```
PROGRAM = "(" ( PRED | INF ) ":-" ( PRED | NUM ) { "," ( PRED | NUM ) } ")" { "(" ( PRED | INF ) ":-"
       ( PRED | NUM ) { "," ( PRED | NUM ) } ")" } "(" PRED ( INF | PRED | NUM ) { "," ( INF | PRED |
      NUM ) } ":-" ( "(" [ "not" ] "(" INF ( PRED | NUM ) { "," ( PRED | NUM ) } ")" { ( "&" | "or" )
      [ "not" ] "(" INF ( PRED | NUM ) { "," ( PRED | NUM ) } ")" } ")" |  ( PRED | NUM ) { "," ( PRED
       | NUM ) } ) ")" { "(" PRED ( INF | PRED | NUM ) { "," ( INF | PRED | NUM ) } ":-" ( "(" [ "not"
       ] "(" INF ( PRED | NUM ) { "," ( PRED | NUM ) } ")" { ( "&" | "or" ) [ "not" ] "(" INF ( PRED |
       NUM ) { "," ( PRED | NUM ) } ")" } ")" |  ( PRED | NUM ) { "," ( PRED | NUM ) } ) ")" } "(" "?-
      " PRED ( PRED | NUM ) { "," ( PRED | NUM ) } ")".
```

WIRTH

```
initial: 0
final: 23
(0, "(") -> 1
(1, PRED) -> 2
(1, INF) -> 2
(2, ":-") -> 3
(3, PRED) -> 4
(3, NUM) -> 4
(4, ",") -> 3
(4, ")") -> 5
(5, "(") -> 6
(6, PRED) -> 7
(6, INF) -> 2
(7, PRED) -> 8
(7, INF) -> 8
(7, ":-") -> 3
(7, NUM) -> 8
(8, ":-") -> 9
(8, ",") -> 10
(9, "(") -> 11
(9, PRED) -> 12
(9, NUM) -> 12
(10, PRED) -> 8
(10, INF) -> 8
(10, NUM) -> 8
(11, "(") -> 16
(11, "not") -> 17
(12, ",") -> 13
(12, ")") -> 14
(13, PRED) -> 12
(13, NUM) -> 12
(14, "(") -> 15
(15, PRED) -> 10
(15, "?-") -> 18
(16, INF) -> 20
(17, "(") -> 16
(18, PRED) -> 19
(19, PRED) -> 21
(19, NUM) -> 21
(20, PRED) -> 22
(20, NUM) -> 22
(21, ",") -> 19
(21, ")") -> 23
(22, ",") -> 20
(22, ")") -> 24
(24, ")") -> 25
(24, "&") -> 11
(24, "or") -> 11
(25, ")") -> 14
```

AFD PROGRAM

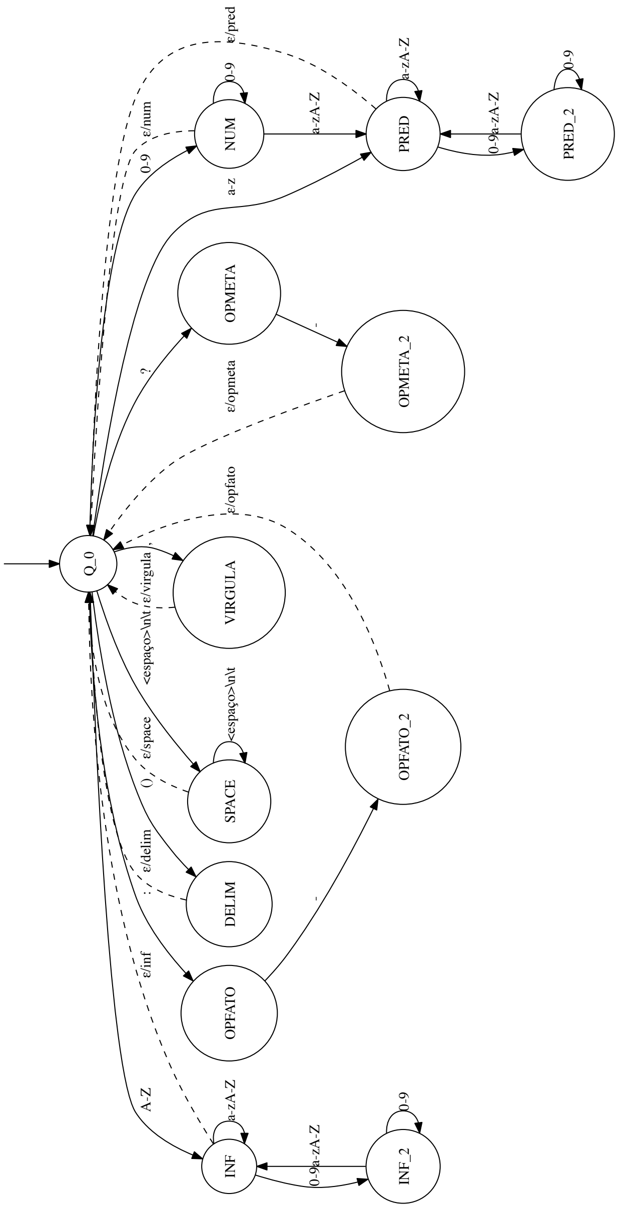

Figure 1: Transdutor Léxico

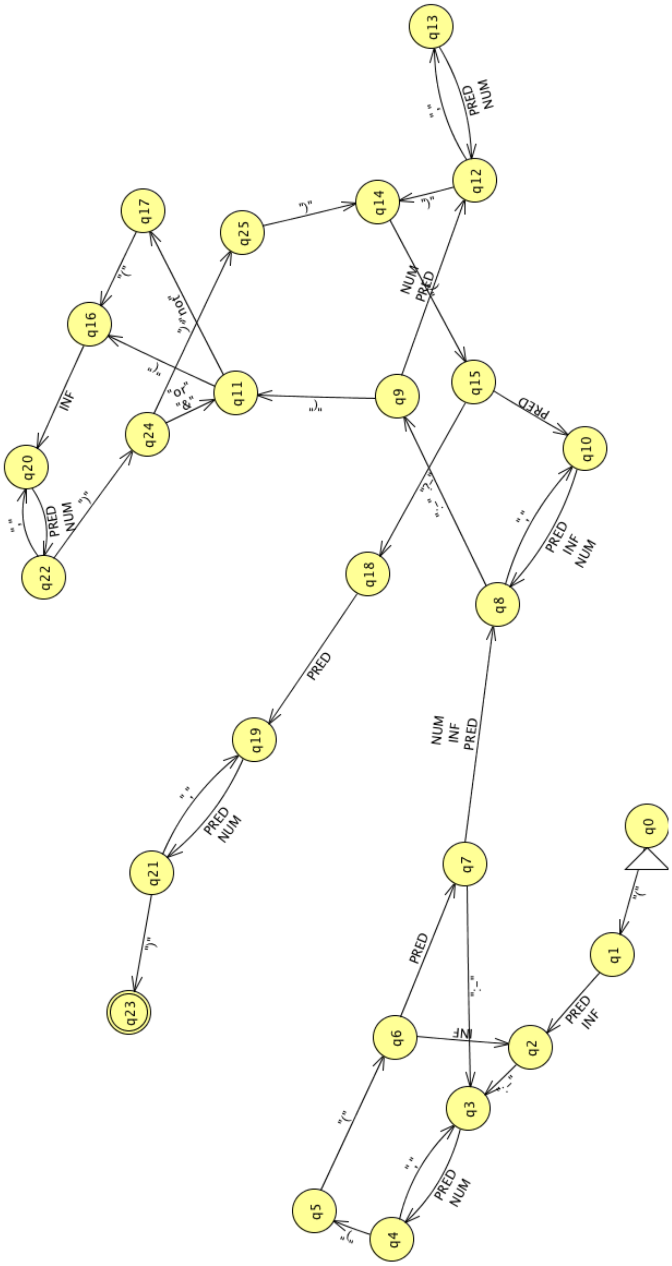

Figure 2: Autômato PROGRAM