

# 全国信息学奥林匹克联赛（NOIP2020）模拟赛

考试时间：4 小时

题目名称	打出春天	真夏は誰のモノ	秋天的第一杯奶茶	冬天的内存
题目类型	传统型	传统型	传统型	传统型
目录	landlord	aqours	milktea	memoryofwinter
可执行文件名	landlord	aqours	milktea	memoryofwinter
输入文件名	landlord.in	aqours.in	milktea.in	memoryofwinter.in
输出文件名	landlord.out	aqours.out	milktea.out	memoryofwinter.out
每个测试点时限	1 秒	1 秒	5 秒	3 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
子任务数目	10	10	20	10
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	landlord.cpp	aqours.cpp	milktea.cpp	memoryofwinter.cpp
对于 C 语言	landlord.c	aqours.c	milktea.c	memoryofwinter.c
对于 Pascal 语言	landlord.pas	aqours.pas	milktea.pas	memoryofwinter.pas

编译选项（不包含任何优化开关）

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

## 注意事项

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. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
4. 评测时采用的机器配置为：Intel(R) Core(TM) i5 – 9500 CPU @ 3.00GHz，内存 4GB。上述时限以此配置为准。
5. 只提供 Linux 格式附加样例文件。
6. 选手应在选手文件夹内建立各个题目对应的子文件夹，各题的源程序应放在相应的子文件夹里。
7. 特别提醒：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。

# 打出春天（landlord）

## 【题目背景】

小愛是一个喜欢斗地主的女孩子。

## 【题目描述】

斗地主是一种使用黑桃、红心、梅花、方片的 A 到 K 加上大小王的共 54 张牌来进行的扑克牌游戏。在斗地主中，牌的大小关系根据牌的数码表示如下： $3 < 4 < 5 < 6 < 7 < 8 < 9 < 10 < J < Q < K < A < 2 < \text{小王} < \text{大王}$ ，而花色并不对牌的大小产生影响。每一局游戏中，一副手牌由若干张牌组成。游戏者每次可以根据规定的牌型进行出牌，首先打光自己的手牌一方取得游戏的胜利。

经过数年的努力，小愛已经通过了【NOIP2015】斗地主，并且成为了斗地主专家。她现在已经不仅仅满足于赢牌。

这天，小愛叫上了小 \$ 和小 ω 陪她玩斗地主。因为小愛很自信，所以她总是地主。地主有 20 张牌，而农民分别只有 17 张。

小愛看了一眼自己的牌，就已经知道能否获胜了。小愛不想简单的胜利，而是希望打出“春天”。

对于地主来说，打出“春天”即在两个农民都没有出一张牌的情况下，将自己的手牌全部打光。

假设小 \$ 和小 ω 能出牌时一定会出牌，小愛想知道，自己的这副牌是否一定能够打出“春天”。她告诉了她的手牌，请你帮她解决这个问题。

小愛玩的斗地主中需要考虑的牌型如下：

牌型	牌型说明	牌型举例
王炸	即双王（双鬼牌）	大王 小王
炸弹	四张同点牌	♠A ♥A ♣A ♦A
单牌	单张牌	♠3
对子	两张点数相同的牌	♠2 ♥2
三条	三张点数相同的牌	♠3 ♥3 ♣3
三带一	三张点数相同的牌 + 一张单牌	♠3 ♥3 ♣3 ♠4
三带二	三张点数相同的牌 + 一对牌	♠3 ♥3 ♣3 ♠4 ♥4
单顺子	五张或更多点数连续的单牌（不包括 2 点和双王）	♠7 ♣8 ♠9 ♣10 ♣J
双顺子	三对或更多点数连续的对子（不包括 2 点和双王）	♣3 ♥3 ♠4 ♥4 ♠5 ♥5
飞机	两个或更多点数连续的三条（不包括 2 点和双王）	♠3 ♥3 ♣3 ♠4 ♥4 ♣4 ♠5 ♥5 ♦5 ♥5
四带二	四张点数相同的牌 + 任意两张不同的单牌（或任意两对不同的牌）	♠5 ♥5 ♣5 ♦5 ♠4 ♥2 ♠5 ♥5 ♣5 ♦5 ♣3 ♦3 ♣8 ♦8
飞机带翅膀	在飞机的基础上加上和三条组数相同的单张（或和三条组数相同的对子）	♣3 ♥3 ♠3 ♥4 ♠4 ♣4 ♥2 ♠9

## 注意事项

1. 四带二中带的单牌/对子不能相同。飞机带翅膀中,每对翅膀都不能相同,翅膀与三条的点数也不能相同。例如:  $\spadesuit 5 \heartsuit 5 \clubsuit 5 \diamondsuit 5 \spadesuit 4 \heartsuit 4, \clubsuit 3 \heartsuit 3 \spadesuit 3 \heartsuit 4 \spadesuit 4 \clubsuit 4 \heartsuit 2 \spadesuit 2$  和  $\clubsuit 3 \heartsuit 3 \spadesuit 3 \heartsuit 4 \spadesuit 4 \clubsuit 4 \heartsuit 2 \diamondsuit 4$  都不是合法的牌型。
2. 大王和小王可以作为单牌和其他牌型组合,但不能作为对子。例如:  $\spadesuit 3 \heartsuit 3 \clubsuit 3$  大王 和  $\spadesuit A \heartsuit A \clubsuit A \diamondsuit A$  大王 小王 都是合法牌型。

在斗地主中,除了王炸和炸弹,只有和上家出的牌牌型一样(牌数必须一样)且更大的牌才能出。同种牌型之间的大小比较规则如下:

1. 单张、对子、三条、炸弹,任意一张点数更大的牌较大。如  $\spadesuit 2 \heartsuit 2 > \spadesuit A \heartsuit A$ , 小王  $> \heartsuit Q$ 。
2. 三带一、三带二、四带二,只比较三张(四张)相同牌的大小,点数更大的牌较大。如  $\spadesuit 2 \heartsuit 2 \diamondsuit 2 \diamondsuit 4 > \spadesuit 3 \heartsuit 3 \diamondsuit 3 \heartsuit 10$ 。
3. 单顺子、双顺子、飞机,比较两组牌中最小(或最大)的那张牌的点数大小,点数更大的牌较大。如  $\spadesuit 10 \heartsuit J \diamondsuit Q \diamondsuit K \clubsuit A > \clubsuit 3 \spadesuit 4 \spadesuit 5 \heartsuit 6 \diamondsuit 7$ 。
4. 飞机带翅膀,只比较飞机的大小。

点数的比较规则见题目描述的开头部分。

王炸能够接其他所有牌型。炸弹能够接除了王炸和比它大的炸弹外的其他所有牌型。也就是说  $\spadesuit 3 \heartsuit 3 \clubsuit 3 \diamondsuit 3$  能接  $\spadesuit 2 \heartsuit 2 \clubsuit 2 \diamondsuit 2$  大王 小王。

保证 54 张牌的每张都出现在游戏中,也就是说,除了小爱的 20 张牌以外,剩下的 34 张牌一定在小 \$ 或小  $\omega$  手中。

## 【输入格式】

从文件 *landlord.in* 中读入数据。

第一行一个正整数  $T$ , 表示数据组数。

接下来  $T$  行, 每行 20 个长度为 2 的字符串, 表示小爱的手牌。

其中, 大王和小王用 \$E 和 \$1 表示。其他牌的第一个字符为 S, H, C, D 中的一种, 分别表示黑桃 (Spade), 红桃 (Heart), 梅花 (Club), 方片 (Diamond)。第二个字符为 2, 3, 4, 5, 6, 7, 8, 9, T, J, Q, K, A 中的一种, 表示点数。其中 T 表示 10。

## 【输出格式】

输出到文件 *landlord.out* 中。

共  $T$  行, 第  $i$  行为一个字符串, 若第  $i$  组数据中小爱存在一定能打出“春天”的出牌方式, 则输出 “Yes”, 否则输出 “No” (均不包含引号)。

### 【样例 1 输入】

```
2
S3 S4 S5 S6 S7 S8 S9 ST SJ SQ SK SA S2 H2 C2 D2 $E H4 C4 D4
S3 S4 S5 S6 S7 S8 S9 ST SJ SQ SK SA S2 H2 C2 D2 HA CA DA DT
```

### 【样例 1 输出】

```
Yes
No
```

### 【样例 1 解释】

对于第一组数据，其中一种可行的出牌顺序为 ♠3 ♠4 ♠5 ♠6 ♠7 ♠8 ♠9 ♠10 ♠J ♠Q ♠K ♠A ,  
♠2 ♥2 ♣2 ♦2, ♥4 ♣4 ♦4 大王。

对于第二组数据，不论小爱第一轮出什么牌，小 \$ 或小 ω 都有可能跟牌。

### 【样例 2 输入】

```
2
S6 H6 C6 D6 S7 H7 C7 D7 S8 H8 C8 D8 S9 H9 C9 D9 SA HK C3 $1
S6 H6 C6 S7 H7 C7 S8 H8 C8 S9 H9 C9 S5 H5 D5 C3 HA SK $1 $E
```

### 【样例 2 输出】

```
No
Yes
```

### 【数据范围】

对于前 10% 的数据，保证小爱仅有两种点数的牌超过一张。  
对于前 20% 的数据，保证小爱有大王和小王。  
对于前 40% 的数据，保证小爱拥有双王外的所有点数的牌各至少一张。  
对于另外 10% 的数据，保证小爱的牌随机生成。  
对于另外 20% 的数据，保证小爱每种点数的牌不超过两张。  
对于 100% 的数据， $1 \leq T \leq 5$ ，保证小爱的牌合法。

## 真夏は誰のモノ（aqours）

### 【题目背景】

赤い太陽のドレスで踊る  
私のことを見つめているの？  
目をそらしたい でもそらせない  
Ah 情熱で灼かれない

小愛是 Aqours 的粉丝。

### 【题目描述】

小愛喜欢听 Aqours 的歌，但是她并不是很懂日语。

为了学习日语的发音，小愛想打印一些资料，用长度为  $n$  的字符串  $s$  表示。但是小愛把电脑卖了去买 Aqours 的 Live 的门票了，所以她请小 Z 帮她打印。

但是小 Z 的电脑的打印功能有很大的问题。打印程序每次会随机选一个  $i$  ( $1 \leq i \leq n$ )，然后会随机将  $i$  左边或右边连续若干个字符（可能是 0 个）都变成  $s_i$ 。由于程序极不稳定，这个操作会被执行任意次数（可能为 0 次）。

还有一个问题是，如果打印的字符串存在一个长度超过  $k$  的连续子段，子段内的字符都一样，则打印机无法打印出字符串。

小 Z 没有办法解决这个问题。幸好小愛懂一些电脑维修的知识，所以她开始修小 Z 的电脑。

为了知道问题的所在，小愛进行了一次打印，她需要知道，对于给定的  $s$  和  $k$ ，打印机能够打印出来的不同字符串的个数是多少。

由于答案可能很大，你只需要告诉小愛答案对  $10^9 + 7$ （是一个质数）取模后的结果即可。

### 【输入格式】

从文件 `aqours.in` 中读入数据。

第一行两个整数  $n, k$ 。

第二行一个长度为  $n$  的仅包含小写英文字母和下划线的字符串  $s$ 。

### 【输出格式】

输出到文件 `aqours.out` 中。

一行一个非负整数表示答案对  $10^9 + 7$  取模后的结果。

### 【样例 1 输入】

```
4 2
abaa
```

**【样例 1 输出】**

5

**【样例 1 解释】**

打印机可能打印出的字符串有：abaa, abba, aabb, aaba, bbaa。

**【样例 2 输入】**

```
6 4
iakioi
```

**【样例 2 输出】**

273

**【样例 3 输入】**

```
12 7
doll_machine
```

**【样例 3 输出】**

635635

**【样例 4】**

见选手目录下的 *aqours/aqours4.in* 与 *aqours/aqours4.ans*。

**【样例 4 解释】**

该样例中的字符串为《真夏は誰のモノ》的罗马音。

**【数据范围】**

对于 10% 的数据， $k \leq n \leq 5$ 。

对于 20% 的数据， $k \leq n \leq 10$ 。

对于 30% 的数据， $k \leq n \leq 50$ 。

对于 50% 的数据， $k \leq n \leq 300$ 。

对于另外 20% 的数据，保证字符串中仅出现 a, b 两种字符。

对于 90% 的数据， $k \leq n \leq 3000$ 。

对于 100% 的数据， $1 \leq k \leq n \leq 6000$ 。

## 【后记】

赤い太陽のドレスで踊る  
私のこと見つめる瞳  
目をそらしたい でもそらせない  
真夏は誰のモノ？  
あなたと私のモノにしたい  
だってね こころがね止まらない季節  
初めて胸のトビラが開いてしまいそうよ  
You knock knock my heart!!  
Ah 情熱的に抱きしめて！



## 秋天的第一杯奶茶（milktea）

### 【题目背景】

小愛是一个喜欢喝奶茶的女孩子。

### 【题目描述】

小愛的学校旁边有一家奶茶店，小愛经常和 xpp 一起中午出来买奶茶。

小愛数了一下，奶茶店里一共有  $n$  种奶茶，编号为  $1, 2, \dots, n$ 。每种奶茶都有一个美味度  $a_1, a_2, \dots, a_n$ 。

这天，小愛买完奶茶之后，感觉很无聊，于是开始观察老板工作。

老板制作奶茶的速度很快，可以忽略不计。所以老板只需要处理客人的订单，以及修改奶茶的美味度来平衡每种奶茶的销量。

小愛一共观察了  $m$  个时刻，每个时刻都会发生下列两种事件中的一种。

1. 老板会选定  $l, r, v$ ，然后对于编号在  $[l, r]$  内的奶茶，老板会修改它的配料，使得这种奶茶的美味度增加  $v$ （可能为负数）。
2. 一位客人选定  $l, r$ ，并买了所有编号在  $[l, r]$  内的奶茶各一杯。这位客人的快乐度为他买的所有奶茶的美味度之和（可能为负数）。

喜欢思考的小愛想到了一个问题：已知每种奶茶的初始美味度，并选定  $x, y$ ，如果只按顺序发生  $[x, y]$  时刻内的事件，那么这段时间里来的客人的快乐度之和是多少呢？

小愛不满足于求解单个问题，所以她选定了  $k$  组  $x, y$ 。

沉浸在思考中的小愛突然意识到快要上课了。于是她匆匆忙忙地进校，并把这个问题交给了你。

### 【输入格式】

从文件 `milktea.in` 中读入数据。

第一行三个整数  $n, m, k$ ，分别表示奶茶的种数，事件的个数以及小愛的询问个数。

第二行  $n$  个整数  $a_1, a_2, \dots, a_n$ ，表示每种奶茶的初始美味度。

接下来  $m$  行，每行第一个整数为  $op$ 。若  $op = 1$ ，则该行接下来紧跟三个整数  $l, r, v$ ，表示老板修改了奶茶美味度。若  $op = 2$ ，则该行接下来紧跟两个整数  $l, r$ ，表示客人来购买奶茶。均满足  $1 \leq l \leq r \leq n$ 。

接下来  $k$  行，每行两个整数  $x, y$ ，表示小愛的询问。

### 【输出格式】

输出到文件 `milktea.out` 中。

共  $k$  行，第  $i$  行输出一个整数表示第  $i$  个询问的答案。

**【样例 1 输入】**

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

**【样例 1 输出】**

```
6
29
0
20
72
```

**【样例 2】**

见选手目录下的 *milktea/milktea2.in* 与 *milktea/milktea2.ans*。

**【样例 3】**

见选手目录下的 *milktea/milktea3.in* 与 *milktea/milktea3.ans*。  
该样例满足测试点 11 的性质。

【数据范围】

测试点编号	$n$	$m$	$k$	特殊性质
1	$\leq 300$	$\leq 300$	$\leq 300$	无
2				
3				
4	$\leq 10^5$	$\leq 10^3$	$\leq 10^3$	
5				
6				
7		$\leq 300$	$\leq 10^5$	
8		$\leq 10^5$		
9				
10				
11				
12		$op = 2$ 时 $l = r$		
13				
14				
15	$\leq 5 \times 10^4$	$\leq 5 \times 10^4$	$\leq 5 \times 10^4$	无
16				
17				
18	$\leq 10^5$	$\leq 10^5$	$\leq 10^5$	
19				
20	$\leq 2 \times 10^5$	$\leq 2 \times 10^5$	$\leq 2 \times 10^5$	

对于 100% 的数据， $1 \leq n, m, k \leq 2 \times 10^5$ ， $1 \leq x \leq y \leq m$ ， $-10^3 \leq a_i, v \leq 10^3$ 。

## 冬天的内存（memoryofwinter）

### 【题目背景】

小愛是一个可爱的女孩子。

### 【题目描述】

小愛有一个朋友叫得便（Debian），他是一位计算机领域的专家，发明了震惊世界的超级计算机——得便机（Deipian's Machine）。得便机的一个最为卓越的成就是，它拥有一套高级的内存分配机制。这种被称作“冬天的内存”（Memory of winter）的内存分配机制运用了前沿的得域理论，能够随季节变换而改变内存大小。得便的这一研究成果也是得便计算机领域的开端。

由于制造得便机的材料极其稀有，所以全世界只有一台得便机，得便把它放在家里用来进行研究。

小愛想玩得便机，于是她问得便去借。到了得便家，小愛发现得便在研究一种具有重要意义的数，被命名为 deb 数。

$k$  阶 deb 数是一类不包含前导零的正整数，它的十进制表示中的逆序对个数恰好为  $k$ 。例如，12243 是 1 阶 deb 数，114514 是 3 阶 deb 数，123456 是 0 阶 deb 数。

为了考验一下小愛，得便给出了三个整数  $l, r, k$ ，他让小愛统计出，在  $l$  与  $r$  之间（包括  $l$  和  $r$ ）的  $0, 1, 2, \dots, k$  阶 deb 数的个数之和。

小愛不会做这个题，所以她请你帮她解决这个问题。由于答案可能很大，你只需要告诉小愛答案对 998,244,353（是一个质数）取模后的结果。

### 【输入格式】

从文件 *memoryofwinter.in* 中读入数据。

一行，三个整数  $l, r, k$ ，意义见题目描述。

### 【输出格式】

输出到文件 *memoryofwinter.out* 中。

一行一个整数表示答案。

### 【样例 1 输入】

30 40 0

### 【样例 1 输出】

7

**【样例 1 解释】**

范围内的 0 阶 deb 数有：33, 34, 35, 36, 37, 38, 39。

**【样例 2 输入】**

10000 23333 2

**【样例 2 输出】**

4962

**【样例 3 输入】**

1 10000000 13

**【样例 3 输出】**

8845540

**【数据范围】**

对于 20% 的数据， $r - l \leq 10^5$ 。

对于 40% 的数据， $r - l \leq 10^8$ 。

对于另外 20% 的数据， $k \leq 3$ 。

对于 100% 的数据， $1 \leq l \leq r \leq 10^9$ ， $0 \leq k \leq 100$ 。

**【后记】**

当你正在做这道题的时候，得便在得便机上写了一份暴力代码，并能瞬间求出  $10^{10^{10000}}$  内的任意阶 deb 数的个数。