**董与黑的碰撞-假想**

时间限制：2000MS

**描述：**

为了能够打赢和黑暗势力的战斗，董先森整天埋头苦思如何应对。这时候一个叫npusjx的谋士提出了一个叫“自动AC机”的假想，并且声称它可以完美地将所有题目AC。

“自动AC机”的主要原理是这样的：

①首先，采用长寿村的神秘方法，获取评测系统所有输入数据（方法过于玄妙，在此不作详述）。假设共有n组数据。

②知道了输入数据之后，仍然想不到解法，但却可以将每组数据的答案缩小到某个范围内。假设第i组数据的答案缩小到了[,]这个范围，并且范围中每个数是答案的概率是等同的。（这里认为,和答案都是整数）

③接下来就是“自动AC机”的核心代码：

**for(i = 1; i <= n; i++) {**

**printf("%d\n", a[i] + k % (b[i] - a[i] + 1));**

**}**

其中k按0,1,2,…的顺序，枚举所有自然数。对于每个k，都提交一次，运气好的话就能AC。

大家对于这一伟大的假想，全都拍手称赞。只有董先森看出来npusjx是在搅局，因为这个“自动AC机”不仅不稳定，而且将收获巨大的罚时。现在已知每次错误提交都会增加20分钟的罚时，请问这个“自动AC机”在AC之前，总共会收到多少罚时？

注意：

①由于总罚时不是确定值，请输出总罚时的数学期望。

②由于答案可能很大，且可能是浮点数，请输出答案整数部分的后8位（包含前缀零）。

③如果存在永远都无法AC的可能，请输出“ongoing”（不包括引号）。

**输入：**

首先输入一个正整数T，表示测试数据的组数。

对于每组数据，第1行输入一个正整数n，接下来n行，每行输入2个整数。

**数据范围：**

1≤T≤8

1≤n≤1000

-≤≤≤

**输出：**

对于每组测试数据，输出8位数字，表示该组测试数据的答案，每组输出占一行。

**样例输入：**

2

1

1 2

2

1 1

1 3

**样例输出：**

00000010

00000020

**提示：**

对于第二组样例，

k=0时，自动AC机输出1、1，

k=1时，自动AC机输出1、2，

k=2时，自动AC机输出1、3，

有1/3的概率答案是1、1，此时直接AC，

有1/3的概率答案是1、2，此时收到20分钟罚时，

有1/3的概率答案是1、3，此时收到40分钟罚时，

最终答案为(20/3+40/3)=20。