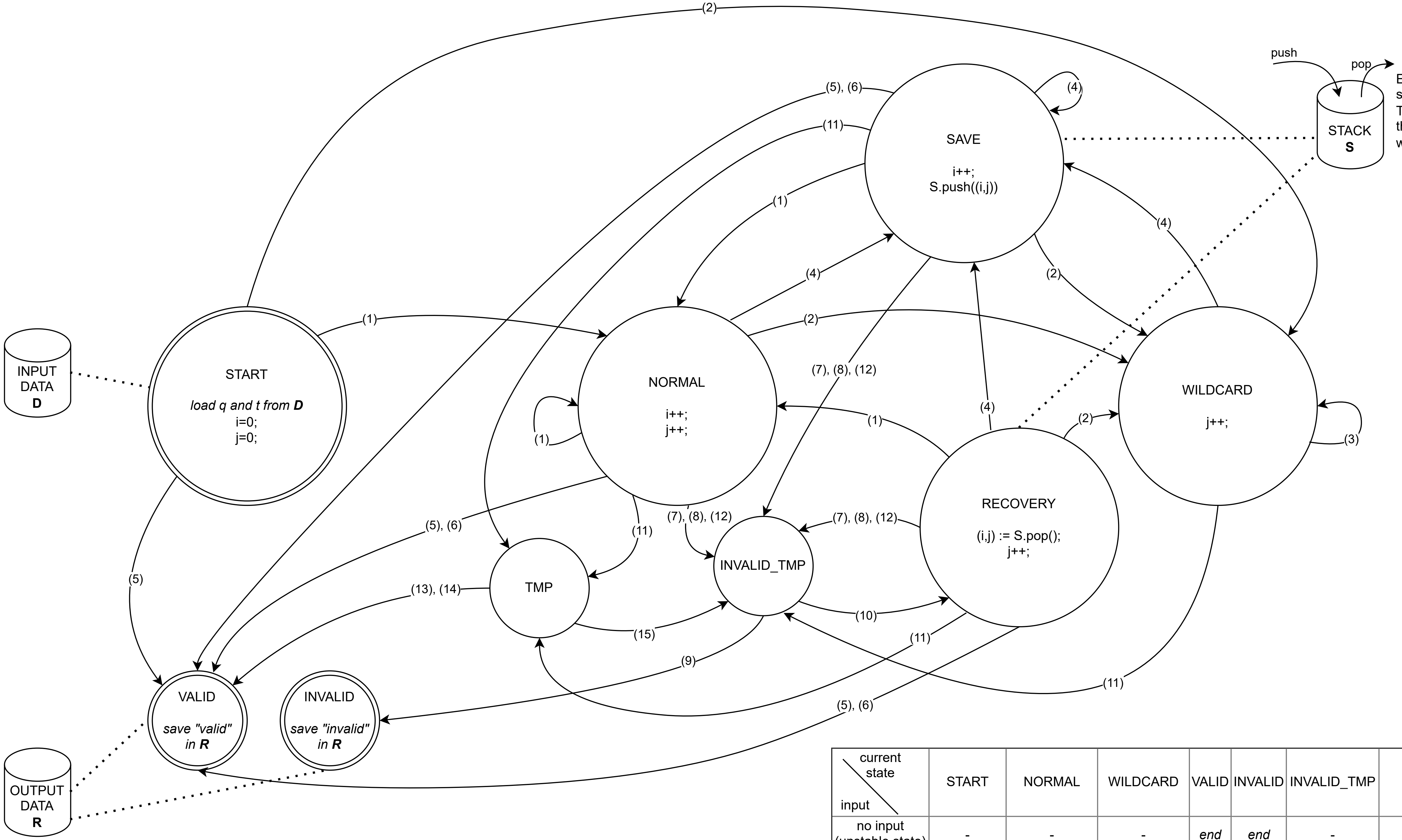
**GOAL**: Given an un-stememd wildcard query *q* and a stemmed token *t*, return true if the stemmed version of the wildcard query is compatible with the given stemmed token, false otherwise.
This problem is solved using an implementation of a modified finite-state machine, where each state can take some input parameter and its state evolves dynamically according to its current state and the input parameters.



Each time that a wildcard is found, the ***currentState*** is added to the stack.
The ***currentState*** is the tuple (*i, j*), where *j* is the index in *t* such that the condition *q[i+1]=t[j]* holds, i.e., the index in *t* of the character from which *t* restart matching *q* after a wildcard was encounterd in *q*.

(1)   i < q.length    AND  j < t.length  AND  q[i] = t[j]

(2)   i < q.length-1  AND  j < t.length  AND  q[i] ≠ t[j]      AND  q[i] = '*'

(3)   i < q.length-1  AND  j < t.length  AND  q[i+1] ≠ t[j]

(4)   i < q.length-1  AND  j < t.length  AND  q[i+1] = t[j]

(5)   i = q.length    AND  j = t.length

(6)   i = q.length-1  AND  j < t.length  AND  q[i] ≠ t[j]      AND  q[i] = '*'

(7)   i = q.length-1  AND  j < t.length  AND  q[i] ≠ t[j]      AND  q[i] ≠ '*'

(8)   i < q.length-1  AND  j < t.length  AND  q[i] ≠ t[j]      AND  q[i] ≠ '*'

(9)   S.isEmpty()

(10) ! S.isEmpty()

(11) i < q.length    AND   j = t.length

(12) i = q.length    AND   j < t.length

(13) j = t.length      AND   i < q.length  AND  *stem(* t + i.substring(i).removeAll("*") *)* = t

(14) q[i] = '*'          AND   i = q.length-1

(15) q[i] ≠ '*'         OR    i < q.length-1

| current state<br>input | START | NORMAL | WILDCARD | VALID | INVALID | INVALID_TMP | TMP | RECOVERY | SAVE |
|---|---|---|---|---|---|---|---|---|---|
| no input<br>(unstable state) | - | - | - | *end* | *end* | - | - | - | - |
| (1) | NORMAL | NORMAL | - | - | - | - | - | NORMAL | NORMAL |
| (2) | WILDCARD | WILDCARD | - | - | - | - | - | WILDCARD | WILDCARD |
| (3) | - | - | WILDCARD | - | - | - | - | - | - |
| (4) | - | SAVE | SAVE | - | - | - | - | SAVE | SAVE |
| (5) | VALID | VALID | - | - | - | - | - | VALID | VALID |
| (6) | - | VALID | - | - | - | - | - | VALID | VALID |
| (7) | - | INVALID_TMP | - | - | - | - | - | INVALID_TMP | INVALID_TMP |
| (8) | - | INVALID_TMP | - | - | - | - | - | INVALID_TMP | INVALID_TMP |
| (9) | - | - | - | - | - | INVALID | - | - | - |
| (10) | - | - | - | - | - | RECOVERY | - | - | - |
| (11) | - | TMP | INVALID_TMP | - | - | - | - | TMP | TMP |
| (12) | - | INVALID_TMP | - | - | - | - | - | INVALID_TMP | INVALID_TMP |
| (13) | - | - | - | - | - | - | VALID | - | - |
| (14) | - | - | - | - | - | - | VALID | - | - |
| (15) | - | - | - | - | - | - | INVALID_TMP | - | - |