

NOIP2017模拟赛

一. 题目概况

中文题目名称	Excellent数	数字分割	远离班主任
英文题目与子目录名	excellent	num	tree
可执行文件名	excellent	num	tree
输入文件名	excellent.in	num.in	tree.in
输出文件名	excellent.out	num.out	tree.out
每个测试点时限	1 秒	1 秒	1 秒
测试点数目	10	20	10
每个测试点分值	10	5	10
附加样例文件	无	无	无
结果比较方式	全文比较（过滤行末空格及文末回车）		
题目类型	传统	传统	传统
运行内存上限	128M	128M	128M

二. 提交源程序文件名

对于 C++语言	excellent.cpp	num.cpp	tree.cpp
对于 C 语言	excellent.c	num.c	tree.c
对于 pascal 语言	excellent.pas	num.pas	tree.pas

Excellent 数(excellent)

【问题描述】

如果一个数，它的每位上是 a 或者 b，我们就称它是 good 数，如果一个 good 数的每位数字之和也是 good 数，我们称它为 Excellent 数。如果 a=1 并且 b=3，则 11 和 13 是 good 数，12 不是 good 数，111 是 Excellent 数而 11 不是。你要做的就是找到长度为 n 的数中，有多少个数是 Excellent 数。

【输入文件】

输入一行，三个整数 a，b，n。

【输出文件】

输出长度为 n 的十进制数中，有多少是 Excellent 数，答案 mod 1000000007。

【样例】

excellent.in	excellent.out	excellent.in	excellent.out
1 3 3	1	2 3 10	165

【数据规模】

对于 20%的数据， $1 \leq n \leq 6$

对于 40%的数据， $1 \leq n \leq 15$

对于 70%的数据， $1 \leq n \leq 10^3$

对于 100%的数据， $1 \leq a, b \leq 9, a \neq b, 1 \leq n \leq 10^6$

数字分割(num)

【问题描述】

一个长度为 n 的数字字符串，现在需要你进行拆分，使得拆分后得到的数是单调递增的，且每个数都不能有前导 0

求方案数。

【输入格式】

第一行一个整数 n ，表示数字字符串的长度。

第二行 n 个整数，表示数字字符串，第一个字符一定不是 0.

【输出格式】

输出所有的拆分方案，结果对 1000000007 进行取余。

【输入输出样例 1 】

santa. in	santa. out	样例解释：
6 123434	8	"123434" = "123434" (maybe the given sequence is just one big number) "123434" = "1" + "23434" "123434" = "12" + "3434" "123434" = "123" + "434" "123434" = "1" + "23" + "434" "123434" = "1" + "2" + "3434" "123434" = "1" + "2" + "3" + "434"

		"123434" = "1" + "2" + "3" + "4" + "34"
--	--	---

【输入输出样例 2 】

santa. in	santa. out	样例解释：
8	4	"20152016" = "20152016"
20152016		"20152016" = "20" + "152016"
		"20152016" = "201" + "52016"
		"20152016" = "2015" + "2016"

【数据范围】

对于 30%的数据，n 的范围[1,15];

对于 50%的数据，n 的范围[1,50];

对于 60%的数据，n 的范围[1,100];

对于 80%的数据，n 的范围[1,300];

对于 90%的数据，n 的范围[1,1000];

对于 100%的数据，n 的范围[1,3000];

远离班主任(tree)

【问题描述】

n 个同学在一棵树上。现在班主任来了。树上有两个点是机房。问所有同学躲进机房最少要花多少时间。 假设树上每条边都要花费 1 的时间。

【输入格式】

第一行一个整数，表示有 n 个同学

接下来 n 行，每行两个整数，表示一条树上的边。

接下来一个整数，表示 m 个询问

接下来 m 行每行两个整数表示机房的位置。

【输出格式】

对于每个询问，输出一个正整数 s 。

【样例输入】

tree. in	tree. out	tree. in	tree. out
3	1	4	2
2 3	1	1 4	1
3 1	1	1 2	2
3		2 3	
2 1		3	
2 3		1 4	
3 1		1 3	
		2 3	

数据说明

测试点	n	m	约定
1	≤ 2000	≤ 2000	无
2	≤ 2000	≤ 5000	
3	≤ 2000	≤ 10000	
4	≤ 100000	≤ 100000	保证树是一条链
5	≤ 100000	≤ 100000	询问时有一个点恒为 1 且树是随机构造的
6	≤ 100000	≤ 100000	询问时有一个点恒为 1
7	≤ 100000	≤ 100000	询问时有一个点恒为 1
8	≤ 100000	≤ 100000	无
9	≤ 100000	≤ 100000	
10	≤ 100000	≤ 100000	

本地测试，数据比较大所以请用栈外挂（放在 **main** 函数下面）：

```
int size = 32 << 20; // 32MB
```

```
char *p = (char*)malloc(size) + size;
```

```
__asm__("movl %0, %%esp\n" :: "r"(p));
```