

POJ1220 【基础】

题目大意：

给出某一进制下的一个数，转换为另一进制下的数字并输出。进制序列为 0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz，即最大 62 进制。

输入：

输入包含若干组测试数据。第一行有一个整数 n ，表示有 n 组测试数据。接下来有 n 行，每一行前面有两个用一个空格分隔的整数 $base1$ ， $base2$ ，分别表示给出的数字的进制和目标转换进制。 $base2$ 后有一个空格，然后跟着一个字符串，表示给出的数字。

输出：

每一组测试数据输出两行，第一行是一个整数 $base1$ +一个空格+原来给出的数字。第二行是一个整数 $base2$ +一个空格+转换后的数字。两组输出之间有一个空行分隔。

题解：

这题是标准的进制转换，涉及到一点高精度除单精度的除法。原来做进制转换我一般都是以 10 进制作为过渡，但是现在发现其实可以一步到位。用一个数组，每一个数组单元保存原来的数的对应位的进制序列编号，比如“EVA”在数组 s 中保存为 $[14][31][10]$ ，然后每次做高精度除单精度的时候照旧即可。这相当于高精度不是一个数存一位而是一个数存多位一样。最后输出注意记得逆序。

POJ1565 【基础】

题目大意、输出输出要求原题中有。

题解：

一题变形的进制转换题，将每一位的单位从 2^k 变成了 $2^{(k+1)-1}$ ，并且每一位有 0、1、2 三种数字。按进制转换题来做就好了。

POJ3299 【基础】

题目大意：

给出 temperature、dewpoint 和 humidex 中的两个数据，根据公式算出第三个数据。

其中，

$\text{humidex} = \text{temperature} + h$

$h = (0.5555) \times (e - 10.0)$

$e = 6.11 \times \exp [5417.7530 \times ((1/273.16) - (1/(\text{dewpoint}+273.16)))]$

$\exp[x]$ 表示 e^x

输入：

有若干组测试数据。每一组测试数据第一行第一个字母如果是“E”，表示测试数据结束，否则每一组测试数据应该有一行，包含两个字母和两个小数，如“T 30 D 15”，T表示后面的数据对应 temperature，D表示后面的数据对应 dewpoint，H表示后面的数据对应 humidex。

输出：

每一组测试数据输出一行，包含 T、H、D 三个参数，格式如“T 30.0 D 15.0 H 34.0”，小数要求输出到小数点后一位。

题解：

简单的转换题，写起来有些麻烦。

POJ1844 【基础】

题目大意：

给出一个数 s ($1 \leq s \leq 100000$)，从 1 到 n 个数，每个数前面可以是负号或者是正号，这样累加起来，结果可以等于 s ，求最小的 n 。

输入：

一行，一个整数 s 。

输出：

一行，一个整数 n 。

题解：

显然，若这 n 个数加起来小于 s ，必不符合题意。当 $\text{sigma}(1..n) \geq s$ 时，我们可以看作是先把这 n 个数相加，再从中把一些加号变成减号。假设变号的这些数字和为 a ，则变号后得到的数比 $\text{sigma}(1..n)$ 小 $2a$ 。也就是说，如果 n 是问题的解，必然有 $\text{sigma}(1..n) - s$ 为偶数，并且因为 $\text{sigma}(1..n) - n < n$ 必然存在 $k < n$ 令 $2*k = \text{sigma}(1..n) - n$ 。因此 n 从 1 向上枚举，找到符合的 n 以后直接输出 n 就可以了。

POJ1019 【基础】

题目大意：

有一串数字串，其规律为

1 12 123 1234 12345 123456 1234567.....

输入位置 n ，计算这一串数字第 n 位是什么数字。注意要求输出的是某一位的数字，而不是数。比如 $n=11$ 的时候输出的是 12345 中第一位的 1。

输入：

第一行有一个整数 t ，表示有 t 组测试数据。接下来有 t 行，每一行有一个整数 n 。

输出：

对每一组测试数据，输出一行，有一个数字，即所求的数字。

题解：

很简单的模拟题。先打表打出一个数组 sum ， $sum[i]$ 表示到第 i 组数的时候总数字串的长度。第 i 组的数字是 123456..... i ，因此第 i 组数字的长度=第 $i-1$ 组数字的长度+ i 的位数。然后做两次查询：首先确定给出的 n 在第 j 组数的位置，然后确定在第 j 组位置中的第 k 个数字。如果刚好是第 k 个数字的最后一位，那么直接模 10 输出，否则先移位再模 10。

POJ3210 【基础】

题目大意：

现在有 n 枚硬币，其初始状态不确定，问最少要改变多少枚硬币的状态，才能让所有的硬币正面朝上或者反面朝上？

输入：

有若干组测试数据，每一组占一行，有一个整数 n ($n \leq 10000$)， $n=0$ 时测试数据结束。

输出：

每一组测试数据输出一行：如果存在 x 满足改变 x 枚硬币的状态一定能让所有硬币正面/反面朝上，那么输出 x ；如果没有满足的解，输出 “No Solution!”。

题解：

1、如果 n 为偶数：

(1) 如果距离目的状态有偶数枚硬币需要被翻动，那么翻动的次数一定是偶数次（翻动了偶数次到目的状态后，可以任选若干枚每枚再翻动 2 次，所以一定是偶数次）；

(2) 如果距离目的状态有奇数枚硬币需要被翻动，那么翻动的次数一定是奇数次（原理同上），与 (1) 矛盾，故 n 为偶数时一定无解。

2、如果 n 为奇数：

正面或者反面的数量一定有一个是偶数，将这奇数个硬币翻一次面，则一定可以达到目的状态。所以取所有小于 n 的偶数中最大的一个，即 $n-1$ 即可。

这题主要是一个简单的推理判断，代码写起来完全没有难度，我是直接在提交页面里写了就交的……

HDU 1868 【中等】

题目大意：

给出一个四字节整数 N ，问 N 能写成多少个形如 $a+(a+1)+(a+2)+\dots+(a+i-1)$ 的连续整数之和，其中 $i>1$ 。

输入：

一行，一个整数 N 。

输出：

一行，一个整数，表示有多少种相加方法。

题解：

$n=a+(a+1)+\dots+(a+i-1)=i*a+(1+2+\dots+(i-1))=i*a+(i*(i-1)/2)$ ， $a=(n-(i*(i-1)/2))/i$ ，所以枚举 i 满足 $(2*n+i-i*i)\%(2*i)==0$ 看最后有多少个即可。

POJ3263 【中等】

题目大意：

N 头牛排成一行，已知最高的牛的高度 H 以及位置 I ，有 R 个信息 (x, y) ，即第 x 头牛向某一个方向最多可以看到第 y 头牛，即位于 $[x+1, y-1]$ 区间内的牛都比它小，问每头牛最大可能高度是多少。所有的牛从 1 开始编号。给出的区间可能有重复，但一定满足条件。（ $1\leq N\leq 10000$ ， $1\leq H\leq 10000$ ， $0\leq R\leq 10000$ ）

输入：

第一行有四个空格分隔的整数 N 、 L 、 H 、 R 。

接下来有 R 行，每一行有两个整数 x_i y_i ，表示一个信息。

输出：

N 行，每一行有一个整数，即第 i 头牛可能的最大身高。

题解：

区间信息维护统计。一开始我差点想杀鸡用牛刀地上 Splay 或者线段树，后来想想大可不必。因为要求的是可能的最大高度，而不是一开始有初始值然后再维护区间，所以只要统计一下某个点被多少个区间覆盖了就知道这个点上的牛最高应该是最高牛的高度减去区间覆盖数。因为如果一头牛在 x 能看到 y （设 $y > x$ ），那么 $[x+1, y-1]$ 中的牛一定比 x 和 y 都要小。这样一来，只要对区间按先左侧降序后右侧降序的优先级来排序，然后去重，根据此得到一个数组记录进入当前点的时候是否进入或离开一个区间。设开始时的高度就是最大高度，对这个数组从头到尾扫一次，如果是的话对应当前高度-1 或者+1，然后输出即可。

P0J1650 【中等】

题目大意：

给出一个小数 A ($0.1 \leq A \leq 10$)，以及一个范围 L ($1 \leq L \leq 100000$)，求 p 、 q 属于 $[0, L]$ ，使得 p/q 最接近 A 。

输入：

第一行有一个小数 A ，第二行有一个整数 L 。

输出：

一行，两个整数，分别是分子和分母。

题解：

暴力的话应该会超时，因此我们使用追赶法。一开始的时候设 $p=q=1$ ，若 $p/q > A$ 则增大 q ，否则增大 p ，直到限制范围为止，中间保存最小的差值。

这里有一个小坑，那就是插值的初始值应该是 10，否则可能会出现 $A > L$ 的情况出现，那么就无法处理了。我在这里 WA 了一次。

POJ2085 【中等偏上】

题目大意：

由 n 个正整数 $1 \sim n$ 组成的序列，规定其逆序数为 $\text{sigma}(i=1 \text{ to } n, \text{第 } i \text{ 位右边比第 } i \text{ 位小的数字的个数})$ 。现在给出 n ，并且给出逆序数 m ，求符合要求的字典序最小的序列。

输入：

有若干组测试数据，每一组测试数据有一行，包含两个整数 n 和 m
($1 \leq n \leq 50000, 1 \leq m \leq n(n-1)/2$)。 $n=m=-1$ 时测试数据结束。

输出：

每一组测试数据输出一行，包含 n 个空格分隔的整数，即所求的序列。

题解：

这题颇费一点脑筋，我想了很久没想出来只好上网查了，看了几篇题解才看到有一篇看得懂的。下面给出一个贪心的解法：

- 1、如果逆序数为 m ，那么正序数为 $n(n-1)/2 - m$ 。如果要字典序最小，那么我们需要让小的数尽可能排在前面，换言之尽量保持前面从 1 开始进行升序；
- 2、因为 p 个数的逆序数最大为 $p(p-1)/2$ ，所以我们先算出最小的 p ，令 $p(p-1)/2 \geq m$ 成立，那么前面 $1 \sim n-p$ 个数就是从 1 开始到 $n-p$ ；
- 3、因为后面的 p 个数最大的逆序数为 $p-1 + p-2 + p-3 + \dots + 1 + 0$ ，如果我们把第 i 项抽出来放到第一位，那么逆序数为 $p-i + p-2 + p-3 + \dots + p-i+1 + p-i-1 + p-i-2 + \dots + 1 + 0$ ，而需要字典序最小，所以把 $n-p(p-1)/2+m$ 抽出来放到第 p 位；
- 4、后面的 $p-1$ 位必须产生最大的逆序数，所以在剩下的数里完全降序即可。