

## A、数字

内存限制：512 MB 时间限制：1000 ms 文件名：math.cpp/in/out

### 题目描述

桌面上总共有  $a + b$  个数字（每个都是  $1 \sim 9$  的数字之一），Alice 将选取其中  $a$  个数字组成一个  $a$  位十进制数；Bob 将选取其中  $b$  个数字组成一个  $b$  位十进制数。

现在他们想让最后得到的两个数乘积尽可能大，请问这个值最大能是多少。

### 输入格式

第一行一个整数  $T$ ，表示有  $T$  组询问。

对于每组询问，第一行两个整数  $a, b$ 。

第二行 9 个整数，分别表示数字  $1 \sim 9$  的个数。

### 输出格式

对于每一组询问，回答一个整数。

### 样例 1 输入

```
1 | 1
2 | 2 7
3 | 1 1 1 1 1 1 1 1 1
```

### 样例 1 输出

```
1 | 840414816
```

### 样例 1 解释

答案为  $96 \times 8754321$ 。能得到的最大的乘积。

### 数据范围

对于 1, 2, 3 号测试点， $T = 1, A + B \leq 10$

对于 4, 5 号测试点， $T = 1, A + B \leq 20$

对于 6, 7 号测试点， $A = B$

对于所有测试点， $\sum (A + B) \leq 10^4, 1 \leq A, B \leq 10^3$ 。

## B、游戏

内存限制：512 MB 时间限制：1000 ms 文件名：game.cpp/in/out

题目描述

Alice 和 Bob 在玩一个游戏。游戏由 Alice 先手，之后 Alice 和Bob 交替操作。

游戏在一个由  $n$  个格子构成的环上举行。第  $i$  个格子与第  $(i \bmod n) + 1$  个格子相邻。

Alice 第一次操作的时候，可以选择任意一个格子并占领它。

Bob 第一次操作的时候，可以选择任意一个未被占领的格子，并占领它。

接下来 Alice 每一次操作都可以占领一个格子，但需要满足这个格子之前未被任何一人占领且相邻的两个格子中存在一个格子被 Alice 占领。

接下来 Bob 每一次操作都可以占领一个格子，但需要满足这个格子之前未被任何一人占领且相邻的两个格子中存在一个格子被 Bob 占领。

每个格子都有一个分数  $v_i$ ，玩家的分数为占领的所有格子分数和。双方都想最大化自己最终的分数。假设双方永远采用最佳策略，请你计算出最终 Alice 分数会是多少。

输入格式

第一行一个整数  $n$ 。

第二行  $n$  个整数表示  $v_i$ 。

输出格式

一行一个整数表示 Alice 可能的最大得分。

样例 1 输入

```
1 | 4
2 | 7 6 8 4
```

样例 1 输出

```
1 | 13
```

数据范围

对于所有数据： $n \geq 2, 1 \leq v_i \leq 2 \times 10^3$ 。

子任务	条件	分数
1	$n \leq 300$	20
2	$n \leq 5 \times 10^3$	20
3	$n \leq 5 \times 10^5$ ，且第一次操作中，Alice 占领格子 1 是最优的。	20
4	$n \leq 5 \times 10^5$	40

C、海报

内存限制：512 MB 时间限制：1000 ms 文件名：posters.cpp/in/out

## 题目描述

有  $k$  个海报要张贴在大小为  $n \times m$  的告示栏中。允许海报重叠，但不允许旋转或者拆分成小块，也不允许超出告示栏大小范围。请问告示栏上最多能有多大的面积被海报覆盖？

## 输入格式

第一行三个整数  $n, m, k$  表示告示栏的长和宽，还有海报数量。

第二行  $k$  个数，第  $i$  个数表示第  $i$  个海报的长度  $w_i$ 。

第三行  $k$  个数，第  $i$  个数表示第  $i$  个海报的宽度  $h_i$ 。

## 输出格式

一行一个整数表示最大可能被覆盖面积。

## 样例 1 输入

```
1 | 10 10 4
2 | 7 4 1 8
3 | 3 5 3 4
```

## 样例 1 输出

```
1 | 74
```

## 样例 1 解释

假设告示栏左下角坐标为  $(1, 1)$ ，右上角坐标为  $(10, 10)$ ，那么四个海报左上角的坐标分别为：

$(1, 10), (7, 9), (10, 3), (1, 4)$ 。

## 数据范围

对于 10% 的数据，保证  $k \leq 1$ 。

对于 20% 的数据，保证  $k \leq 2$ 。

对于 30% 的数据，保证  $k \leq 3$ 。

对于 50% 的数据，保证  $k \leq 4$ 。

对于 100% 的数据，保证  $k \leq 5, n, m \leq 100, 1 \leq w \leq n, 1 \leq h \leq m$ 。

在  $k = 5$  的数据中，存在部分测试点保证数据随机。这里的随机方式为：选定一组合法的  $n, m$ ，然后在  $[1, n]$  内随机生成  $k$  个整数  $w_i$ ，在  $[1, m]$  内随机生成  $k$  个整数  $h_i$ 。

## D、环

内存限制：32 MB 时间限制：2000 ms 文件名：ring.cpp/in/out

## 题目描述

给定一个大小为  $n$  的序列  $a$  , 其中  $a_i$  属于编号为  $c_i$  的环。

设属于编号为  $x$  的环的数下标分别为  $p_1 < p_2 < \dots < p_k$  , 定义一次编号为  $x$  的环的旋转操作为依次执行  $swap(a_{p_k}, a_{p_{k-1}}), swap(a_{p_{k-1}}, a_{p_{k-2}}), \dots, swap(a_2, a_1)$ 。

现在需要你支持以下两种操作：

- `1 l r` 表示查询  $\sum_{i=l}^r a_i$  的值。
- `2 x` 表示让编号为  $x$  的环执行一次旋转操作。

## 输入格式

第一行三个整数  $n, m, q$  表示序列大小, 环的个数和操作个数。

第二行  $n$  个整数表示  $c_i$  。

第三行  $n$  个整数表示  $a_i$  。

接下来  $q$  行, 每行要么形如 `1 l r` 要么形如 `2 x` 表示一次操作。

## 输出格式

对于每个 `1 l r` 操作输出一个整数表示答案。

## 样例 1 输入

1	5 2 5
2	1 2 1 2 2
3	1 2 3 4 5
4	1 1 5
5	2 1
6	1 3 5
7	2 2
8	1 1 3

## 样例 1 输出

1	15
2	10
3	9

## 数据范围

请注意本题特殊空间限制

保证  $m \leq n \leq 150000, q \leq 150000, c_i \leq m, a_i \leq 10^9, l \leq r \leq n, x \leq n$ 。

### 子任务 1 (10 pts)

$n, q \leq 1000$

### 子任务 2 (20 pts)

操作 1 个数不超过 100 个。

### 子任务 3 (15 pts)

$m = 1$

**子任务 4 (25 pts)**

$$c_i \leq c_{i+1}$$

**子任务 5 (30 pts)**

无特殊限制