

Problem A. 坦白

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 512 megabytes

真是自作自受啊…… 逃避，逃避，
就这么不停地逃避着，
甚至还想着"干脆让谁说出来好了"，
这不是如你所愿吗？
啊……真想就这样消失呢。

— Mizuki

无人的 Sekai 里，Mizuki 正在回忆她不愿面对的一切。Ena 究竟是什么表情？自己真的已经成为所有人眼里的"怪物"了吗？

她知道维持现状对她来说永远是当下最好的选择，但是哪怕是鼓起了一次勇气，现在自己还会落得这样的结局吗……

她笑了笑，回忆过去对她来说已经于事无补，自己的结局在「那一刻」已经确定了，世界上的秘密终究不能永远是秘密。

Mizuki 可以回忆起 n 个有序的事件，每个事件都是好的或者坏的，好的事件会为她带来 $+1$ 的收益，坏的事件会为她带来 -1 的收益。

对于每个事件，如果她选择坦白，都会取消原来的收益，改为获得 $\oplus 1$ 的收益，其中 \oplus 为异或运算。

总收益值定义为 2^{64} 依次经过每个事件的收益计算后的值。

你要知道，如果 Mizuki 选择恰好 m 个事件坦白，她可以获得的最大收益，你只需要输出最大收益减掉 2^{64} 的值。

因为 Mizuki 想要知道坦白是否是好的，你需要对 $m = 0, 1, \dots, n$ 分别计算。

Input

从文件 **confess.in** 中读入数据。

本题有多组测试数据。

第一行输入一个整数 T ，代表数据组数。

接下来 T 行，每行输入一个字符串 S ，第 i 个字符为 $+$ 代表第 i 个事件是好的，反之代表这个事件是坏的。

Output

输出到文件 **confess.out** 中。

对于每组数据输出一行 $|S| + 1$ 个整数，分别代表坦白 $0, 1, \dots, m$ 次的最大收益减 2^{64} 的值。

Scoring

本题共 10 个测试点，全部测试点满足 $1 \leq T \leq 10$ ， $1 \leq |S| \leq 3 \times 10^5$ ， $S_i \in \{+, -\}$ 。

下表记 $N = \max(|S|)$ 。

测试点	$N \leq$	特殊限制
1	9	数据随机生成
2 ~ 3	18	
4 ~ 5	200	数据随机生成
6 ~ 7	10^3	
8	10^5	数据随机生成
9	10^5	
10	3×10^5	

数据随机生成: $|S| = N$, 每个字符等概率为 $+$, $-$ 中的一个。

Example

standard input	standard output
4 + -- +-- ++-+--+	1 1 -2 0 0 -1 1 1 1 2 4 6 6 6 6 4 2 0

Note

对于第一组样例, 以下为最优策略:

- $m = 1$ 时坦白第 1 个事件。

对于第二组样例, 以下为最优策略:

- $m = 1$ 时坦白第 1 个事件。
- $m = 2$ 时坦白第 1, 2 个事件。

对于第三组样例, 以下为最优策略:

- $m = 1$ 时坦白第 3 个事件。
- $m = 2$ 时坦白第 1, 3 个事件。
- $m = 3$ 时坦白第 1, 2, 3 个事件。

Problem B. 秘密

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

dXqwq 是个急性子的女孩子。

她和一些朋友在一个社团里，这 n 个人都生活在数轴上互不相同的位置 a_i 。

一些时候，某个人会收到一条秘密消息，他想要尽快把这些消息线下告诉每个人。所有人的移动速度都是 1 单位长度每秒，两个人在相同位置的时候可以交换消息，交换消息不消耗时间。

为了让大家能够尽快地收到新消息，dXqwq 想要让你编写一个程序，来计算所有人都收到消息需要花费的最小时间。

接下来 q 天，每天会发生以下三种事件的一种：

- $+$ x : 一个位置为 x 的人加入了社团，保证此时没有人的位置为 x 。
- $-$ x : 一个位置为 x 的人退出了社团，保证此时有人的位置为 x 。
- $?$ x : 一个位置为 x 的人收到了一条秘密消息，你要求出所有人都收到消息的最小时间，保证此时有人的位置为 x 。

我们认为每一天结束后，所有人都会回到自己的初始位置。

Input

从文件 secret.in 中读入数据。

第一行输入两个整数 n, q 。

第二行输入 n 个整数 a_i 。

接下来 q 行，每行输入一个字符 o 和一个整数 x ，代表事件类型和参数。

Output

输出到文件 secret.out 中。

对于每个 $o = ?$ 的询问输出一行一个浮点数，代表所有人都收到消息的最小时间，保留两位小数。

Scoring

本题共 10 个测试点，全部测试点满足 $2 \leq n, q \leq 10^5$ ， $0 \leq a_i, x \leq 10^9$ ， a_i 互不相同， $o = ?$ 时社团至少有两个人，保证存在 $o = ?$ 。

测试点	$n \leq$	$q \leq$	特殊限制
1	3	3	$o = ?$
2	9	9	$o = ?$
3	10^3	10^3	$o = ?$
4	10^5	10	$o = ?$
5 ~ 6	10^5	10^5	$o = ?$
7 ~ 8	10^3	10^3	
9 ~ 10	10^5	10^5	

Example

standard input	standard output
3 6	2.00
1 3 5	2.00
? 3	1.50
- 3	
? 5	
+ 2	
+ 4	
? 2	

Note

对于第一次询问，社团里的人分别在 $\{1, 3, 5\}$ ，只需要所有人都在 3 位置汇合即可。

对于第二次询问，社团里的人分别在 $\{1, 5\}$ ，仍然只需要所有人都在 3 位置汇合即可。

对于第三次询问，社团里的人分别在 $\{1, 2, 4, 5\}$ ，策略分为两步：

- 前两个人向后走 1 秒，后两个人向左走 1 秒，分别到 $\{2, 3, 3, 4\}$ ，中间两个人传递信息。
- 前两个人向对方走 0.5 秒，后两个人向对方走 0.5 秒，分别到 $\{2.5, 2.5, 3.5, 3.5\}$ ，此时所有人都收到信息，共花费 1.5 秒。

注意，本题不设 Special Judge，因为可以证明你的答案的小数点后第三位不是 4 或 5。

Problem C. 潜力值

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 512 megabytes

……题目的名称和描述有什么关系呢？

dXqwq 有一个长度为 n 的排列 a 和一个长度为 m 的队列 q ，初始时 q 中的元素全部为 0，她会对于 $x = 1 \sim n$ 依次执行以下操作：

- 如果队首元素 $q_1 < a_x$ ，则弹出队首，并将 a_x 放入队尾。

在操作结束后，dXqwq 定义自己的**潜力值**为 q 中所有元素的和。

dXqwq 想知道，对于给定排列 b 通过重排得到的所有排列 a ，自己会得到的**潜力值**的和。因为答案实在太大了，她只需要你输出答案对 $10^9 + 7$ 取模的值。

Input

从文件 potential.in 中读入数据。

第一行输入两个整数 n, m 。

第二行输入 n 个整数 b_i 。

Output

输出到文件 potential.out 中。

输出一行一个整数，代表潜力值的和对 $10^9 + 7$ 取模的值。

Scoring

本题共 20 个测试点，全部测试点满足 $2 \leq m \leq n \leq 500$ ， $1 \leq b_i < 10^9 + 7$ ， b_i 严格递增。

测试点	$n \leq$	$m \leq$
1	5	5
2	10	10
3 ~ 4	18	2
5 ~ 7	50	2
8 ~ 10	50	3
11 ~ 12	50	4
13 ~ 15	50	50
16 ~ 18	200	200
19 ~ 20	500	500

Examples

standard input	standard output
3 2 1 2 3	28
10 4 1 2 3 4 5 6 7 8 9 10	105878520

Note

对于第一组样例， a 有以下六种可能：

- $a = [1, 2, 3]$, $q = [2, 3]$, 潜力值为 5。
- $a = [1, 3, 2]$, $q = [3, 2]$, 潜力值为 5。
- $a = [2, 1, 3]$, $q = [1, 3]$, 潜力值为 4。
- $a = [2, 3, 1]$, $q = [2, 3]$, 潜力值为 5。
- $a = [3, 1, 2]$, $q = [3, 1]$, 潜力值为 4。
- $a = [3, 2, 1]$, $q = [3, 2]$, 潜力值为 5。

它们的潜力值之和为 $5 + 5 + 4 + 5 + 4 + 5 = 28$ 。

对于第二组样例，如果 $a = [1, 9, 2, 6, 3, 5, 7, 8, 10, 4]$ ，操作过程如下：

- 初始时 $q = [0, 0, 0, 0]$ 。
- 弹出 0，插入 1， $q = [0, 0, 0, 1]$ 。
- 弹出 0，插入 9， $q = [0, 0, 1, 9]$ 。
- 弹出 0，插入 2， $q = [0, 1, 9, 2]$ 。
- 弹出 0，插入 6， $q = [1, 9, 2, 6]$ 。
- 弹出 1，插入 3， $q = [9, 2, 6, 3]$ 。
- 5, 7, 8 都没有 9 大，所以不进行修改。
- 弹出 9，插入 10， $q = [2, 6, 3, 10]$ 。
- 弹出 2，插入 4， $q = [6, 3, 10, 4]$ 。

所以 $q = [6, 3, 10, 4]$ ，潜力值为 23。

Problem D. 括号

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

dXqwq 和 Haitang 在玩一个括号串上的游戏。

有一个初始全为黑色的括号串 S ，两个人各有一把刷子，dXqwq 的刷子是红色的，Haitang 的刷子是蓝色的。

她们会依次执行以下操作，直到所有括号都被染色，且 dXqwq 先执行操作：

- 选择一个没有被染色的括号，用自己的刷子将它染色。

在操作后，dXqwq 会取出**红色括号**的一个可以匹配的子序列，并将其长度的一半定义为游戏分数。

dXqwq 希望最大化游戏分数，而 Haitang 则希望最小化这个值，你需要输出两人都在最优策略下操作后的游戏分数。

Input

从文件 **bracket.in** 中读入数据。

本题有多组测试数据。

第一行输入一个整数 T ，代表数据组数。

接下来 T 行，每行输入一个括号串 S 。

Output

输出到文件 **bracket.out** 中。

对于每组数据输出一行一个整数，代表最优策略下的游戏分数。

Scoring

本题共 10 个测试点，全部测试点满足 $T = 10$ ， $1 \leq |S| \leq 10^5$ ， $S_i \in \{ (,) \}$ 。

下表记 $N = \max(|S|)$ 。

测试点	$N \leq$	特殊限制
1	9	数据随机生成
2	9	
3	20	数据随机生成
4 ~ 5	20	
6	100	S 不存在子序列 $()$
7 ~ 8	100	S 不存在子序列 $)()$
9	10^5	S 不存在子序列 $)()$
10 ~ 11	100	数据随机生成
12 ~ 13	100	
14 ~ 15	500	数据随机生成
16	500	
17	2×10^3	数据随机生成
18	2×10^3	
19	10^5	数据随机生成
20	10^5	

Example

standard input	standard output
4	0
) (1
(())	2
(()) ()	2
(()) (())	

Note

对于第一组样例，一种最优的决策过程如下：

- dXqwq 涂 S_1 , $S = \textcolor{red}{)} ($
- Haitang 涂 S_2 , $S = \textcolor{red}{)} (\textcolor{blue}{(}$

对于第二组样例，一种最优的决策过程如下：

- dXqwq 涂 S_1 , $S = (\textcolor{red}{(})$
- Haitang 涂 S_3 , $S = (\textcolor{red}{(}\textcolor{blue}{(})$
- dXqwq 涂 S_4 , $S = (\textcolor{red}{(}\textcolor{blue}{(}\textcolor{red}{)})$
- Haitang 涂 S_2 , $S = (\textcolor{red}{(}\textcolor{blue}{(}\textcolor{blue}{(}\textcolor{red}{)})$