Competition 05

Due: September 28

GOOD LUCK, EVERYBODY!

Problem A. Bessie the cow has a new cell phone and enjoys sending text messages, although she keeps making spelling errors since she has trouble typing on such a small screen with her large hooves. Farmer John has agreed to help her by writing an auto-completion app that takes a partial word and suggests how to complete it.

The auto-completion app has access to a dictionary of W words, each consisting of lowercase letters in the range from `a` to `z`, where the total number of letters among all words is at most $1,000,000$. The app is given as input a list of N partial words ($1 \leq N \leq 1000$), each containing at most 1000 lowercase letters. Along with each partial word i, an integer K_i is also provided, such that the app must find the K_i-th word in alphabetical order that has partial word i as a prefix. That is, if one ordered all of the valid completions of the i-th partial word, the app should output the completion that is K_i-th in this sequence.

input format: The input consists of $\rho \geq 1$ *challenges*, each in the format:

- Line 1: Two integers: W and N.
- Lines $2 \ldots W+1$: Line $i+1$ has the i-th word in the dictionary.
- Lines $W+2 \ldots W+N+1$: Line $W+i+1$ has a single integer K_i followed by a partial word.

No extra lines intervene between two successive challenges. The last challenge is followed by a line containing a single copy of the character `#`.

sample challenge:

```
10 3
dab
ba
ab
daa
aa
aaa
aab
abc
ac
dadba
4 a
2 da
4 da
```

sample details: There are 10 words in the dictionary: `dab`, `ba`, etc. The app is given 3 requests: to return the 4th match of `a`, the 2nd match of `da`, and the 4th match of `da`.

output format: The output should consist of ρ *responses*, each in the format:

- Lines $1 \ldots N$: Line i should contain the index within the dictionary (an integer in the range $1 \ldots W$) of the K_i-th completion (in alphabetical order) of the i-th partial word, or -1 if there are less than K_i completions.

No extra lines should intervene between two successive responses.

sample response:

```
3
1
-1
```

sample details: The completions of `a` are {`aa`, `aaa`, `aab`, `ab`, `abc`, `ac`}. The 4th is `ab`, which is listed on line 3 of the dictionary. The completions of `da` are `daa`, `dab`, `dadba`. The 2nd is `dab`, listed on line 1 of the dictionary. There is no 4th completion of `da`.

Problem B. Due to recent budget cuts, Farmer John has downsized his farm so that the grazing area for his cows is only a 5 meter by 5 meter square field! The field is laid out like a 5×5 grid of 1 meter by 1 meter squares, with $(1, 1)$ being the location of the upper-left square, and $(5, 5)$ being the location of the lower-right square:

```
(1,1) (1,2) (1,3) (1,4) (1,5)
(2,1) (2,2) (2,3) (2,4) (2,5)
(3,1) (3,2) (3,3) (3,4) (3,5)
(4,1) (4,2) (4,3) (4,4) (4,5)
(5,1) (5,2) (5,3) (5,4) (5,5)
```

Each square is filled with delicious grass, except for K barren squares ($0 \leq K \leq 22$, K even), which have no grass. Bessie the cow starts grazing in square $(1, 1)$, which is always filled with grass, and Mildred the cow starts grazing in square $(5, 5)$, which also is always filled with grass.

Each half-hour, Bessie and Mildred finish eating all the grass in their respective squares and each both move to adjacent grassy squares (north, south, east, or west). They want to consume all the grassy squares and end up in exactly the same final location. Please compute the number of different ways this can happen. Bessie and Mildred always move onto grassy squares, and they never both move onto the same square unless that is the very last grassy square remaining.

input format: The input consists of $\rho \geq 1$ *challenges*, each in the format:

- Line 1: The integer K.
- Lines $2 \ldots 1+K$: Each line contains the location (i, j) of a non-grassy square by listing the two space-separated integers i and j.

No extra lines intervene between two successive challenges. The last challenge is followed by a line containing a single copy of the character `#`.

sample challenge:

```
4
3 2
3 3
3 4
3 1
```

sample details: The grid looks like this (where `.` denotes a grassy square, `x` denotes a non-grassy square, `b` denotes the start location of Bessie, and `m` denotes the start location of Mildred):

```
b . . . .
. . . . .
x x x x .
. . . . .
. . . . m
```

output format: The output should consist of ρ *responses*, each in the format:

- Line 1: The number of different possible ways Bessie and Mildred can walk across the field to eat all the grass and end up in the same final location.

No extra lines should intervene between two successive responses.

sample response:

```
1
```

sample details: There is only one possible solution, with Bessie and Mildred meeting at $(3, 5)$:

```
b  b--b  b--b
|  |  |  |  |
b--b  b--b  b
            |
x  x  x  x b/m
            |
m--m--m--m--m
|
m--m--m--m--m
```

Problem C. Bessie and her friends are playing a unique version of poker involving a deck with N different ranks ($1 \leq N \leq 100,000$), conveniently numbered $1 \ldots N$ (a normal deck has $N = 13$). In this game, there is only one type of hand to play: one may choose a card labeled i and a card labeled j and play one card of every value from i to j. This type of hand is called a "straight".

Bessie's hand currently holds a_i cards of rank i ($0 \leq a_i \leq 100000$), for each $i = 1, \ldots, N$. Help her find the minimum number of hands she must play to get rid of all her cards.

input format: The input consists of $\rho \geq 1$ *challenges*, each in the format:

- Line 1: The integer N.
- Lines $2 \ldots 1+N$: Line $i+1$ contains the value of a_i.

No extra lines intervene between two successive challenges. The last challenge is followed by a line containing a single copy of the character `#`.

sample challenge:

```
5
2
4
1
2
3
```

sample details: There are 5 different ranks. Bessie's current hand contains 2 cards of rank 1, 4 cards or rank 2, 1 card of rank 3, 2 cards of rank 4, and 3 cards of rank 5.

output format: The output should consist of ρ *responses*, each in the format:

- Line 1: The minimum number of straights Bessie must play to get rid of all her cards.

No extra lines should intervene between two successive responses.

sample response:

```
6
```

sample details: Bessie can play a straight from 1 to 5, a straight from 1 to 2, a straight from 4 to 5, two straights from 2 to 2, and a straight from 5 to 5, for a total of 6 rounds necessary to get rid of all her cards.