# Finite and infinite languages

Let us recall some necessary definitions. The concepts defined here are probably very well known to you, we provide the definitions just for the sake of completness.

A **finite automaton** X is a five-tuple (A, S, $S_0$, $\delta$, F) where
A is an alphabet consisting of M ordered characters $a_0 < a_1 < \dots\ a_{M-1}$, ($1 \le M < \infty$),
S is an unempty set of states,
$S_0$ is a start state, $S_0 \in S$,
$\delta$ is a transition function $\delta$: $S \times A \rightarrow \boldsymbol{P}(S)$,
F is an unempty subset of S, it is a set of final states.
Symbol ***P***(S) denotes the power set of S, i.e. the set of all subsets of S including S itself and empty set.
Note that this is a definition of nondeterministic finite automaton.
We denote by symbol L(X) the **language accepted by automaton** X .

Lexicographical order of elements of $A^*$ is induced by the order of characters of A as follows:
For any two words w1, w2 $\in A^*$, w1 ≠ w2, we say that w1 is **lexicographically smaller** than w2 and denote this fact by w1 $<_{lex}$ w2 if either of the following holds
1. w1 is prefix of w2 and length(w1) < length(w2),
2. w1 and w2 are not empty words and w1 is not a prefix of w2 and a < b, where a, resp. b is the character in alphabet A immediately following the longest common prefix of w1 and w2 in the word w1, resp w2. Note that longest common prefix might be empty word in which case a, resp. b are the first characters of w1, resp. w2.

### The task

When language L(X) is not empty, we define the minimal word MINWL(X) $\in$ L(X) as follows
$\forall$ w $\in$ L(X): w ≠ MINWL(X) $\Rightarrow$ ( (length(MINWL(X) ≤ length(w)) and
( (length(MINWL(X)) = length(w)) $\Rightarrow$ ( MINWL(X) $<_{lex}$ w ) ) ).
When language L(X) is not empty and finite, we define the maximal word MAXWL(X)$\in$ L(X) as follows
$\forall$ w $\in$ L(X): w ≠ MAXWL(X) $\Rightarrow$ ( (length(MAXWL(X) ≥ length(w)) and
( (length(MAXWL(X)) = length(w)) $\Rightarrow$ ( w $<_{lex}$ MAXWL(X)) ) ).
When L(X) is empty, both MINWL(X) and MAXWL(X) are empty words.
We are given a nondeterministic finite automaton X. We have to decide if L(X) is finite. When L(X) is finite we have to find word MAXWL(X). When L(X) is not finite we have to find word MINWL(X).

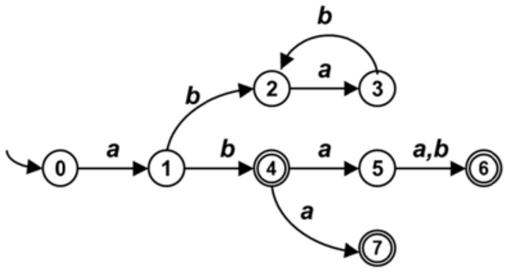
### Input

Let J be any finite set of k integers (k ≥0). We define **PAL_set_listing** of J to be a sequence of k+1 integers where k is the first element of the sequence followed by elements of J in arbitrary order.
Input specifies NFA X = (A, S, S0, $\delta$, F). We assume that S = {0, 1, 2. ..., N–1 } (N > 0), $S_0 = 0$. We also assume that A is a subset of {'a', 'b', .... 'z'}, A = {$a_0$, $a_1$. ..., $a_{M-1}$} = {'a', 'b', ...} , 1 ≤ M ≤ 26, 'a'= $a_0$ < 'b' = $a_1$ < ... .
There are two formats of input.
First line of input always contains three positive integers N, M, Q. N =|S|, M = |A|, Q $\in$ {1, 2}. Q specifies the format in which transition function $\delta$ is defined.
If Q = 1 then are exactly N following input lines which completely specify transition function $\delta$. Each line starts with state number $s_j$ and then contains M PAL_set_listings (defined above) of sets $\delta(s_j, a_0)$, $\delta(s_j, a_1)$, .... $\delta(s_j, a_{M-1})$ in this order.
If Q = 2 then the second input line contains six nonnegative integers B, C, T, U, V, W in this order separated by space.
Transition function $\delta$ is defined for state sj $\in$ S and character $a_h \in$ A as follows:
$\delta(s_j, a_h) = \{\ 1+\ s_j + ( (B\times s_j\times k+C\times h) \mod (N-\lfloor N/B\rfloor-1) ) \mid k \in \{1, 2, ..., H(s_j, h)\}\}$, for $s_j < \lfloor N/B\rfloor$,
$\delta(s_j, a_h) = \{\lfloor N/B\rfloor + ( (B\times s_j\times k+C\times h) \mod (N-\lfloor N/B\rfloor) ) \mid k \in \{1, 2, ..., H(s_j, h)\}\}$, for $s_j \ge \lfloor N/B\rfloor$,
where
$H(s_j, h) = T + ((U\times s_j+V\times h) \mod W)$, $2 \le B < N$, $W \ge 1$.
Note that $|\delta(s_j, a_h)| \le H(s_j, h)$, the two values are not necessarily equal.
Last line of input contains PAL_set_listing of F.
All values on any input line are separated by one or more spaces.
You may assume that N×M = |S|×|A| ≤ 150000.

**Output**
Output contains one text line. When input automaton X accepts finite language the output line first contains the string FINITE followed by one space followed by word MAXWL(X). When input automaton X accepts infinite language the output line first contains the string INFINITE followed by one space followed by word MINWL(X).
When MAXWL(X) or MINWL(X) is empty word it is substituted by the string EPSILON .

**Example 1**
Input

```
8 2 1
0  1 1   0
1  0     2 2 4
2  1 3   0
3  0     1 2
4  2 5 7   0
5  1 6   1 6
6  0     0
7  0     0
3  4 6 7
```



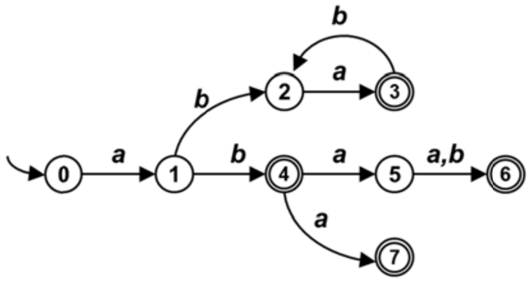

Output

```
FINITE abab
```

The transition diagram of the automaton is depicted to the right of input data. .

**Example 2**
Input

```
8 2 1
0  1 1   0
1  0     2 2 4
2  1 3   0
3  0     1 2
4  2 5 7   0
5  1 6   1 6
6  0     0
7  0     0
4  3 4 6 7
```



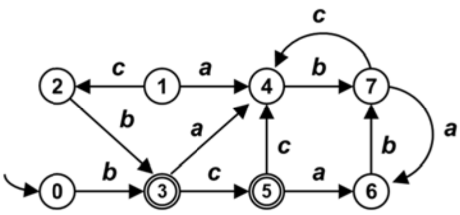

Output

```
INFINITE ab
```

The transition diagram of the automaton is depicted to the right of input data. .
Note that it differs from example 1 only in finality of state 3.

**Example 3**
Input

```
8 3 2
2 11  0 1 1 2
2 3 5
```




Output

```
FINITE bc
```

The transition diagram of the automaton is depicted to the right of input data. .

**Example 4**

**Input**

11 3 2

3 10  2 1 1 2

2 6 10

Output

**`INFINITE ca`**

The transition table of the input automaton is

| state | a | b | c | |
|---|---|---|---|---|
| 0 | [1] | [4] | [7] | |
| 1 | [4,5,8] | [4,8] | [3,4,7] | |
| 2 | [8,9] | [3,4,5] | [7,8] | |
| 3 | [4,5,6] | [6,7] | [8,9,10] | |
| 4 | [3,7] | [5,9] | [3,7] | |
| 5 | [8,9,10] | [3,4] | [4,5,6] | |
| 6 | [5,7] | [3,7,9] | [3,9] | F |
| 7 | [5,8,10] | [7,10] | [4,6,9] | |
| 8 | [3] | [5] | [7] | |
| 9 | [4,6,9] | [3,8] | [5,8,10] | |
| 10 | [7,9] | [3,7,9] | [3,5] | F |