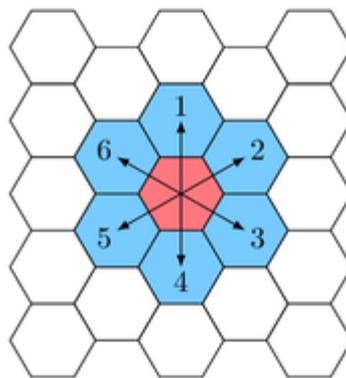


육각형 영역

육각형 타일로 이루어진 칸들이 무한하게 놓여 있고, 재현이는 이 중 "시작 칸"이라고 부르는 칸 위에서 서 있다. 두 칸은 모서리를 공유하면 이웃하고 있다고 하자. 재현이는 매번 다음 그림과 같이, 1부터 6까지 번호가 붙은 방향을 따라서 이웃한 칸 중 하나로 이동할 수 있다.

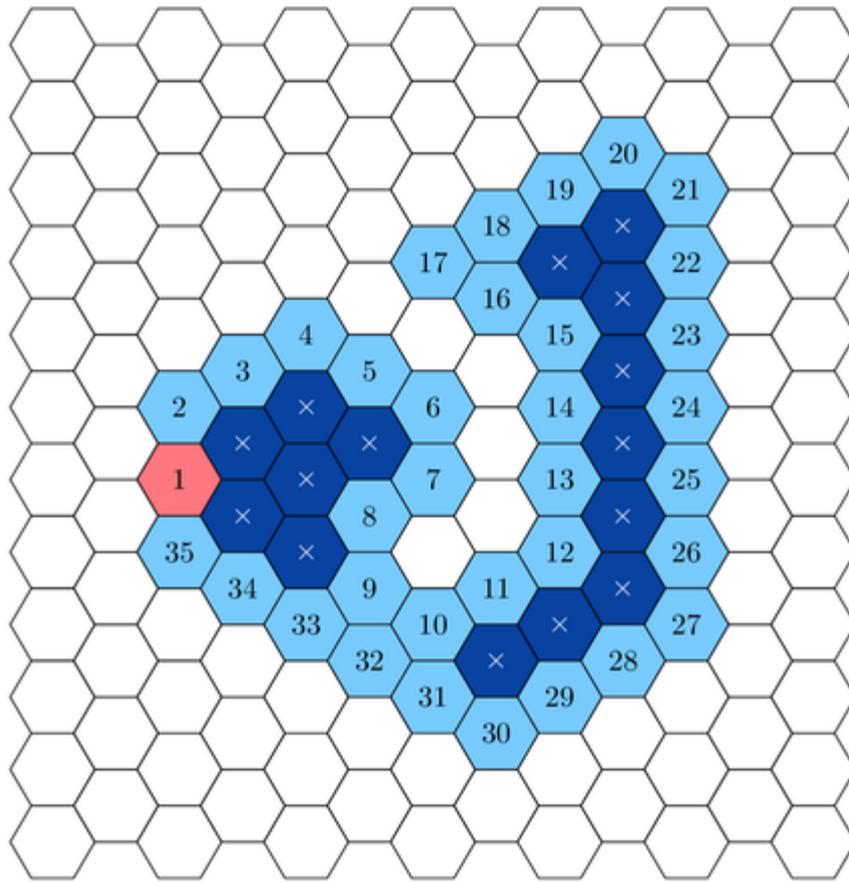


재현이가 총 N 번 움직였을 때 만드는 칸들로 이루어지는 경로는 영역을 형성한다. i 번째 움직일 때는 $D[i]$ 방향으로 $L[i]$ 칸을 이동한다. 이 경로는 다음과 같은 특징이 있다.

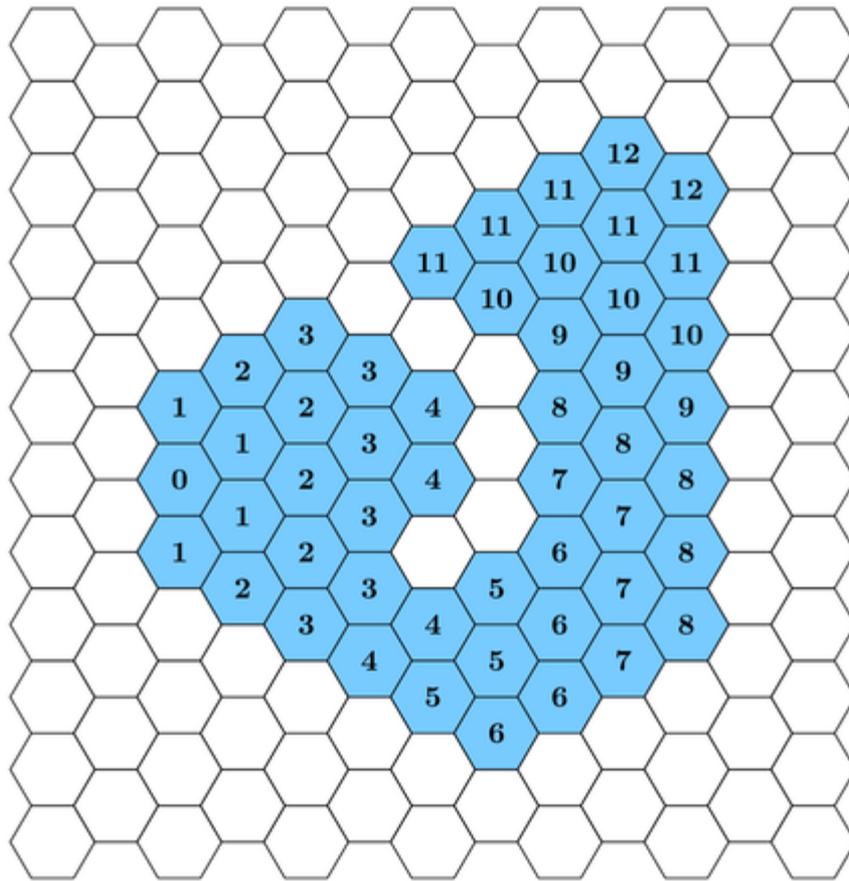
- 이 경로는 *닫혀있는데*, 마지막에 도착하는 칸은 시작 칸과 동일하다는 뜻이다.
- 이 경로는 *단순한데*, 맨 처음 시작한 칸만 빼고 모든 칸은 최대 한 번 방문한다는 뜻이다. 맨 처음 시작한 칸은 맨 마지막까지 합쳐서 정확하게 두 번 방문한다.
- 이 경로는 *드러나 있는데*, 경로에 포함되는 모든 칸은 최소한 한 개의 경로에 포함되지 않으면서 *내부에* 있지 않는 칸과 이웃한다.
 - 만약 어떤 칸이 경로에 포함되지 않으면서, 경로에 포함되는 칸을 지나지 않고 방문할 수 있는 칸의 개수가 유한하다면 이 칸은 *내부에* 있다고 한다.

다음은 재현이가 갈 수 있는 경로 중 하나의 예이다.

- 1번 칸 (핑크색)이 시작 (그리고 마지막) 칸이다.
- 열은 파란색 칸들은 경로에 포함되는 칸들이며, 방문 순서가 칸 안에 쓰여 있다.
- x표시가 되어 있는 짙은 파란색 칸들은 내부에 있는 칸들이다.



형성된 영역은 경로에 포함되거나, 내부에 있는 모든 칸들로 이루어진다. 영역 안의 칸 c 의 거리는 영역 안의 칸들만 방문해서 시작 칸부터 칸 c 에 도착할 때까지 필요한 움직임의 최소값이다. 영역 안의 칸에 대한 점수는 $A + d \times B$ 인데, A 와 B 는 재현이가 미리 정한 상수값이며, d 는 이 칸의 거리이다. 다음은 위 예제의 경로에 의해 형성된 영역 안의 각 칸들의 거리를 보여준다.



재현이가 N 번 움직일 때 만드는 칸들로 이루어진 경로에 의해 형성된 영역의 모든 칸의 점수의 총합을 구하는 프로그램을 작성하시오. 점수의 총합은 매우 큰 값일 수 있으므로, 이 값을 $10^9 + 7$ 로 나눈 나머지를 구하시오.

상세 구현

다음 함수를 구현해야 한다.

```
int draw_territory(int N, int A, int B, int[] D, int[] L)
```

- N : 움직임의 수.
- A, B : 점수 계산에 필요한 상수
- D : 길이 N 인 배열로, $D[i]$ 는 i 번째 움직임의 방향
- L : 길이 N 인 배열로, $L[i]$ 는 i 번째 움직임에서 이동한 칸 수
- 이 함수는 경로에 의해 형성된 영역의 모든 칸의 점수의 총합을 $10^9 + 7$ 로 나눈 나머지를 리턴한다.
- 이 함수는 정확하게 한 번 호출된다.

예제

다음 호출을 생각해보자.

```
draw_territory(17, 2, 3,
               [1, 2, 3, 4, 5, 4, 3, 2, 1, 6, 2, 3, 4, 5, 6, 6, 1],
               [1, 2, 2, 1, 1, 1, 1, 2, 3, 2, 3, 1, 6, 3, 3, 2, 1])
```

이는 위 예제에서 설명한 것과 동일하다. 다음 표는 영역에서 가능한 거리마다 각 칸의 점수를 보여준다.

거리	칸 수	각 칸의 점수	총 점수
0	1	$2 + 0 \times 3 = 2$	$1 \times 2 = 2$
1	4	$2 + 1 \times 3 = 5$	$4 \times 5 = 20$
2	5	$2 + 2 \times 3 = 8$	$5 \times 8 = 40$
3	6	$2 + 3 \times 3 = 11$	$6 \times 11 = 66$
4	4	$2 + 4 \times 3 = 14$	$4 \times 14 = 56$
5	3	$2 + 5 \times 3 = 17$	$3 \times 17 = 51$
6	4	$2 + 6 \times 3 = 20$	$4 \times 20 = 80$
7	4	$2 + 7 \times 3 = 23$	$4 \times 23 = 92$
8	5	$2 + 8 \times 3 = 26$	$5 \times 26 = 130$
9	3	$2 + 9 \times 3 = 29$	$3 \times 29 = 87$
10	4	$2 + 10 \times 3 = 32$	$4 \times 32 = 128$
11	5	$2 + 11 \times 3 = 35$	$5 \times 35 = 175$
12	2	$2 + 12 \times 3 = 38$	$2 \times 38 = 76$

점수의 총합은 $2 + 20 + 40 + 66 + 56 + 51 + 80 + 92 + 130 + 87 + 128 + 175 + 76 = 1003$ 이다. 따라서, `draw_territory` 함수의 리턴값은 1003이어야 한다.

제약 조건

- $3 \leq N \leq 200\,000$
- $0 \leq A, B \leq 10^9$
- $1 \leq D[i] \leq 6$ (모든 $0 \leq i \leq N - 1$)
- $1 \leq L[i]$ (모든 $0 \leq i \leq N - 1$)
- L 의 모든 원소의 합은 10^9 을 넘지 않는다.
-

부분 문제

1. (3 점) $N = 3, B = 0$
2. (6 점) $N = 3$
3. (11 점) L 의 모든 원소의 합은 2000을 넘지 않는다.

4. (12 점) $B = 0$ 이고 L 의 모든 원소의 합은 200 000을 넘지 않는다.
5. (15 점) $B = 0$
6. (19 점) L 의 모든 원소의 합은 200 000을 넘지 않는다.
7. (18 점) $L[i] = L[i + 1]$ (for all $0 \leq i \leq N - 2$)
8. (16 점) 추가적인 제약 조건이 없다.

샘플 그레이더

샘플 그레이더는 다음 양식으로 입력을 읽는다.

- line 1: $N A B$
- line $2 + i$ ($0 \leq i \leq N - 1$): $D[i] L[i]$

샘플 그레이더는 다음 양식으로 답을 출력한다.

- line 1: draw_territory의 리턴값