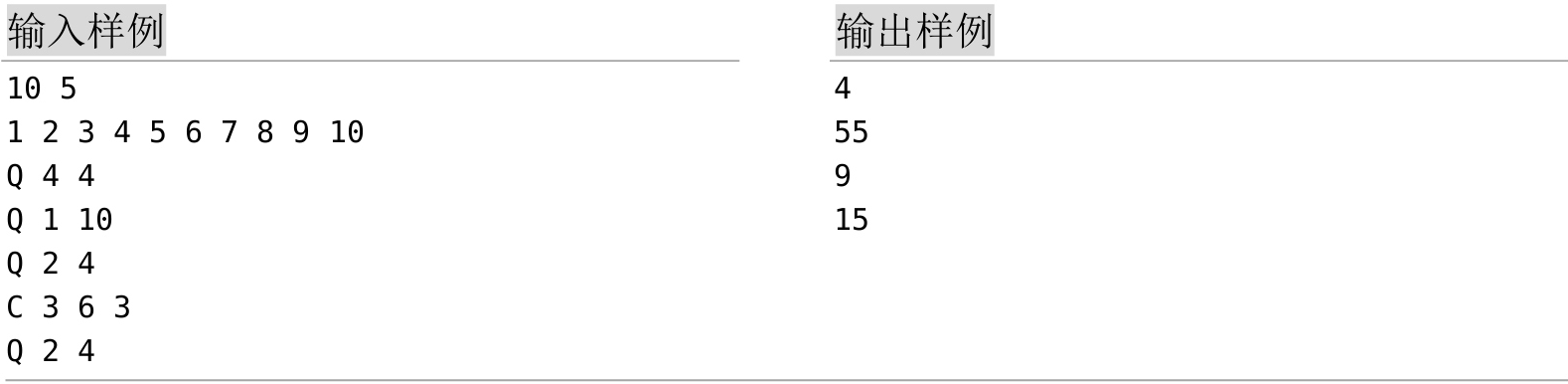
**POJ3648**

**题目描述（POJ3648）：**有N个整数A1, A2, …, AN，需要对其进行两种操作，一种操作是对给定区间中的每个数都添加一个给定的数，另一种操作是查询给定区间中数的总和。

**输入：**第1行包含两个数N和Q（1≤N，Q≤105）；第2行包含N个数，为A1, A2, …, AN的初始值（-109≤Ai≤109）；接下来的Q行，每行都表示一种操作，“C a b c”表示将Aa, Aa+1, …, Ab中的每一个数都加c（-104≤c≤104），“Q a b”表示查询Aa, Aa+1, …, Ab的总和。

**输出：**对每个查询，都单行输出区间和的值。



**提示：**总和可能超过32位整数的范围。

**POJ1019**

**题目描述（POJ1019）：**给出单个正整数i，编写程序以找到位于数字组S1, S2, …, Sk序列中第i位上的数字。每个组Sk都由一系列正整数组成，范围为1～k，一个接一个地写入。序列的前80位数字如下：



**输入：**第1行包含一个整数t（1≤t≤10），表示测试用例的数量。每个测试用例后都跟一行，包含单个整数i（1≤i≤2, 147, 483, 647）。

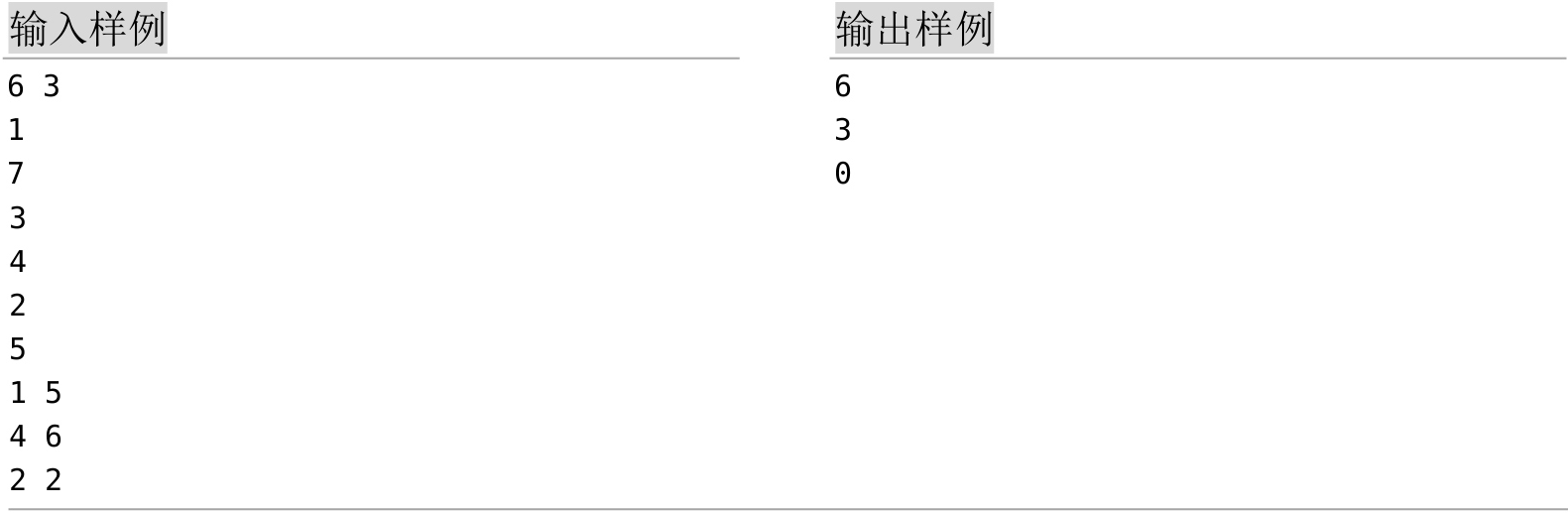
**输出：**对每个测试用例，都单行输出第i位上的数字。

**POJ3264**

**题目描述（POJ3264）：**每天挤奶时，约翰的N头奶牛（1≤N≤50,000）都以相同的顺序排队。他挑选一系列连续的奶牛来玩游戏。为了让所有奶牛都玩得开心，它们的高度差异不应太大。约翰列出了Q组（1≤Q≤200,000）奶牛和它们的高度（1≤height≤1,000,000）。他希望确定每个小组中最高和最矮的奶牛之间的高度差异。

**输入：**第1行包含两个整数N和Q。接下来N行，每行都包含一个整数，表示奶牛的高度。最后Q行，每行都包含两个整数A和B（1≤A≤B≤N），代表从A到B的奶牛范围。

**输出：**输出Q行，每行都包含一个整数，表示该范围内最高和最矮奶牛的高度差。

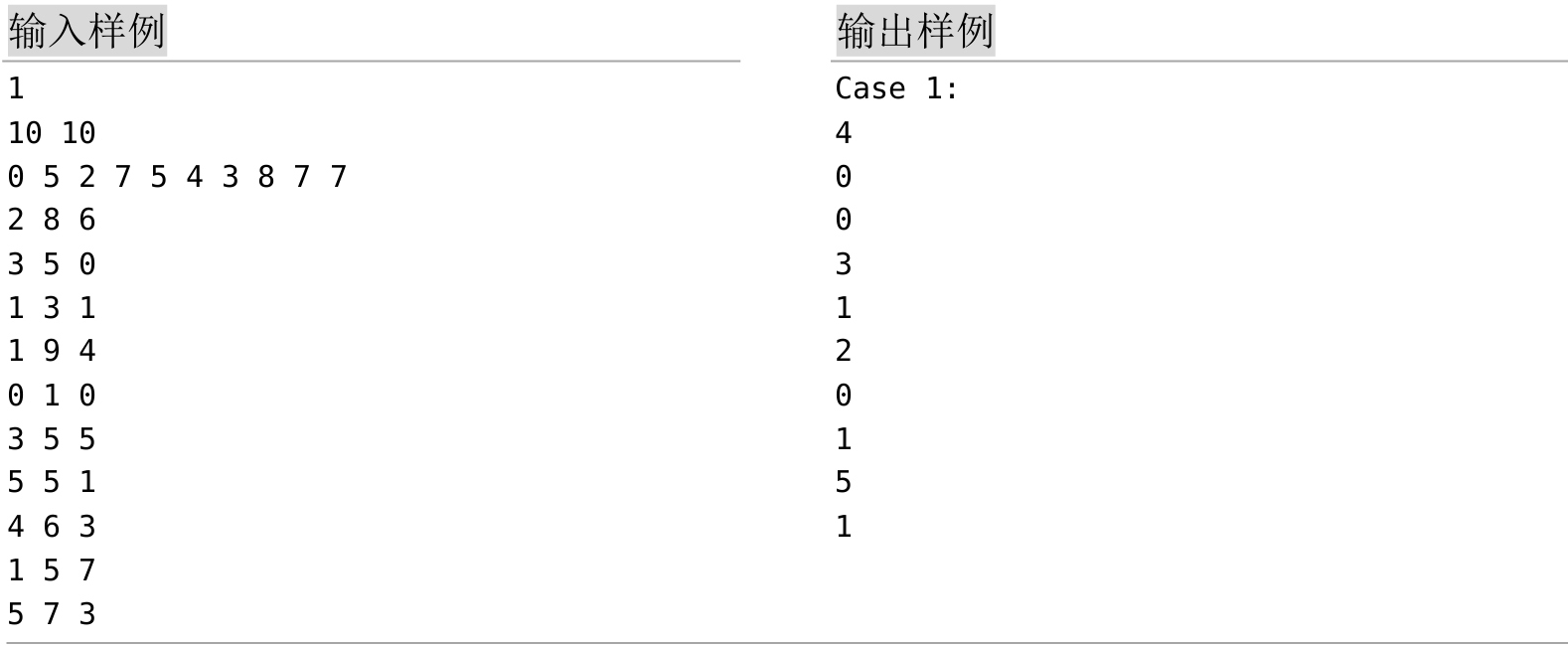


**HDU4417**

**题目描述（HDU4417）：**可怜的公主陷入困境，马里奥需要拯救他的情人。把通往城堡的道路视为一条线（长度为n），在每个整数点i上都有一块高度为hi的砖，马里奥可以跳的最大高度是H，求他在[L, R]区间可以跳过多少砖块。

**输入：**第1行是整数T，表示测试用例的数量。每个测试用例的第1行都包含两个整数n、m（1≤n，m≤105），n是道路的长度，m是查询的数量。下一行包含n个整数，表示每个砖的高度（范围是[0, 109]）。接下来的m行，每行都包含三个整数L、R、H（0≤L≤R<n，0≤H≤109）。

**输出：**对每种情况都输出“Case X：”（X是从1开始的案例编号），后跟m行，每行都包含一个整数。第i个整数是第i个查询中马里奥跳过的砖块数。



**HDU5057**

**题目描述（HDU5057）：**有由N个非负整数组成的序列：a[1], a[2], …, a[N]，对该序列进行M个操作，操作形式：①S X Y，将a[X]的值设置为Y（a[X]=Y）；②Q L R D P，求[L, R]区间第D位是P的元素个数，L和R是序列的索引。注意：第1位是最低有效位。

**输入：**第1行包含一个整数T，表示测试用例的数量。每个测试用例的第1行都包含两个整数N和M。第2行包含N个整数：a[1], a[2], …, a[N]。接下来的M行操作，若类型为S，则在该行中将包含两个整数X、Y；若类型为Q，则将包含4个整数L、R、D、P。其中：1≤T≤50，1≤N, M≤105，0≤a[i]≤231-1，1≤X≤N，0≤Y≤231-1，1≤L≤R≤N，1≤D≤10，0≤P≤9。

**输出：**对每个Q操作，都单行输出答案。

