1. 矩阵乘法（matrix.cpp/ matrix.c）

总时间限制：1s　　　　内存限制：256MB

【问题描述】

设*A*为 $m \times p$ 的矩阵，*B*为 $p \times n$ 的矩阵，那么称 $m \times n$ 的矩阵*C*为矩阵*A*与*B*的乘积，记作 $C = AB$，其中矩阵C中的第 i 行第 j 列元素可以表示为：

$$(AB)_{ij} = \sum_{k=1}^{p} a_{ik}b_{kj} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{ip}b_{pj}$$

如下所示：

$$C = AB = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix} = \begin{pmatrix} 1\times1+2\times2+3\times3 & 1\times4+2\times5+3\times6 \\ 4\times1+5\times2+6\times3 & 4\times4+5\times5+6\times6 \end{pmatrix} = \begin{pmatrix} 14 & 32 \\ 32 & 77 \end{pmatrix}$$

输入矩阵 A 和 B，编程求 A×B。

【输入格式】

第一行，输入 m，p，n。

接下来是 m 行 p 列，共 m×p 个数，即矩阵 A，矩阵 A 中每个数用一个空格隔开。

接下来是 p 行 n 列，共 p×n 个数，即矩阵 B，矩阵 B 中每个数用一个空格隔开。

【输出格式】

输出 m 行 n 列共 m×n 个数，即矩阵 C，矩阵 C 中每个数用一个空格隔开。

【输入样例】（matrix.in）

```
2 3 2
1 2 3
4 5 6
1 4
2 5
3 6
```

【输出样例】（matrix.out）

```
14 32
32 77
```

【数据规模】

1<=m,n,p<=100

2. 取余运算（mod.cpp）

总时间限制：1s　　　　内存限制：64MB

【问题描述】

输入 b，p，k 的值，求 b^p mod k 的值。其中 b，p，k*k 为长整形数。

【输入格式】

输入一行，三个整数 b，p，k。

【输出格式】

一行，具体参见样例输出。

【输入样例】（mod.in）

2　10　9

【输出样例】（mod.out）

2^10 mod 9=7

---

3. 2011^n （2011.cpp）

总时间限制：1s　　　　内存限制：256MB

【问题描述】

已知长度最大为 200 位的正整数 n，请求出 2011^n 的后四位。

【输入格式】

第一行为一个正整数 k，代表有 k 组数据（k<=200）。

接下来的 k 行，每行都有一个正整数 n，n 的位数不大于 200。

【输出格式】

每一个 n 的结果为一个整数，占一行，若不足 4 位，去除高位多余的 0。

【输入样例】（2011.in）

3
5
28
792

【输出样例】（2011.out）

1051
81
5521