

# NOIP2017 模拟赛

## 【题目概况】

中文题目名称	艾奇摘苹果	工作计划	玩串
英文题目与文件名	apple	havefun	fkstring
可执行文件名	apple	havefun	fkstring
输入文件名	apple.in	havefun.in	fkstring.in
输出文件名	apple.out	havefun.out	fkstring.out
单个测试点时限	1s	1s	1s
测试点数目	10	10	10
每个测试点分值	10	10	10
结果比较方式	传统	传统	special judge
运行内存上限	256M	256M	256M

## 【提交源程序文件名】

对于 C 语言	apple.c	havefun.c	fkstring.c
对于 C++语言	apple.cpp	havefun.cpp	fkstring.cpp
对于 pascal 语言	apple.pas	havefun.pas	fkstring.pas

最终测试时打开 -O2 优化开关。  
题目不难，GL&HF。

## 艾奇摘苹果 (apple)

### 【问题描述】

艾奇在遥远的美洲大陆有一片苹果树森林。

听上去非常不错的是，苹果森林每年收成都还不错。

艾奇摘苹果的方式非常有趣，由于苹果树太高，她常直接折断树枝来获得一大堆(整棵子树)的苹果。

艾奇也不想把苹果树破坏的太狠，毕竟苹果树也不是只结一年的苹果。所以她对于一棵树，至多会折断一根树枝。

艾奇不像过去的某鲁姓大师，她并没有那个力气倒拔苹果树，因为一棵树的根是埋在土里的。所以她并不能获得整棵树的苹果。

众所周知，美洲大陆龙卷风贼多。龙卷风既然能摧毁停车场，那么吹翻一片苹果森林应该不是什么难事。所以艾奇养了一些小机器人给艾奇预警龙卷风。

艾奇把这些小机器人放在花上，这样一来小机器人就能吸收树的养分工作，但是这也导致了这朵花不能结果。

如果你折断了一根树枝，小机器人就会因为没有养分而狂躁，它们只吃原本和它们属于**同一棵树上**被摘下来的苹果。如果这些苹果不够吃，它们就会把艾奇吃掉。

树林有点大，艾奇想知道，照她这种收果实的方法，在自己不被吃掉的情况下，至多能收获多少苹果呢？

### 【输入格式】

输入文件名为 *apple.in*。

第一行两个整数  $n$   $m$  分别表示这片森林的节点数和树枝数量(就是边的数量)。我们默认一棵树的根就是这棵树中编号最小的节点。

接下来  $n$  个整数。

如果该整数为正，那么表示这个节点结苹果的数量。

如果该整数为负，那么表示这个节点养了一个小机器人，它被摘下来要吃的果子数量。

对于一棵树的根节点，这个节点的权值可以忽略。

接下来  $m$  行，每行两个整数  $i, j$  表示  $i$  和  $j$  在树上有边相连。

### 【输出格式】

输出文件名为 *apple.out*。

一行一个整数，表示艾奇至多能收获多少苹果。

### 【输入样例 1】

详见 *apple1.in*

### 【输出样例 1】

详见 *apple1.out*

### 【输入样例 2】

详见 *apple2.in*

### 【输出样例 2】

详见 *apple2.out*

### 【输入样例 3】

详见 *apple3.in*

### 【输出样例 3】

详见 *apple3.out*

【数据范围】

测试点编号	$n$	备注
1	$\leq 10$	对于所有的数据 $m \leq n-1$ 设节点权值为 $x_i$ $ x_i  \leq 10^5$
2	$\leq 20$	
3	$\leq 100$	
4	$\leq 500$	
5	$\leq 1000$	
6	$\leq 5000$	
7	$\leq 10000$	
8	$\leq 50000$	
9	$\leq 100000$	
10	$\leq 1000000$	

## 工作计划 (havefun)

### 【问题描述】

工作使艾奇快乐。

勤奋的工作为国家直接贡献了 GDP，艾奇认为只要对国家有利，即使牺牲自己生命也心甘情愿，绝不会因为自己可能受到祸害而躲开。

当艾奇无聊的时候，她就会去工作，然而并不是每次工作都是轻松而愉悦的。当天艾奇又一次来到了学校，等待着她的是一个有  $n$  行  $m$  列的巨大的矩阵和  $q$  个任务。对于每个任务，艾奇被要求交换这个矩阵中的两个子矩阵。

每个任务，艾奇会获得六个正整数  $x1,y1,x2,y2,h,w$ 。 $x1,y1$  代表了第一个矩阵左上角的行列位置（即在第  $x1$  行第  $y1$  列）； $x2,y2$  代表了第二个矩阵左上角的行列位置， $h,w$  代表了这两个矩阵的高和宽（即行数和列数）。

数据保证所有需要交换的矩阵互不相交或相邻。也就是说，没有任何一个元素同时属于这两个矩阵，也不存在某两个元素分别属于两个矩阵且相邻（共边）。

### 【输入格式】

输入文件名为 *havefun.in*。

第一行两个正整数  $n,m,q$  表示矩阵的高和宽（即行数和列数）和任务数。

接下来一行五个参数，将用于生成矩阵，详见备注。

接下来  $q$  行，每行 6 个数， $x1,y1,x2,y2,h,w$ ，分别表示第一个矩阵的左上位置，第二个矩阵的左上位置，这两个矩阵的高和宽（即行数和列数）。

### 【输出格式】

输出文件名为 *havefun.out*。

输出一个数，将用于校验你的矩阵，详见备注。

### 【输入样例 1】

详见 *havefun1.in*

### 【输出样例 1】

详见 *havefun1.out*

### 【输入样例 2】

详见 *havefun2.in*

### 【输出样例 2】

详见 *havefun2.out*

### 【输入样例 3】

详见 *havefun3.in*

### 【输出样例 3】

详见 *havefun3.out*

### 【数据范围】

测试点编号	$n,m$	$q$	备注
1	=100	=100	
2	=100	=1000	
3	=100	=10000	
4	=400	=40000	
5	=1000	=100	

6	=2000	=1000	矩阵所有元素为 0 或 1 y1 Mod 32 == 1 y2 Mod 32 == 1 w Mod 32 == 0
7	=2000	=1000	
8	=2000	=1000	
9	=2000	=1000	
10	=2000	=1000	

#### 【备注】

由于读入规模较大，经过测试，在标准配置中，极限数据使用 `scanf` 读入需要约 1.35s

所以我们决定，将读入矩阵的方式改为生成矩阵。我们将给你五个参数用于生成矩阵，`P, Q, R, S, Mod`。测试证明，这段代码在标准配置中，极限数据的生成只需要 0.12s。你只需要调用 `spawning` 函数，并申请一个至少为 `2005*2005` 的全局数组 `A` 接收即可，下标默认为(1,1)到(n,m)。输入保证 `P, Q, R, S, Mod < 10^9`。你必须保证数组 `A` 中所有元素为 0，矩阵的长宽即 `n, m` 为全局变量。

当然，输出矩阵也是非常缓慢的。请你将你的答案全部赋值给全局数组 `A` 矩阵，并调用 `checker` 函数，你的主程序中不需要输出任何多余信息。

你并不需要对样例三中的答案为负数感到惊讶或担心。给出的 `checker` 函数在极限数据下计算本身就是超出 `long long` 范围的。但凡你的最终矩阵是正确的，该函数就会输出一个正确的结果。

我们已经将这两段生成代码以文件形式下发，文件名为 `Too_young.cpp`。

## 玩串 (fkstring)

### 【问题描述】

艾奇非常喜欢玩串。

艾奇觉得，字符串什么的非常优美。她会有事没事地给一个字符串数它有多少个好的非空子序列。

一个好的子序列，满足它是 SS 的形式。

比如说，abcabc 就是好的，而 2333 就不是好的。

众所周知，一个字符串的子序列有  $2^{\text{长度}}$  的长度次方个，而好的字符串长度一定是偶数。

而一个字符串的权值，就是本质不同的好的子序列的个数。

两个子序列本质不同，当且仅当它们之中任意一个字符所处的位置不同。

(以上大家应该都知道吧)

一天，艾奇数着数累了，她想知道对于一个权值  $x$ ，你能否找到这样一个字符串呢？

当然，本题可能有多个满足条件的解，你只需要输出任何一个即可。

当然我不可能让你随意输出对吧，不然这题还做个球。

任何情况下，字符串的长度不能超过 100，字符集大小不能超过 50。

就这么简单。

### 【输入格式】

输入文件名为 *fkstring.in*。

一行一个整数，表示你输出的串需要包含的权值。

### 【输出格式】

输出文件名为 *fkstring.out*。

第一行一个数  $n$ ，表示你输出字符串的长度。

第二行  $n$  个数，表示你输出的满足要求的串。

### 【输入样例 1】

详见 *fkstring1.in*

### 【输出样例 1】

详见 *fkstring1.out*

### 【输入样例 2】

详见 *fkstring2.in*

### 【输出样例 2】

详见 *fkstring2.out*

### 【数据范围】

测试点编号	$n$	备注
1	$=1$	
2	$=233$	
3	$\leq 50$	
4	$\leq 50$	
5	$\leq 3 \cdot 10^7$	$n=2^t-1$
6	$\leq 3 \cdot 10^7$	$n=2^t-1$
7	$\leq 3 \cdot 10^7$	$n=2^t-1$
8	$\leq 3 \cdot 10^7$	

9	$\leq 3 \times 10^7$	
10	$\leq 3 \times 10^7$	

**【备注】**

下发的样例输出文件，只是供作参考，表示一种合法的字符串。

字符串的输出形式，你需要输出一行不超过 100 个数。每个数的大小不超过 50。每个数表示一个对应的字符。不同的数字代表不同的字符。