

第 21 届上海大学程序设计联赛 春季赛



比赛题目列表

- A Antiamuny wants to learn binary search
- B Bespread with chequers
- C Converse the string
- D Deliver the string
- E Eliminate suspicion
- F Fill out the multiple choice questions
- G Golden jade matrix checker
- H How to know the function
- I I like UNO !
- J Juxtaposed brackets

上海大学

2023 年 4 月 16 日

由上海大学第零训练委员会命题

Problem A. Antiamuny wants to learn binary search

Input file: standard input
Output file: standard output

Why did Antiamuny want to learn binary search?

Because he was tired of looking for his keys using linear search!

Antiamuny 正在学习二分！所以他通过不同的语言实现了一个二分的子程序：

C/C++:

```
int f(int l,int r,int x) { // l <= x <= r
    int cnt = 0;
    while(l <= r) {
        cnt++;
        int mid = (l + r) / 2;
        if (mid == x) break;
        if (mid < x) l = mid + 1;
        else r = mid - 1;
    }
    return cnt;
}
```

Java:

```
public static int f(int l,int r,int x) { // l <= x <= r
    int cnt = 0;
    while(l <= r) {
        cnt++;
        int mid = (l + r) / 2;
        if (mid == x) break;
        if (mid < x) l = mid + 1;
        else r = mid - 1;
    }
    return cnt;
}
```

Python:

```
def f(l, r, x): # l <= x <= r
    cnt = 0
    while l <= r:
        cnt += 1
        mid = (l + r) // 2
        if mid == x: break
        elif mid < x: l = mid + 1
        else: r = mid - 1
    return cnt
```

可以发现，上述程序实现了在 $[L, R]$ 中寻找 x 并且返回了 `while` 循环的运行次数 `cnt`。

Antiamuny 非常好奇：对于给定的 L, R, x ，函数 $f(L, R, x)$ 返回的 `cnt` 是多少，请你写个程序帮帮他。

Input

第一行一个整数 T ($1 \leq T \leq 10^2$) 表示数据组数。

接下来 T 行，每行有三个整数 L, R, x ($1 \leq L \leq x \leq R \leq 10^3$)，分别对应函数 $f(L, R, x)$ 的三个参数。

Output

输出 T 行，每行一个整数，表示对应的 $f(L, R, x)$ 返回的值。

Examples

standard input	standard output
5	2
3 7 6	2
6 12 7	3
2 10 2	3
6 14 13	1
5 8 6	
1	1
1 1 1	

Problem B. Bespread with chequers

Input file: standard input
Output file: standard output

The floor of the grand hall was bespread with chequers, creating a mesmerizing pattern that drew the eye towards the regal throne at the end of the room.

JYH 有一个 2 行, 但列数不定的棋盘, 他邀请你在他的棋盘上堆满棋子。

你带了一个神奇口袋, 里面装着无穷多的 1×2 大小和 2×2 大小的棋子, 其中 1×2 大小的棋子可以横着占据 1 行 2 列的格子, 也可以竖着占据 2 行 1 列的格子。

JYH 想跟你玩 T 次游戏, 每次游戏他会告诉你当前棋盘的列数 n , 请你对于每次游戏输出铺满当前棋盘的方案总数。

由于方案总数可能很大, 请你输出方案总数对 $10^9 + 7$ 取模后的值。

Input

第一行输入一个整数 T ($1 \leq T \leq 10^3$), 表示游戏次数。

接下来 T 行每行一个整数 n ($1 \leq n \leq 10^6$), 表示当前棋盘的列数。

Output

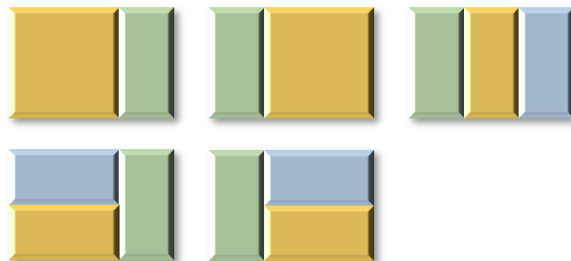
对于每次游戏输出一行一个整数, 表示铺满当前棋盘的方案总数对 $10^9 + 7$ 取模后的值。

Example

standard input	standard output
6	1
1	5
3	11
4	3
2	43
6	490028042
1000000	

Note

下图为 $n = 3$ 时对应的可能的铺棋子方案。



Problem C. Converse the string

Input file: standard input
Output file: standard output

“Converse the string?”

“You are an excellent English speaker, Hah?”

“Reverse the string! idiot!”

在题目之前，先给出一些关于字符串的定义 (如果你已经熟悉字符串，你可以跳过这一段):

字符串: 一个字符串 S 是将 n 个字符顺次排列形成的序列, n 称为 S 的长度, 表示为 $|S|$ 。

在这道题中, 字符串的下标从 1 开始, S 的第 i 个字符表示为 $S[i]$ 。

子串: 字符串 S 的子串 $S[i..j]$ ($i \leq j$), 表示 S 串中从 i 到 j 这一段, 也就是顺次排列 $S[i], S[i+1], \dots, S[j]$ 形成的字符串。

前缀: 前缀是指从串首开始到某个位置 i 结束的一个子串。

字符串 “abcba” 的前缀有 {“a”, “ab”, “abc”, “abcb”, “abcba”}。

翻转: 一个长度为 n 的字符串 S 翻转后会变为字符串 S_{rev} ,

并且满足对于所有 i , $S_{rev}[i] = S[n - i + 1]$ ($1 \leq i \leq n$)。

字典序: 对于两个长度相同的字符串 x, y , 我们称 x 在字典序上小于 y ,

当且仅当存在一个 i , 满足 $x_1 = y_1, x_2 = y_2, \dots, x_{i-1} = y_{i-1}, x_i < y_i$ 。

给定一个非空字符串 S , 你可以对 S 进行若干次操作。

在第一次操作中, 你可以选择 S 中任意长度的一个前缀, 将其翻转。

在之后的每一次操作中, 你可以选择新串中任意比上一次操作选择的前缀更长的前缀, 将其翻转。

请你求出: 在若干次操作之后, 你可以得到的字典序最小的串。

Input

第一行一个整数 T ($1 \leq T \leq 3 \times 10^5$) 表示数据组数。

接下来有 T 行, 每行有一个字符串 S , 含义见上。

保证在同一数据点内, $\sum |S| \leq 3 \times 10^5$, 字符串的所有字符都是小写字母。

Output

输出 T 行, 每行一个字符串, 表示可以得到的最小的串。

Example

standard input	standard output
5	aaabb
ababa	abccb
ccbab	a
a	jjzqzq
jzqjzq	qqw
qwq	

Note

样例 1 解释: “ababa” $\rightarrow$ “baaba” $\rightarrow$ “aabba” $\rightarrow$ “bbaaa” $\rightarrow$ “aaabb”。

样例 2 解释: “ccbab” $\rightarrow$ “abccb”。

样例 3 解释: “a”。

样例 4 解释: "jqjzq" → "qzjjzq" → "jjzqzq"。

样例 5 解释: "qwq" → "wqq" → "qqw"。

Problem D. Deliver the string

Input file: standard input
Output file: standard output

What's the meaning of "v" in "ctrl+v" ?

给定两个非空字符串 A 和 B ，同时你有一个空串 C 。

每次操作你可以选择字符串 A 的一个子串，并将这个子串加到字符串 C 的尾部 (注意这个过程并不会改变字符串 A)。

请问最少使用几次操作能使字符串 C 与字符串 B 相等。

Input

第一行一个整数 $T(1 \leq T \leq 2 \times 10^5)$ 表示数据组数。

接下来每组数据有 2 行，第一行是字符串 A ，第二行是字符串 B 。

在同一数据点内，我们保证 $\sum |A|, \sum |B| \leq 2 \times 10^5$ 。

输入数据保证一定能通过操作使字符串 C 与字符串 B 相等。

输入数据保证所有字符都是小写字母。

Output

输出 T 行，每行一个整数表示最少操作。

Example

standard input	standard output
3	2
jzq	8
jzqjzq	2
abcd	
dcbadcba	
aabaaabaaaab	
aabaaaaaba	

Note

对于第一组样例：连续两次把 A 串插入在 C 尾部即可。

对于第二组样例：依次插入"d","c","b","a","d","c","b","a"。

对于第三组样例：依次插入"aabaaaa","aba"。

Problem E. Eliminate suspicion

Input file: standard input
Output file: standard output

“Eliminate suspicion” refers to the act of removing doubt or uncertainty in a particular situation or circumstance.

This can involve gathering more information, providing evidence, or addressing concerns in a transparent and honest manner.

现在有 n 个人和 m 个犯罪现场，你需要断定哪些人肯定没有犯罪的嫌疑。

每个人有一个不在场证明 (x_i, y_i, t_i) ，表示在 t_i 时刻，他出现在了 (x_i, y_i) 的位置。

每个犯罪现场有三个参数 (x_j, y_j, t_j) ，表示在 t_j 时刻，有一件恶性事件发生在了 (x_j, y_j) 的位置。

断定一个人 i 没有犯罪嫌疑当且仅当对于所有的犯罪现场 $j (1 \leq j \leq m)$ 满足 $|x_i - x_j| + |y_i - y_j| > |t_i - t_j|$ 。

Input

第一行两个整数 n, m ($1 \leq n, m \leq 3 \times 10^5$)，表示人数和犯罪事件的数量。

接下来 n 行每行有三个整数 x_i, y_i, t_i ($1 \leq x_i, y_i, t_i \leq 10^9$)，表示第 i 个人的不在场证明的参数。

接下来 m 行每行有三个整数 x_j, y_j, t_j ($1 \leq x_j, y_j, t_j \leq 10^9$)，表示第 j 个犯罪事件的参数。

Output

输出一个整数表示有多少个人没有犯罪嫌疑。

Examples

standard input	standard output
5 5 6 5 4 8 7 4 8 8 6 2 5 8 2 1 5 6 3 8 8 1 8 6 3 8 3 6 8 6 6 5	4
5 5 6 1 6 6 5 5 8 4 1 8 5 8 4 3 8 5 2 7 7 6 4 5 8 7 8 2 6 7 1 8	2

Note

样例一解释：第一个人不满足第五个犯罪现场的条件。

样例二解释：第一个人不满足第五个犯罪现场的条件。第三个人不满足第四个犯罪现场的条件。第四个人不满足第二个犯罪现场的条件。

Problem F. Fill out the multiple choice questions

Input file: standard input
Output file: standard output

*Why did the student bring a ladder to the exam?
Because the multiple choice questions were “up” for interpretation.*

为了便于理解题意，在正式题意之前，先给出样例和样例解释。

```
5
3 A C D A
2 C B C C
5 C C D C
1 D C D C
3 C B D C
```

这是样例中的第一组数据：

第1题：第3题的答案是？ (A) A (B) C (C) D (D) A

第2题：第2题的答案是？ (A) C (B) B (C) C (D) C

第3题：第5题的答案是？ (A) C (B) C (C) D (D) C

第4题：第1题的答案是？ (A) D (B) C (C) D (D) C

第5题：第3题的答案是？ (A) C (B) B (C) D (D) C

一份答卷中必须对每个选择题给出一个选项并且选出的选项并不会与其它选项矛盾，比如上述数据中第1题，如果你选择了(C)D，那么你第3题就必须选择(D)C。

可行的答卷有：CBDBC, CCDBC, CBDDC, CCDDC。

接下来是形式化题意：

给定 n 个单项选择题 (单选题)，选择题的题面是这样的：

(i) 第 a_i 题选择题的答案是？ (括号里的 i 表示本题题目编号)

(A) b_{iA} (B) b_{iB} (C) b_{iC} (D) b_{iD}

你需要求出所有可行的答卷数量，如果两份答卷至少有一个选择题给出的答案不同，那么这两份答卷被认为是不同的。

Input

第一行一个整数 T ($1 \leq T \leq 2 \times 10^5$) 表示数据组数。

对于每组数据，第一行有一个整数 n ($1 \leq n \leq 2 \times 10^5$)。

接下来 n 行，第 i 行有一个整数 a_i 和四个大写字母 $b_{iA}, b_{iB}, b_{iC}, b_{iD}$ ($1 \leq a_i \leq n, b_{iA}, b_{iB}, b_{iC}, b_{iD} \in \{A, B, C, D\}$)，含义见题面。

输入数据保证 $\sum n \leq 2 \times 10^5$ 。

Output

输出 T 行，每行一个整数，表示满足条件的答卷数量。

因为答案可能过大，所以你需要输出答案对于 998244353 的余数。

Example

standard input	standard output
4	4
5	1
3 A C D A	2
2 C B C C	0
5 C C D C	
1 D C D C	
3 C B D C	
5	
4 D D A B	
3 A B D A	
4 B A A A	
3 D C B A	
3 D A C A	
5	
3 D C A B	
4 A C D D	
5 A B D D	
4 D A C C	
5 D C B D	
1	
1 C D A B	

Note

对于样例一：有答案"CBDBC","CCDBC","CBDDC","CCDDC"。

对于样例二：有答案"CCDAA"。

对于样例三：有答案"BBCCD","ABDCD"。

对于样例四：没有可行的答案。

Problem G. Golden jade matrix checker

Input file: standard input
Output file: standard output

*Why did the Golden jade matrix fail the checker test?
Because it couldn't find its "gold" in all the "jade" around it!*

在去年的上海大学夏季赛中，有这样一道题：

小 M 喜欢一种叫“金玉其外矩阵”的特殊矩阵，这种矩阵的性质如下：

1. 矩阵有 N 行 M 列。其中每个数都在 -10^9 到 10^9 之间。
2. 矩阵中的所有数的和是正数。
3. 对矩阵中的每一个 h 行 w 列的**子矩阵**，该子矩阵中所有数的和是负数。

在这道题中，**子矩阵** 是指从一个矩阵中选取**连续**的若干行和**连续**的若干列，用它们的交叉位置组成的新矩阵。

现在给你 N, M, h, w ，请你构造出这样的一个矩阵，或指出这样的矩阵不存在。

在今天，我们想请你写一个程序：对于一个 N 行 M 列的矩阵，并且给你参数 h 和 w ，你需要回答给出的矩阵是否是“金玉其外矩阵”。

“金玉其外矩阵”的定义与上方的往年题目中对“金玉其外矩阵”的定义一致。

Input

第一行一个整数 $T(1 \leq T \leq 2 \times 10^5)$ 表示数据组数。

对于每组数据，第一行有四个整数 N, M, h, w ，含义如上所示

$(1 \leq N, M \leq 2000, 1 \leq h \leq N, 1 \leq w \leq M)$ 。

接下来有 N 行，每行 M 个整数，其中第 i 行第 j 个数表示 $a_{i,j}(-10^9 \leq a_{i,j} \leq 10^9)$ ，表示需要判断的矩阵。

数据保证 $\sum N \times M \leq 4 \times 10^6$ 。

Output

输出 T 行，每行一个“YES”或“NO”(不包括引号)，表示给出的矩阵是否是“金玉其外矩阵”。

Example

standard input	standard output
3	YES
3 4 2 2	NO
9 3 -5 4	NO
-12 -2 2 -3	
11 0 -2 2	
2 5 1 1	
1 -1 1 -1 1	
1 -1 1 -1 1	
1 1 1 1	
1	

Note

对于第一组数据：满足所有条件。

对于第二组和第三组数据：所有元素的和大于 0，所以不是符合要求的矩阵。

本题输入量较大，请注意使用效率较高的读入方式。例如：

- 在 C++ 语言中，可以使用 `scanf` 代替 `std::cin`，使用 `printf` 代替 `std::cout`；
- 在 Java 语言中，可以使用 `BufferedReader` 代替 `Scanner`，使用 `BufferedWriter` 代替 `System.out`；
- 在 Python 语言中，可以使用 `sys.stdin.readline` 代替 `input`，使用 `sys.stdout.write` 代替 `print`。

Problem H. How to know the function

Input file: standard input
Output file: standard output

*If you want to know the function, just ask it "What's your sign?"
If it replies " $y = mx + b$ ", then you've got yourself a linear relationship.*

JYH 最近在学习方程组，他遇到了这样一个问题：

有一个各项系数均为正整数的未知项数的多项式函数 $f(x) = \sum_{i=0}^n a_i x^i$ ，每次你可以询问当 x 取一个正整数时对应的函数值，求最少询问多少次可以确定这个多项式函数。

令人惊讶的是，我们只需要询问两次便可以确定这个多项式函数：

第一次，询问 $f(1)$ ，设得到的值为 S 。

第二次，询问 $f(S+1)$ ，接下来我们把得到的数字转化为 $S+1$ 进制，得到的 $S+1$ 进制数每一位的值便与多项式函数中的系数一一对应。

如果你没有理解具体如何操作，你可以先看下方的 Note。

JYH 想知道，如果我们在询问之前便已经得知了这个多项式函数最高项的次数为多少（即 n 的值为多大），我们的最少询问次数是否会发生变化。

现在有 T 个多项式函数，对于每个多项式函数，JYH 会告知你这个多项式函数最高项的次数，请你输出你需要的最少询问次数。

Input

第一行输入一个整数 T ($1 \leq T \leq 10^2$)，表示游戏次数。

接下来 T 行每行一个整数 n ($0 \leq n \leq 10^{12}$)，表示当前多项式函数最高项的次数。

Output

对于每个多项式函数输出一行一个整数，表示需要的最少询问次数。

Example

standard input	standard output
1	2
3	

Note

下面是使用两次询问得到 $f(x) = 1 + 2x + 3x^2 + 3x^3$ 的过程：

第一步：询问 $f(1)$ ，得到 $f(1) = S = 9$ 。

第二步：询问 $f(10)$ ，得到 $f(10) = 3321$ 。

$$a_0 = 3321 \bmod 10 = 1, \lfloor \frac{3321}{10} \rfloor = 332$$

$$a_1 = 332 \bmod 10 = 2, \lfloor \frac{332}{10} \rfloor = 33$$

$$a_2 = 33 \bmod 10 = 3, \lfloor \frac{33}{10} \rfloor = 3$$

$$a_3 = 3$$

至此，我们就通过两次询问得到了 $f(x)$ 。

Problem I. I like UNO !

Input file: standard input
Output file: standard output

Why did the UNO player bring a deck of cards to their job interview?

Because they wanted to show they have a great deal of experience in "matching colors" and "playing their cards right"!

机房里有四个人 ("A","B","C","D") 正在玩 UNO, 下面是 UNO 的具体规则 (规则可能与正统规则并不相同, 所以请您仔细阅读题面):

UNO 牌分两类牌: 数字牌和功能牌。

数字牌: 数字牌由红黄蓝绿 ("R","Y","B","G") 4 色组成, 每色都有数字牌 "0"-"9"。

功能牌: 功能牌也由红黄蓝绿 ("R","Y","B","G") 4 色组成。它们的名称分别为 "跳过"、"反转"、"+2"。它们有着特殊的功能。

跳过 ("S"): 打出跳过后, 你的下家不能出牌, 轮到再下家出牌。

反转 ("R"): 打出反转后, 当前出牌时针顺序将反转 (顺时针变逆时针, 逆时针变顺时针), 轮到下家 (原来的上家) 出牌。

+2 牌 ("2"): 打出 +2 后, 下家将被罚摸 2 张牌, 并且不能出牌, 直接轮到再下家出牌。

游戏开始时, 每个玩家开始会拥有 5 张牌, 接着庄家 (一定是 "A") 开始出牌。出牌的初始顺序是顺时针 ("A->B->C->D->A")。

出牌规则: 必须和牌堆最底部的一张牌 (参照牌) 的颜色或图案相同 (如果是数字牌则为数字相同, 功能牌则为功能相同), 每次只可以出 1 张牌, 打出的牌会插入至牌堆的底部 (即成为最底部的一张牌)。

请注意: 当游戏开始的时候, 如果牌堆底部的那一张牌 (参照牌) 是一张功能牌, 那这张功能牌不会生效 (举例: 如果开场的牌堆底部是一张反转牌, 那么出牌顺序依然是顺时针)。

出牌规则举例: 假设参照牌为 "R3", 那你可以出: 红色的所有牌 (颜色相同), 所有颜色的 3 (数字相同); 假设参照牌为 "Y+2", 那你可以出: 黄色的所有牌 (颜色相同), 其他颜色的 +2 (功能相同)。

若玩家无牌可出, 则从牌堆顶部抽 1 张牌 (即抽取牌堆最顶部的牌), 若这张牌可出 (符合要求) 那么就需要立即打出, 否则就跳过, 让下家出牌。

UNO 的获胜条件是: 某个玩家出完自己的所有的牌, 当某个玩家获胜的时候, 游戏结束。

现在我们会告诉你牌堆和每个人起始拥有的五张手牌, 请你编写一个程序, 告诉我们谁最终会胜利 (输入数据保证一定会出现一个胜者并且不会出现牌堆里的牌数量不够的情况)。

鉴于大家头脑简单, 四肢发达, 因此四个人的出牌策略都是一致的, 下面是四个人的出牌策略:

如果当前可以打出一张数字牌, 那么他会打出数字编号最小的那一张牌, 如果有多张编号最小的牌, 那么按照 ("R","Y","B","G") 的颜色优先级打出;

否则, 如果当前可以打出一张功能牌, 那么他会按照 ("2", "R","S"), 的优先级打出一张功能牌, 如果有多张优先级相同的功能牌, 那么他会按照 ("R","Y","B","G") 的颜色优先级打出对应的功能牌。

举例:

如果上一张是 "Y+2", 那么打出的牌的优先级是 "Y0" < "Y9" < "R+2" < "Y+2" < "B+2" < "G+2" < "YR" < "YS"。

如果上一张是 "Y4", 那么打出的牌的优先级是 "Y0" < "Y3" < "R4" < "Y4" < "B4" < "G4" < "Y5" < "Y9" < "Y+2" < "YR" < "YS"。

每次玩家都会打出优先级最小的牌, 如果上一张是 "Y4", 而玩家手上有 "R4" 和 "Y4", 那么玩家会打出 "R4"。

Input

每张卡牌由两个字符表示, 第一个字符表示颜色 ("R","Y","B","G"), 第二个字符表示牌值 ("0"-"9", "+", "R", "S")。

输入的前四行分别表示玩家"A","B","C","D" 的手牌，每行有 5 个长度为 2 的字符串，两个字符串间由一个空格隔开。

第五行一个整数 n ($n \leq 3 \times 10^5$)，表示牌堆的大小。

接下来一行，有 n 个长度为 2 的字符串，表示牌堆中的 n 张牌，从左到右依次是牌顶到牌底的牌，两个字符串间由一个空格隔开。

输入数据保证一定会出现一个胜者并且不会出现牌堆里的牌数量不够的情况。

输入数据保证游戏结束时打出的手牌数小于 8×10^5 。

Output

输出一个字符 ("A","B","C","D")，表示胜利者。

Examples

standard input	standard output
Y6 Y6 B7 GS R+ G5 Y7 R1 B2 BR R8 Y+ GS BR B5 B0 RS G2 G3 G6 30 R4 YS B8 RR Y7 G7 G4 G0 G+ R5 R2 R1 R7 Y6 B9 R1 G2 B7 B7 G5 B6 Y0 Y3 RR G0 G8 B0 G6 G9 Y2	C
G1 G0 Y6 G9 YR G9 G8 BR G7 B5 R0 Y2 B6 BS YS R8 R+ G8 RS G3 30 B4 B9 B6 G8 R2 Y9 R7 BR Y2 GS G4 YS GS G6 B8 Y6 R4 B6 R+ Y4 Y2 G8 G4 R7 G9 R2 G9 Y1 G0 Y6	D
G3 G7 R7 R1 R3 Y3 B9 Y8 B0 BR Y5 R7 Y8 Y7 G1 YR B9 R7 R0 Y4 30 Y4 Y8 YR Y8 R0 R8 G6 B2 G3 R6 R7 B5 R4 R8 YR B4 G2 GS R3 B5 G6 B3 R6 RS Y6 B1 Y9 B1 B1 Y+	D
R3 B6 R3 B6 B9 G4 B1 R5 GR R4 R7 Y1 G3 G7 B8 YR GS YS R+ RS 30 B8 B1 R1 B3 Y+ R4 Y1 R+ R2 G0 G3 R2 B5 Y5 R3 BR YR R7 G2 B1 BR G2 B9 G4 B1 G5 G6 B0 R4 R1	C
GR R9 G7 YR B8 G0 B5 Y0 Y2 R3 Y4 B8 B3 R9 GR G5 G8 B8 R5 G4 30 YS RS G2 RR G+ R1 R7 G6 BS Y6 BS YS GR RS G4 Y6 Y5 Y7 B0 YS G3 B2 R7 R9 RS Y2 G3 R9 R9 GR	A

Note

由于篇幅限制，样例输入中牌顶到牌底的牌在展示时加入了换行符，在实际测试用例中的输入为一行。
本题的样例解释由附件形式单独给出，可以在本题的附件中找到。

Problem J. Juxtaposed brackets

Input file: standard input
Output file: standard output

*Why did the programmer get stuck in the Juxtaposed brackets?
Because he couldn't find his way out of the paren-theses!*

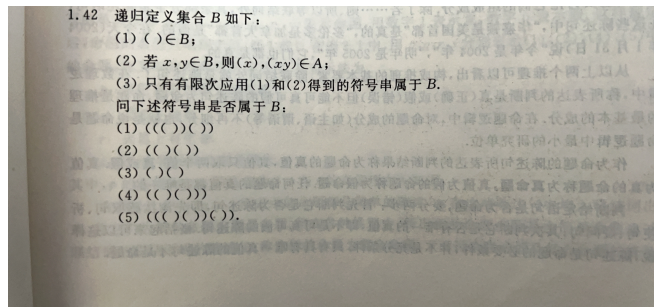
递归定义集合 B 如下:

(1) $() \in B$

(2) 若 $x, y \in B$, 则 $(x), (xy) \in B$.

给定一个非空字符串 S , 其中 S 只由两种字符"(" 和")" 组成。

请你回答这个字符串是否属于 B 。



离散数学第 3 版第 21 页

Input

第一行一个整数 $T(1 \leq T \leq 3 \times 10^5)$ 表示数据组数。

接下来有 T 行, 每行有一个字符串 S , 含义见上。

保证 $\sum |S| \leq 3 \times 10^5$, 字符串的所有字符均为“(”或“)”。

Output

输出 T 行, 如果给定的字符串属于 B , 那么输出“YES”, 否则输出“NO”。

Example

standard input	standard output
5	YES
((()())	YES
(()())	NO
()()	NO
((()())	YES
((()())()	

Note

这里展示了样例用递归定义展开成原串的过程:

样例一解释: $S \rightarrow (SS) \rightarrow ((S)S) \rightarrow (((())S) \rightarrow (((())())$ 。

样例二解释: $S \rightarrow (SS) \rightarrow ((S)S) \rightarrow ((()())$ 。

样例五解释: $S \rightarrow (SS) \rightarrow ((SS)S) \rightarrow (((()S)S) \rightarrow (((()())S) \rightarrow (((()())())$ 。