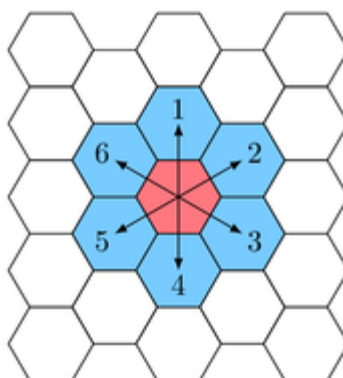


六角形領土

Pak Dengklek 正站在一塊鋪有無限個六邊形階磚的地上。在這裡, 每一塊階磚稱為一個單元格, 而他所站在的階磚被稱為初始單元格。如果兩塊六角形階磚有著一條共同的邊, 則它們稱為相鄰。在一步中, Pak Dengklek 可以通過朝六個可能的方向之一移動, 這 6 個方向以 1 到 6 作為編號, 如下圖所示。

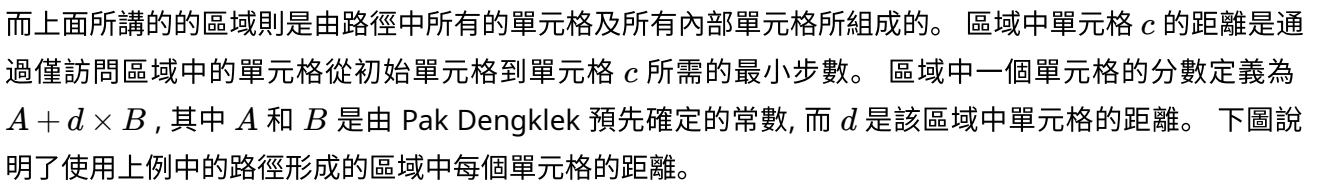


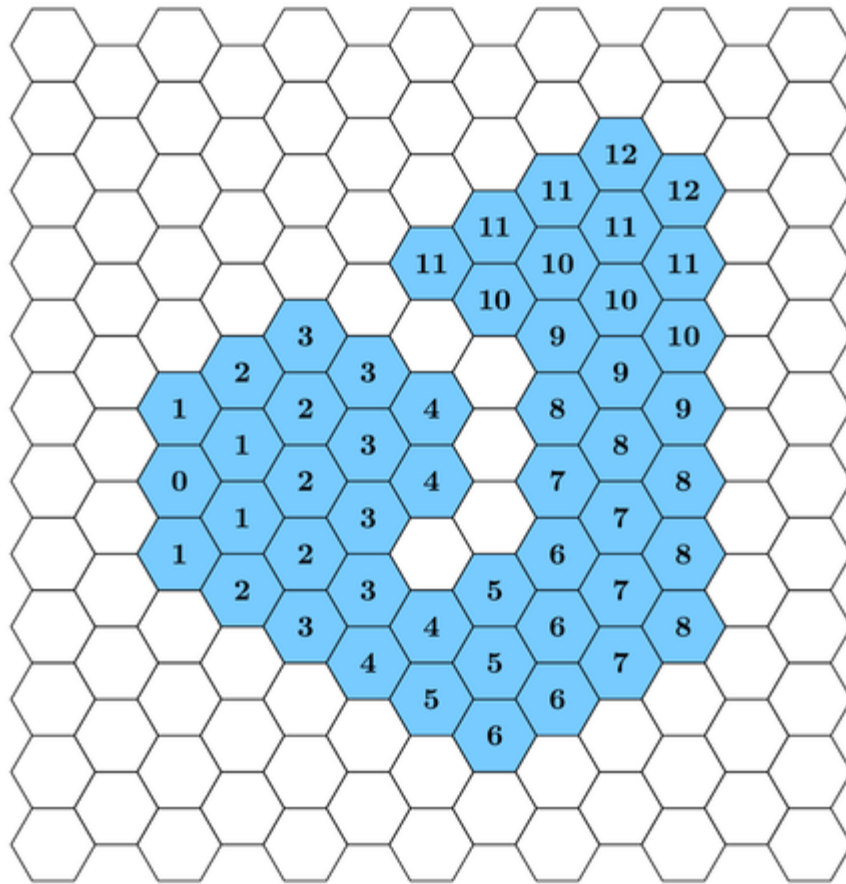
Pak Dengklek 將要建立一個區域, 這個區域是由一系列單元格組成的路徑所形成的。而這個路徑是由一系列 N 個移動所組成的。在第 i 個移動中, 首先是選擇方向 $D[i]$, 然後沿所選方向執行 $L[i]$ 步。該路徑具有以下屬性:

- 路徑是 封閉 的, 這表示路徑序列末尾的單元格與序列開頭的單元格相同。
- 路徑是 簡單 的, 這意味著每個單元格最多可以訪問一次, 除了初始單元格外, 該單元格只能被兩次訪問 (在開始和結束時)。
- 路徑是 暴露 的, 這意味著路徑中的每個單元格都至少與一個不在路徑中且不在 內部 的單元格相鄰。
 - 一個單元格被稱為 內部 單元格, 若它不在路徑中, 並且在不訪問任何在路徑上的單元格的情形下通過任何合法的移動, 你只能訪問有限數目的單元格。

以下是 Pak Dengklek 可以遵循的路徑示例。

- 編號為 1 (粉紅色) 的單元格是初始 (也是最終) 單元格。
- 有編號的單元格 (淺藍色) 是路徑中的單元格, 按照它們被訪問的順序編號。
- 有交叉的單元格 (深藍色) 是內部的單元格。





幫助 Pak Denglek 計算他將進行的 N 動作形成的區域內所有單元格的總分。由於總分可能很大, 因此答案只要取 $\text{mod } (10^9 + 7)$ 。

實現細節

您應該實現以下函數。

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

- N : 移動次數。
- A, B : 用於計算分數的常數。
- D : 長度為 N 的數組, 其中 $D[i]$ 是第 i 次移動方向。
- L : 長度為 N 的數組, 其中 $L[i]$ 是第 i 次移動的步數。
- 此函數應以 $10^9 + 7$ 為模, 返回繪製區域的總分。
- 此過程僅被調用一次。

例子

考慮以下調用：

```
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])
```

這些動作實際上與上面說明中所說明的圖形相同。下表列出了區域中每個可能距離的每個單元格的分數。

| 距離 | 單元格數目 | 每個單元格的分數 | 總分 |
|----|-------|------------------------|---------------------|
| 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 | $2 + 12 \times 3 = 38$ | $2 \times 38 = 76$ |

總得分為 $2 + 20 + 40 + 66 + 56 + 51 + 80 + 92 + 130 + 87 + 128 + 175 + 76 = 1003$ 。因此, 過程 `draw_territory` 應返回 1003。

限制

- $3 \leq N \leq 200\,000$
- $0 \leq A, B \leq 10^9$
- $1 \leq D[i] \leq 6$ (對於所有 $0 \leq i \leq N - 1$)
- $1 \leq L[i]$ (對於所有 $0 \leq i \leq N - 1$)
- L 的所有元素的總和不超過 10^9 。
- 路徑是封閉, 簡單且暴露的。

子任務

1. (3分) $N = 3, B = 0$
2. (6分) $N = 3$
3. (11分) L 的所有元素之和不超過 2000。

- 4. (12分) $B = 0$, L 的所有元素的總和不超過 200 000。
- 5. (15分) $B = 0$
- 6. (19分) L 的所有元素之和不超過 200 000。
- 7. (18分) $L[i] = L[i + 1]$ (對於所有 $0 \leq i \leq N - 2$)
- 8. (16分) 無其他限制。

樣例評分器

樣例評分器會以以下格式讀取輸入：

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

評分員以以下格式打印您的答案：

- 第 1 行：`draw_territory`的返回值