

## A.闰年产生的原因

### 题目概括:

“地球公转一圈”实际是 365 天 5 小时 48 分 46 秒，每 4 年就多了 23 小时 15 分 04 秒，因此每 4 年就调整 1 天。这样每 100 年又少了 5 小时 16 分 40 秒，所以每 400 年要再“闰”回来，求到西元 $m$  年的最后一秒误差（假设西元 1 年 1 月 1 日是“原点”，即无误差）

### 输入输出描述:

输入:一个正整数  $m$  ( $1 < m < 3000$ )

输出:三个整数（时，分，秒）表示误差的时间

### 思路:

定三个变量，用循环模拟出地球前  $m$  天的时间，每循环一次让时，分，秒分别加上 5，48，46（误差），如果年份（循环次数）是 4 的倍数，那就让时间减少一天（24 时），但是这样会把 100 也当成闰年，所以如果年份（循环次数）是 100 的倍数，那就让时间增加一天，还要判断年份（循环次数）是否为 400 的倍数，如果是的话要把时间减回去（让时间减少一天）。

循环结束之后还要为了防止分钟和秒数大于六十，我们还要让分钟加上秒数除以六十的商，让秒数等于秒数除以六十的余数，让小时加上分钟除以六十的商，让分钟等于分钟除以六十的余数。

最后输出时，分，秒。

# B.找出叛徒

## 题目概括

神盾局最近泄露了很多高级文件，可能存在叛徒。由于神盾局内部人员的 id 都是加密过的，加密函数为  $f(x) = x + x$  各位数字之和， 现在已有的情报是 id 加密后为  $f(x)$  的人为叛徒，为了确定叛徒，神盾局找到了技术高超的你，希望你能解决这个问题。由于可能存在多个答案，请把所有答案按照升序依次输出

## 输入输出描述

输入一个数代表  $f(x)$  ( $f(x) \leq 10^{18}$ )

第一行输出答案个数(可能为 0)，第二行按照升序输出所有答案

## 思路

首先我们先定义一个变量 id 储存输入和一个计数变量 sum，使  $sum = 0$ ，再定义一个数组 ans 来储存答案，而根据题目可知，输入的数最多为  $10^{18}$ ，也就是说加密后的数与加密前的数最多相差 162(18 个 9)，我们可以把它约等于为 200，那么数组的长度我们就可以定义为 200

然后我们定义一个循环，使 i 从 id - 200 开始，直到 id 结束。在循环中我们用 if 判断  $f(i)$  是否等于 id，这时我们可以写一个函数 check()，传入两个参数 i 和 id 来判断，如果为 true 的话，就让  $ans[sum]$  等于 i，再让  $sum++$ 。

最后我们输出 sum，再按顺序输出 ans

# C.击杀怪兽

## 题目概括

Mas 正在玩另一款电脑游戏，他操控的角色在杀死一些怪物。有  $n$  只怪物编号  $1 \sim n$ ，其中  $i$  只怪物的初始生命值为  $h_i$ 。Mas 的角色有一个能力，可以对当前生命值最高的怪物造成  $k$  点的生命值伤害。如果有几只生命值相同的怪物，那么 Mas 将选择编号较小的那只。如果怪物的生命值在 Mas 使用能力后小于或等于 0，那么它就会死亡。

Mas 会不断的使用他的能力直到所有怪物死亡。请你输出怪物死亡的顺序。

## 输入输出描述

第一行输入两个空格分隔的正整数  $n, k$  第二行输入  $n$  个空格分隔的整数，分别表示  $h_i$

输出一行  $n$  个整数空格分隔的整数，其中第  $i$  个整数表示第  $i$  只怪兽被杀死的顺序

## 思路

让我们先举一个例子，怪物生命值为 6 8 4，Mas 伤害为 3。 $[6 \ 8 \ 4] \rightarrow [6 \ 5 \ 4] \rightarrow [3 \ 5 \ 4] \rightarrow [3 \ 2 \ 4] \rightarrow [3 \ 2 \ 1]$  这时我们可以发现所有的怪物都可以被 Mas 一击杀死，也就是说前面的伤害对结果不重要，重要的是最后所有的怪物都可以被一击杀死的时候，那么我们就可以让所有怪物的生命值取余  $k$ (伤害)，但有时怪物的生命值可能为 0，那我们就应该把它设为  $k$ (伤害)

然后我们定义一个结构体数组，里面包含两个变量  $id$  和  $h$ 。然后再对结构体数组赋值，然后再用  $sort$  对结构体数组排序，其中  $cmp$  为如果血量相等，返回  $id < id$ ，否则返回  $血量 > 血量$

最后我们按顺序输出结构体数组中的  $id$

## D. 无限加密

### 题目概括:

Mas 设计了一种密文加密策略: 初始时得到密文  $S$  先将文本复制一份, 再将复制的文本循环右移一位, 最后将移动后的文本接在原文后。现在给出原文  $S$ , 请你求出进行了若干次变换后  $S$  的第  $n$  个字符。

### 输入输出描述:

输入: 第一行一个字符串  $S$  ( $1 \leq |S| \leq 30$ ) 第二行一个整数  $n$  ( $1 \leq n \leq 10$  的  $18$  次方)

输出: 一个字符, 表示进行了若干次变换后  $S$  的第  $n$  个字符。

### 思路:

我们用文中的 MAS 举例, 发现每个位置的值都等于  $n-S$  的长度乘上  $x$  最大的不大于  $n$  二的  $m$  次方-1。

首先我们可以写一个递归模拟字符串中  $n$  的每一次向前移动, 在递归最前方写上  $n$  是否小于  $S$  的大小, 如果小于, 返回字符串的第  $n$  项, 否则循环找出最大的不大于  $n$  二的  $m$  次方中  $m$  的值, 然后判断如果  $n-m-1$  为 0, 递归  $n-1$ , 否则递归  $n-m-1$ 。

然后输入  $S$  和  $n$ , 输出递归函数的返回值。