Problem L. Pianissimo

Source file name:	Pianissimo.c, Pianissimo.cpp, Pianissimo.java, Pianissimo.py
Input:	Standard
Output:	Standard

Composers use dynamics in sheet music to indicate how loud or soft the notes shall be played. Consider an 8-scale dynamic system:

- `ppp`: pianississimo (the softest)

- `pp`: pianissimo (very soft)

- `p`: piano (soft)

- `mp`: mezzopiano (moderately soft)

- `mf`: mezzoforte (moderately loud)

- `f`: forte (loud)

- `ff`: fortissimo (very loud)

- `fff`: fortississimo (the loudest)

When a musician performs, it practically does not matter how absolutely loud or soft a note is played. The audience only hear their relative difference. Suppose we use numbers to describe the absolute power of a note, where larger numbers indicate larger absolute power. Consider a musician who plays all her notes with power up to 100. When she goes from power 40 to 70, some audience may think that she goes from `p` to `f`, while some others may think that she goes from `pp` to `mf`.

The musician's interpretation must be consistent throughout the entire piece. That is, between two notes of different dynamics, the note with a louder dynamic must always have a *strictly* larger absolute power. Between two notes of the same dynamic, their power can vary (usually slightly, but there is no strict requirement).

You just heard a musician perform a piece with n notes. You know the piece very well and you remember all the dynamics that the composer wrote. How consistent is the musician's performance according to the written dynamics? You would like to count the unordered pairs of notes that violate the dynamics. Such a pair of notes is not necessarily consecutive and could be far from each other in the sheet music.

Input

The first line of input contains two integers n and m ($1 \leq m \leq n \leq 2 \cdot 10^5$), the number of notes and the number of dynamic changes.

The next n lines each have an integer between 1 and 10^9, giving the absolute power of a note played by the musician. Larger numbers indicate larger absolute power.

The next m lines each have a 1-based index i and a dynamic d, which mean the composer wrote that starting from note i the dynamic should change to d. Each dynamic is a string from the 8 dynamics: `ppp`, `pp`, `p`, `mp`, `mf`, `f`, `ff`, `fff`. The dynamics have distinct indices and are given in increasing order of their index. The first dynamic always starts from note 1.

Output

Output a single integer, the number of unordered pairs of notes that violate the dynamics.

Example

Input	Output
8 6 10 15 20 30 19 20 35 40 1 p 3 f 4 ff 5 p 7 f 8 ff	2
12 5 5 10 9 20 30 40 90 90 11 10 4 3 1 p 4 f 7 ff 9 p 11 ppp	0