

문제 A. SMUPC의 등장

시간 제한	메모리 제한
1 초	1024MB

문제

2021년 5월 8일 SMUPC 대회의 첫 개최에 신이 난 화은이는 SMUPC를 기념하기 위해 "SMUPC"를 예술적으로 출력하는 프로그램을 작성하고자 했다. 화은이는 각 알파벳에 해당하는 아스키 코드 값을 10진법으로 나타내었을 때, 각 자릿수의 합만큼 출력하도록 하는 프로그램을 만들고자 한다. 하지만 화은이는 지금 문제를 출제해야 해서 정신이 없다. 문제 출제로 정신이 없는 화은이를 위해 화은이가 만들고자 하는 프로그램을 대신 만들어 보자.

입력

첫 줄에 알파벳 소문자로만 이루어진 단어 S 가 주어진다. 이때, 입력되는 S 의 길이는 10을 넘지 않는다.

출력

단어 S 의 각 문자에 대한 프로그램의 출력값을 각 줄별로 출력한다.

예제 입출력

표준 입력(stdin)	표준 출력(stdout)
smupc	SSSSSSS mmmmmmmmmm uuuuuuuuuu pppp cccccccccccccccccc

문제 B. 눈덩이 굴리기

시간 제한	메모리 제한
1 초	1024MB

문제

눈송이들이 많은 동네인 숙명여대 운동장에서 눈사람 만들기 대회를 연다. 운동장의 길이는 N 이고 위치 1 부터 위치 N 까지만 눈이 쌓여있다. 위치 i 에 눈이 a_i 만큼 쌓여있다. 대회 규칙은 해당 운동장에서 M 초 동안 눈덩이를 굴려 눈사람을 만드는 것이다. 눈덩이의 시작 크기는 1 이다. 눈덩이의 시작 위치는 0 이다.

가장 큰 눈사람을 만들고 싶던 수수는 눈덩이를 굴리는 법을 연구했다. 눈덩이를 굴리는 방법에는 두 가지가 있다. 눈덩이를 굴리는 데에는 1 초가 소모된다.

1. 눈덩이를 현재 위치 +1 칸으로 굴린다. 현재 칸의 위치를 i 라고 하면 눈덩이의 크기는 a_{i+1} 만큼 늘어난다.
2. 눈덩이를 현재 위치 +2 칸으로 던진다. 눈덩이가 착지하며 충격을 받아 눈덩이의 크기는 원래의 크기의 반으로 줄어들고 현재 칸의 위치를 i 라고 하면 눈덩이의 크기는 a_{i+2} 만큼 늘어난다. 이 때 소수점은 절사한다. 눈덩이를 던져 크기가 0 이 되어도 눈덩이는 사라지지 않는다.

눈덩이가 운동장의 끝에 도달한 경우 남은 시간과 관계없이 눈덩이 굴리기는 끝이 난다. 대회 시간 내에 가장 크게 만들 수 있는 눈덩이의 크기를 구하는 프로그램을 작성해보자.

입력

첫째 줄에 공백을 기준으로 두 양의 정수 N, M 이 주어진다. ($1 \leq N \leq 100, 1 \leq M \leq 10$)

둘째 줄에 길이가 N 인 수열 a 가 주어진다. ($1 \leq a_i \leq 1000000$)

출력

첫째 줄에 대회 시간 내에 가장 크게 만들 수 있는 눈덩이의 크기를 출력한다.

예제 입출력

표준 입력(stdin)	표준 출력(stdout)
10 5 1 3 4 5 6 7 8 10 12 14	28

문제 C. 헌내기는 친구가 필요해

시간 제한	메모리 제한
C, C++ : 1 초 Java : 2 초 Python : 2 초	1024MB

문제

2020 년에 입학한 헌내기 도연이가 있다. 도연이는 비대면 수업 때문에 학교에 가지 못해 학교에 아는 친구가 없었다. 드디어 대면 수업을 하게 된 도연이는 어서 캠퍼스 내의 사람들과 친해지고 싶다.

도연이가 다니는 대학의 캠퍼스는 $N \times M$ 크기이며 캠퍼스에서 이동하는 방법은 벽이 아닌 상하좌우로 이동하는 것이다. 예를 들어, 도연이가 (x, y) 에 있다면 이동할 수 있는 곳은 $(x+1, y)$, $(x, y+1)$, $(x-1, y)$, $(x, y-1)$ 이다. 단, 캠퍼스의 밖으로 이동할 수는 없다.

불쌍한 도연이를 위하여 캠퍼스에서 도연이가 만날 수 있는 사람의 수를 출력하는 프로그램을 작성해보자.

입력

첫째 줄에는 캠퍼스의 크기를 나타내는 두 정수 N ($1 \leq N \leq 600$), M ($1 \leq M \leq 600$)이 주어진다.

둘째 줄부터 N 개의 줄에는 캠퍼스의 정보들이 주어진다. **O** 는 빈 공간, **X** 는 벽, **I** 는 도연이, **P** 는 사람이다. **I** 가 한 번만 주어짐이 보장된다.

출력

첫째 줄에 도연이가 만날 수 있는 사람의 수를 출력한다. 단, 아무도 만나지 못한 경우 **TT** 를 출력한다.

예제 입출력

표준 입력(stdin)	표준 출력(stdout)
3 5 000P0 0I00X 000XP	1

문제 D. SMUPC 계산기

시간 제한	메모리 제한
1.2 초	1024MB

문제

SMUPC 를 기념하기 위해 ALGOS 와 DSC Sookmyung 에서는 SMUPC 의 각 글자로 계산이 이루어지는 계산기를 만들었다. 가은이와 헤민이는 이 계산기와 같은 방식으로 작동하는 프로그램을 만들고자 한다. 가은이와 헤민이는 몇 가지 계산 작업을 통해 SMUPC 에 해당하는 각 기호가 아래의 표와 같이 동작함을 밝혀낼 수 있었다.

[SMUPC 계산기의 동작 방식]

알파벳	S	M	U	P	C
해당 수식	-	*	/	+	여태까지의 계산 결과값 반환

SMUPC 계산기는 기존의 사칙연산 방식과는 다르게 앞에서부터 순서대로 계산이 이루어진다. 단, 이 계산기에서 음수를 양수로 나누는 경우는 C++14 의 기준을 따른다. 이는 음수에 -1 을 곱해 양수로 바꾼 뒤 몫을 취하고, 그 몫에 -1 을 곱한 것과 같다. 예를 들어, $5/3=1$, $(-5)/3=-1$ 로 계산된다. 더불어 SMUPC 계산기에 입력하는 수식은 다음과 같은 규칙을 따라야 한다고 한다.

- 수식은 0 부터 9 까지의 숫자와 S, M, U, P, C 의 알파벳만을 포함할 수 있다.
- 수식은 수로 시작해야 하며, 알파벳으로 끝나야 한다.
- 수식에 음수를 입력할 수 없다.
- 수식에 입력하는 수는 0 으로 시작할 수 있다.
- 알파벳 C 로 계산 결과값을 반환한 후에 추가적인 계산을 원한다면 알파벳 기호를 추가적으로 사용하여 입력을 이어나가야 한다.
- 알파벳 S, M, U, P 는 서로 연속해서 입력할 수 없으며 알파벳 S, M, U, P 을 입력한 직후 다른 수의 입력 없이 알파벳 C 를 바로 입력할 수 없다.

SMUPC 계산기와 같은 작업을 수행할 수 있는 프로그램을 작성하여라.

입력

첫째 줄에 수식에 들어갈 기호의 개수 $N(1 \leq N \leq 500000)$ 이 주어진다.

둘째 줄에 N 개의 기호가 사용된 수식이 주어진다. 수식의 길이는 1000000 을 넘지 않는다. 단, 계산 도중 입력되는 수나 계산 결과값의 범위는 -2^{31} 이상 $2^{31}-1$ 이하이다. 더불어 어떤 수를 0 으로 나누는 경우는 존재하지 않는다.

출력

알파벳 **C** 가 나올 때마다 여태까지의 계산 결과 값을 띄어쓰기 간격으로 출력하도록 한다. 알파벳 **C** 가 한번도 나오지 않을 경우 **NO OUTPUT** 을 출력한다.

예제 입출력

표준 입력(stdin)	표준 출력(stdout)
5 3S2M3U1P2C	5
4 0328CS00325CC	328 3 3
1 000232738S	NO OUTPUT

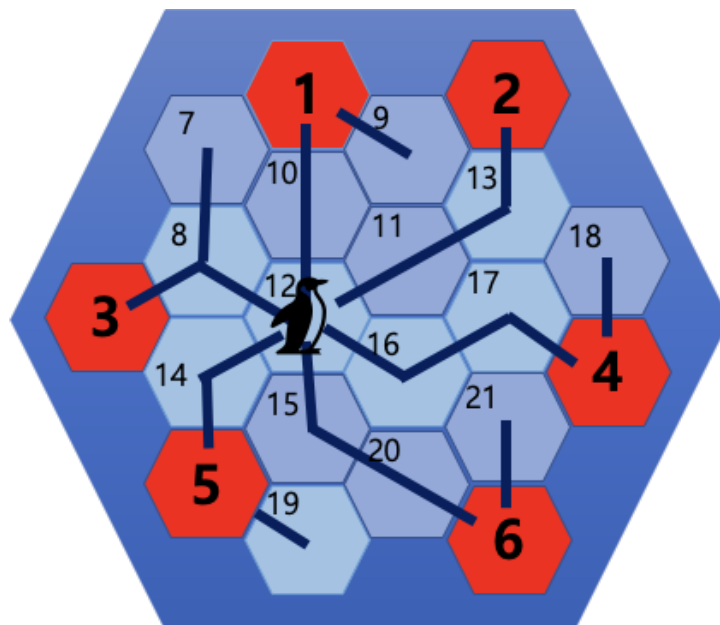
문제 E. 얼음깨기 펭귄

시간 제한	메모리 제한
C, C++ : 1 초 Java : 3 초 Python : 3 초	1024MB

문제

도도는 심심해서 보드게임 카페에 갔다. 마침 평소에 즐겨 했던 얼음 깨기 펭귄의 업그레이드 버전으로 특수 얼음 깨기 펭귄 보드게임이 나와 직접 플레이해 보기로 결정했다. 특수 얼음 깨기 펭귄 게임은 특수 안경이 있어 특수 안경을 끼고 얼음들을 보면 얼음들 간의 연결 관계가 보인다.

특수 얼음 깨기 펭귄 게임에 있는 얼음의 종류로는 지지대의 역할을 하는 얼음과 일반 얼음 총 2 가지의 얼음이 존재한다. 지지대의 역할을 하는 얼음의 경우, 빨간색으로 구분하여 볼 수 있으며 일반 얼음을 지탱해 주어 일반 얼음들이 깨지지 않도록 도와준다. 일반 얼음의 경우에는 1 개의 지지대만이 연결되어 있어도 얼음이 깨지지 않지만 펭귄이 올라가 있는 얼음은 2 개 이상의 지지대의 역할을 하는 얼음이 연결되어 있어야만 얼음이 깨지지 않는다. 이때, 지지대가 연결되어 있다는 것은 지지대로부터 서로 다른 일반 얼음들을 통해 연결 관계가 이어져 있는 것을 이야기한다. 특수 얼음 깨기 펭귄 게임에서 도도가 펭귄을 떨어뜨리지 않고 최대 몇 개의 얼음을 깰 수 있을까?



입력

첫째 줄에 얼음 블록의 개수 $N(3 \leq N \leq 328000)$ 과 지지대의 역할을 하게 되는 얼음의 개수 $S(2 \leq S \leq N-1)$,
펭귄이 위치한 얼음 블록의 번호 $P(S < P \leq N)$ 가 주어진다. 지지대의 역할을 하게 되는 얼음의 개수가 S 일
때, 1 번부터 S 번까지의 얼음은 지지대의 역할을 한다.

둘째 줄부터 $N-1$ 개의 줄에 두 개의 정수 A, B 가 주어진다. 이는 A 번 얼음과 B 번 얼음이 연결되어
있음을 의미하며 같은 연결은 여러 번 주어지지 않는다.

게임 시작 시 펭귄은 일반 얼음 위에 위치해 있고 어떤 얼음도 깨지지 않은 상태로 시작하게 된다. 각
얼음들은 1 번부터 N 번까지 정수 번호로 주어져 있으며 서로 다른 두 얼음을 잇는 경로는 하나뿐이다.
더불어 서로 다른 지지대가 펭귄이 올라가 있는 얼음을 거치지 않고 연결되어 있는 경우는 없다.

출력

플레이어가 펭귄을 떨어트리지 않고 깰 수 있는 얼음의 최대 개수를 구하여라. 지지대의 역할을 하는
얼음 역시 깰 수 있는 얼음에 속한다.

예제 입출력

표준 입력(stdin)	표준 출력(stdout)
21 6 12	16
1 9	
1 10	
10 12	
2 13	
13 11	
11 12	
3 8	
8 7	
8 12	
5 19	
5 14	
14 12	
6 20	
6 21	
20 15	
15 12	
4 18	
4 17	
17 16	
16 12	

문제 F. 펭귄 네비게이터

시간 제한	메모리 제한
1 초	1024MB

문제

펭귄은 집까지 가고 싶다. 하지만 누군가에 의해 얼음길이 다 깨져 집에 갈 수 없게 되었다. 현진이는 펭귄들을 위해 얼음길을 만들어줄 예정이다. 얼음길은 $2 \times N$ 크기이며 얼음마다 중복하지 않고 1 부터 $2N$ 까지의 번호를 붙일 수 있다. 이 펭귄들에겐 특이한 습성이 있다. 펭귄은 현재 위치의 우측이나 아래로만 이동한다. 하지만 이동한 곳의 번호가 원래 위치의 숫자보다 커야만 한다. 펭귄이 어떤 움직임을 하든지 무조건 집으로 갈 수 있게 만드는 얼음길의 개수를 구하는 프로그램을 작성해보자.

입력

첫 번째 줄에는 얼음길 가로의 길이 N ($1 \leq N \leq 10000$)이 주어진다.

출력

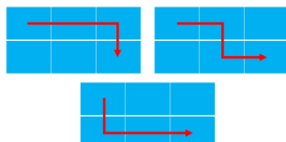
첫째 줄에 만들어지는 얼음길의 개수를 10^9+7 로 나눈 나머지를 출력한다.

예제 입출력

표준 입력(stdin)	표준 출력(stdout)
3	5

1	2	3	1	2	4	1	2	5
4	5	6	3	5	6	3	4	6
	1	3	4		1	3	5	
	2	5	6		2	4	6	

다음과 같은 얼음길 5 개는 (1, 1)에서 (3, 2)까지 가는 세 가지 방법을 모두 사용 가능하다.



문제 G. 도도의 수학놀이

시간 제한	메모리 제한
2 초	1024MB

문제

길이가 N 인 수열이 주어진다. 도도는 이 수열의 수를 이어붙여 180 도 회전시켰을 때 가장 큰 수를 만들려고 한다. 각 숫자를 180 도 회전시켰을 때 환원되는 숫자는 다음과 같다.

- 0 -> 0
- 1 -> 1
- 2 -> 2
- 5 -> 5
- 6 -> 9
- 8 -> 8
- 9 -> 6

숫자 3, 4, 7 이 포함되어있는 수는 입력으로 주어지지 않는다. 모든 수를 이어붙인 수를 뒤집었을 때, 불필요한 0 들은 제거된다. 단 한 번, 한 수를 두 번 사용할 수 있다.

이어붙인 수 중 180 도 회전시켰을 때 가장 큰 값을 갖게 되는 수를 구하는 프로그램을 작성해보자.

입력

첫째 줄에 정수 N ($1 \leq N \leq 100000$)이 주어진다.

둘째 줄에는 길이가 N 인 수열 a ($1 \leq a_i \leq 1000000$)가 주어진다.

출력

첫째 줄에 숫자를 이어붙여 만든 수 중 180 도 회전시켰을 때 가장 큰 값을 갖게 되는 수를 출력한다.

예제 입출력

표준 입력(stdin)	표준 출력(stdout)
4 1 6 8 9	19866