

题目名称	拆分	祖先	序列	关系
题目类型	传统型	传统型	传统型	传统型
目录	split	ancestor	sequence	relation
可执行文件名	split	ancestor	sequence	relation
输入文件名	split.in	ancestor.in	sequence.in	relation.in
输出文件名	split.out	ancestor.out	sequence.out	relation.out
每个测试点时限	2.0 秒	2.0 秒	2.0 秒	2.0 秒
内存限制	512MiB	512MiB	512MiB	512MiB
子任务数目	20	20	20	20
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	split.cpp	ancestor.cpp	sequence.cpp	relation.cpp
对于 C 语言	split.c	ancestor.c	sequence.c	relation.c
对于 Pascal 语言	split.pas	ancestor.pas	sequence.pas	relation.pas

编译选项

对于 C++ 语言	-lm
对于 C 语言	-lm
对于 Pascal 语言	

**注意事项与提醒（请选手务必仔细阅读）**

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++ 中函数 main() 的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
3. 提交的程序代码文件的放置位置请参照各省的具体要求。
4. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
5. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）
6. 程序可使用的栈内存空间限制与题目的内存限制一致。
7. 全国统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8700K CPU @3.70GHz，内存 32GB。上述时限以此配置为准。
8. 只提供 Linux 格式附加样例文件。
9. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以其为准。
10. 最终评测时所用的编译命令中不含任何优化开关。
11. 对于 IO 规模大的数据，请采用较快的 IO 方式（IO: Input/Output, 输入和输出）
12. 保证时间限制至少是标程运行时间的 1.5 倍

## 拆分 (split)

### 【题目描述】

给定正整数  $N, P, K$  与长度为  $K$  的正整数序列  $\{a_i\}$ , 不妨令正整数序列  $\{b_i\}$  的和为  $N$ ,  $B$  为序列  $\{b_i\}$  所构成的集合, 令  $A$  为序列  $\{a_i\}$  所构成的集合, 若  $\exists x \in B, x \notin A$ , 则称正整数序列  $\{b_i\}$  为一个有效序列, 求有效序列的数量在  $\text{mod } P$  意义下的值

### 【输入格式】

从文件 `split.in` 中读入数据。

第一行三个整数  $N, P, K$ , 表示数字  $N$ , 模数  $P$ , 数列  $\{a_i\}$  的长度  $K$

第二行包含  $K$  个整数, 表示序列  $\{a_i\}$

### 【输出格式】

输出到文件 `split.out` 中。

仅一行一个整数表示有效序列的数量在  $\text{mod } P$  意义下的值

### 【样例 1 输入】

```
4 10007 2
2 3
```

### 【样例 1 输出】

```
7
```

### 【样例 1 解释】

有效序列分别是:

(1,1,1,1) (1,1,2) (1,2,1) (2,1,1) (1,3) (3,1) (4)

### 【其余样例】

见选手目录下的 `split/split*.in` 与 `split/split*.ans`。

## 【数据范围】

测试点编号	$N \leq$
1 ~ 2	10
3 ~ 6	$10^3$
7 ~ 10	$10^7$
11 ~ 20	$10^{18}$

对于所有测试点： $N \leq 10^{18}, P \leq 10^9, K \leq 100, a_i \leq 100$

## 祖先 (ancestor)

### 【题目描述】

给定一棵无根树  $T$  和  $q$  组询问, 每次询问三个正整数  $rt, x, y$  表示询问无根树  $T$  的根为  $rt$  时,  $x$  和  $y$  的最近公共祖先的编号

### 【输入格式】

从文件 *ancestor.in* 中读入数据。

第一行两个整数  $n, q$ , 表示无根树  $T$  有  $n$  个节点,  $q$  次询问

第  $2 \sim n$  行, 每行两个整数  $u, v$  表示  $u, v$  之间有一条无向边 ( $1 \leq u, v \leq n$ )

第  $n+1 \sim n+q$  行, 每行三个数字  $rt, x, y$ , 意义见题目描述。

### 【输出格式】

输出到文件 *ancestor.out* 中。

输出包含  $q$  行, 对于每组询问, 给出一行一个整数表示对应的答案

### 【样例 1 输入】

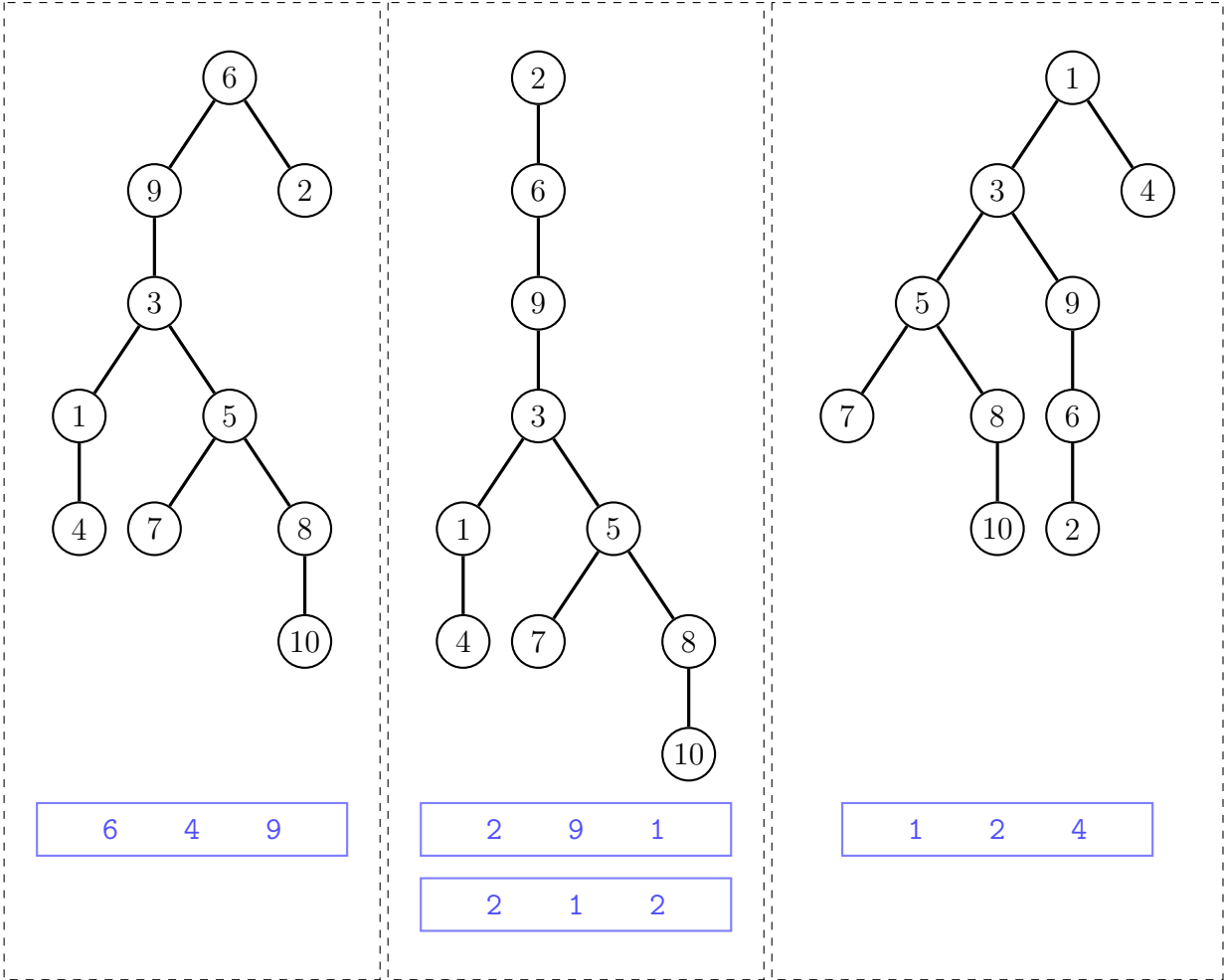
```
10 4
3 9
1 3
6 9
3 5
1 4
10 8
8 5
7 5
2 6
6 4 9
2 9 1
2 1 2
1 2 4
```

### 【样例 1 输出】

```
9
```

9  
2  
1

【样例 1 解释】



【其余样例】

见选手目录下的 ancestor/ancestor\*.in 与 ancestor/ancestor\*.ans。

【数据范围】

测试点编号	$n \leq$	$q \leq$
1 ~ 4	$2 \times 10^3$	$2 \times 10^3$
5 ~ 12	$5 \times 10^5$	$5 \times 10^5$
13 ~ 20	$3 \times 10^6$	$5 \times 10^5$

对于所有测试点:  $n \leq 3 \times 10^6, q \leq 5 \times 10^5$

## 序列 (sequence)

### 【题目描述】

不妨定义**单峰序列**为一个不下降序列和一个不上升序列拼在一起的序列  
从  $[1, n]$  中选出  $m$  个数生成一个序列, 求生成序列恰好为**单峰序列**的方案数  
答案对  $10^9 + 7$  取模。

### 【输入格式】

本题输入包含多组测试数据  
从文件 `sequence.in` 中读入数据。  
第一行一个整数  $T$  表示有  $T$  组数据  
第  $2 \sim T + 1$  行, 每行两个正整数  $n, m$ , 意义见题目描述

### 【输出格式】

输出到文件 `sequence.out` 中。  
输出共包含  $T$  行, 对于每组数据输出一行一个整数表示答案

### 【样例 1 输入】

```
2
2 3
3 3
```

### 【样例 1 输出】

```
7
22
```

### 【样例 1 解释】

对于第一组数据来说, 符合条件的序列有  
 $(1, 1, 1), (1, 1, 2), (1, 2, 1), (1, 2, 2), (2, 1, 1), (2, 2, 1), (2, 2, 2)$ , 不符合条件的序列有  
 $(2, 1, 2)$ , 故答案为 7。

### 【其余样例】

见选手目录下的 `sequence/sequence*.in` 与 `sequence/sequence*.ans`。

## 【数据范围】

测试点编号	$n \leq$	$m \leq$
1	6	6
2	$5 \times 10^5$	2
3 ~ 4	$5 \times 10^5$	5
5 ~ 8	200	200
9 ~ 10	$10^3$	$10^3$
11 ~ 16	$10^5$	$10^5$
17 ~ 20	$5 \times 10^6$	$5 \times 10^6$

对于所有测试点：  $1 \leq T \leq 10$ ,  $1 \leq n \leq 5 \times 10^6$ ,  $1 \leq m \leq 5 \times 10^6$



## 关系 (relation)

### 【题目描述】

有  $n$  个点，每个点有点权  $a_i$ ,  $i$  与  $j$  节点之间有边当且仅当  $a_i \text{ AND } a_j = 0$   
不妨定义点集  $V$  是由这  $n$  个点构成的集合，存在一个初始为空集的集合  $T$ ,  
对于节点  $u, v \in V$  可以进行如下操作

1. 将节点  $u \notin T$  加入集合  $T$ , 贡献为 0
2. 若节点  $u$  和节点  $v$  之间有连边且  $u \notin T, v \in T$ , 将  $u$  加入集合  $T$ , 贡献为  $a_v$

求当  $\forall u \in V, u \in T$  时的最大贡献值

其中  $\text{AND}$  表示按位与操作

### 【输入格式】

从文件 `relation.in` 中读入数据。

第一行一个正整数  $n$

第二行  $n$  个正整数  $a_i$ , 意义见题目描述

### 【输出格式】

输出到文件 `relation.out` 中。

仅一行一个正整数表示最大贡献值

### 【样例 1 输入】

```
3
1 2 3
```

### 【样例 1 输出】

```
2
```

### 【样例 1 解释】

将 2 加入集合  $T$ , 贡献为 0

因为  $1 \text{ AND } 2 = 0, 2 \in T$ , 将 1 加入集合  $T$ , 贡献为 2

因为没有节点和 3 之间有边, 只能直接加入集合  $T$ , 贡献为 0

所以最大贡献值为  $0 + 2 + 0 = 2$

**【其余样例】**

见选手目录下的 relation/relation\*.in 与 relation/relation\*.ans。

**【数据范围】**

测试点编号	$n \leq$
1 ~ 4	8
5 ~ 8	18
9 ~ 10	$10^3$
11 ~ 14	$2 \times 10^5$
15 ~ 20	$5 \times 10^5$

对于所有测试点： $1 \leq n \leq 5 \times 10^5$ ,  $1 \leq a_i \leq 2^{18} - 1$