 个整数: n p 。

Output

输出所求。

Sample

<code>comb.in</code>	<code>comb.out</code>
5 2	1

Note

样例中, 一共有 5 个组合数: 1, 5, 10, 5, 1, 其中和 2 公约数为 2 的数只有一个 10, 故输出 1。

- 对于 30% 的数据, 满足 $1 \leq n, p \leq 10^3$
- 对于 100% 的数据, 满足 $1 \leq n, p \leq 10^{18}$

Problem 4. derange

Input file: `derange.in`
Output file: `derange.out`
Time limit: 1 second
Memory limit: 256 MB

Mr. Hu 需要给机房的 n 位同学重新安排位置, 为了使换位的效果好, 需要保证每位同学在排位前和排位后所在的位置都不同。

Mr. Hu 想知道有多少种可能的排位方法。因为排位数可能很大, 所以你只需要输出它模 $10^9 + 7$ 的结果。

Input

输入文件中只有 1 个数 n , 表示学生人数。

Output

输出可能的排位数取模后的结果。

Sample

<code>derange.in</code>	<code>derange.out</code>
3	2

Note

样例说明: 假设一开始三个人的位置为 $(1, 2, 3)$, 那么合法的换位结果为: $(2, 3, 1)$ 和 $(3, 1, 2)$ 。

- 对于 30% 的数据, $1 \leq n \leq 9$
- 对于 70% 的数据, $1 \leq n \leq 10^3$
- 对于 100% 的数据, $1 \leq n \leq 10^6$

Problem 5. mulfuc

Input file: `mulfuc.in`
Output file: `mulfuc.out`
Time limit: 1 second
Memory limit: 256 MB

Mr. Hu 想让大家了解一下积性函数。

积性是函数的一种重要性态，就像单调性、周期性一样。

一个函数 $f(n)$ 如果是积性的，当且仅当：

$$f(nm) = f(n)f(m) \quad (\gcd(m, n) = 1)$$

如果 $f(n)$ 是定义在 Z^+ 上的积性函数，这样定义 Z^+ 上的 $g(n)$ ：

$$g(n) = \sum_{d|n} f(d)$$

那么可以证明 $g(n)$ 也是一个积性函数。

而在积性函数中，经常使用到以下几个重要的积性函数（容易证明它们都是积性的）：

- $\tau(n)$ 表示正整数 n 的正因子个数。
- $\sigma(n)$ 表示正整数 n 的正因子和。
- $\mu(n)$ 表示正整数 n 的 Mobius 函数值。
- $\varphi(n)$ 表示正整数 n 的欧拉函数值，即 $[1, n]$ 中与 n 互质的数的个数。

其中 Mobius 函数的定义如下：

$$\mu(n) = \begin{cases} 1, & n = 1 \\ (-1)^r & n = p_1 p_2 \dots p_r \ (p_1 < p_2 < \dots < p_r) \ p_i \text{ is prime} \\ 0 & \text{other cases} \end{cases}$$

现在再定义两个函数：

$$I(n) = \sum_{d|n} \mu(d)$$

$$E(n) = \sum_{d|n} \varphi(d)$$

现在 Mr. Hu 需要你去求出上面六个函数在 $1 \leq n \leq 10^6$ 范围内的值。

Input

第 1 行包含一个整数 opt ，表示 Mr. Hu 需要你求出的函数的标号。对应关系是：

opt	1	2	3	4	5	6
函数	τ	σ	μ	φ	I	E

Output

输出 1 行，包含 opt 对应的函数在 $[1, 10^6]$ 范围内的函数值，两个函数值之间用一个空格隔开。

Sample

<code>mulfunc.in</code>	<code>mulfunc.out</code>
1	1 2 2 3 2 4 ...
<code>mulfunc.in</code>	<code>mulfunc.out</code>
2	1 3 4 7 6 12 ...
<code>mulfunc.in</code>	<code>mulfunc.out</code>
3	1 -1 -1 0 -1 1 ...
<code>mulfunc.in</code>	<code>mulfunc.out</code>
4	1 1 2 2 4 2 ...

Note

上面的样例输出只给出了前面几项，后面用省略号代替了，你需要输出全部的项。

本题满分 100 分，分为 6 个测试点，每个测试点 100/6 分

- 对于第 1 个测试点， $opt = 1$ 。
- 对于第 2 个测试点， $opt = 2$ 。
- 对于第 3 个测试点， $opt = 3$ 。
- 对于第 4 个测试点， $opt = 4$ 。
- 对于第 5 个测试点， $opt = 5$ 。
- 对于第 6 个测试点， $opt = 6$ 。