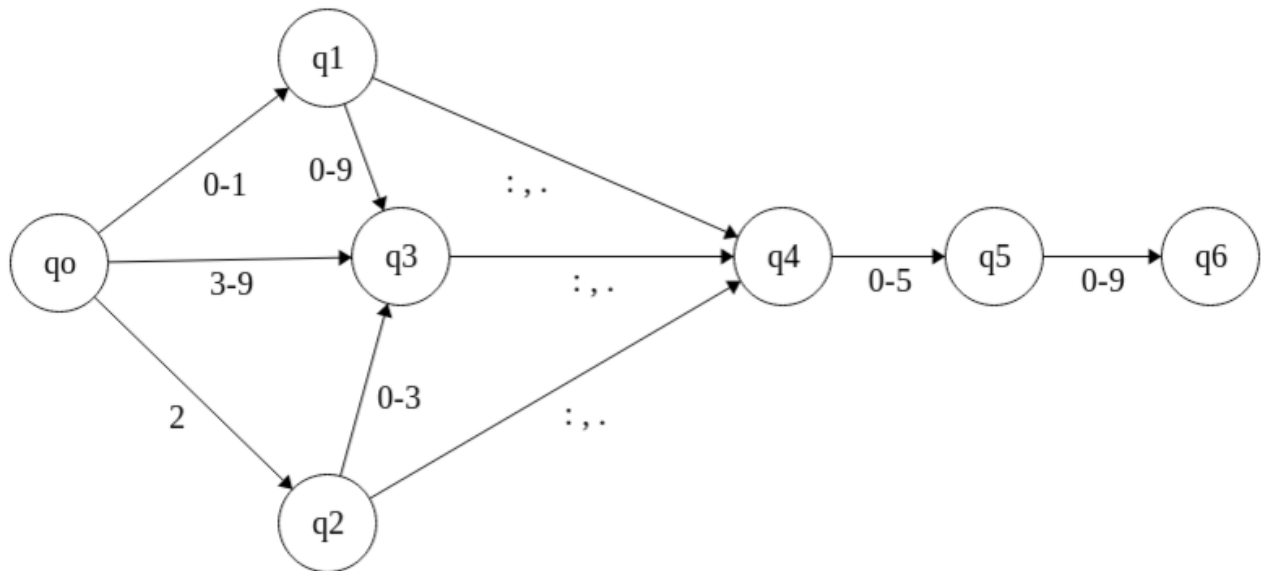


- Σχήμα του ντετερμινιστικού αυτομάτου πεπερασμένων καταστάσεων .



- **Περιγραφή του πίνακα μεταβάσεων.**

Ο πίνακας μεταβάσεων είναι ένα λεξικό το οποίο περιέχει όλες τις δυνατές καταστάσεις του αυτομάτου μας. Για να περάσει λοιπόν από την κατάσταση q_0 στην κατάσταση q_1 πρέπει να έχουμε είτε το 1 είτε το 0. Αν έχουμε το 2 μεταφερόμαστε στο q_2 ενώ για αριθμούς από το 3 έως το 9 πηγαίνουμε στο q_3 . Αν από το q_1 έχουμε αριθμό από το 0 έως το 9 πάμε στην κατάσταση q_3 ενώ αν έχουμε τα σύμβολα $'.'$ ή $':'$ πηγαίνουμε στην κατάσταση q_4 . Από την κατάσταση q_2 αν έχουμε αριθμό από το 0 έως το 3 πηγαίνουμε στην κατάσταση q_3 ενώ αν έχουμε τα σύμβολα $'.'$ ή $':'$ πηγαίνουμε στην κατάσταση q_4 . Από την κατάσταση q_4 για να συνεχίσει στην κατάσταση q_5 πρέπει να έχουμε αριθμό από το 0 έως το 5. Εξίσου και για την κατάσταση q_5 για να πάμε στην κατάσταση q_6 θέλουμε αριθμό από 0 έως 9.

- **Συνοπτική περιγραφή του κώδικά σας.**

Ο κώδικας αποτελείται από 2 συναρτήσεις την `getchar()` και την `scan()` καθώς και 2 λεξικά, τον πίνακα μεταβάσεων και τον πίνακα αποδεκτών καταστάσεων.

- *Αποτελέσματα εξόδου για έγκυρες και άκυρες μορφές ώρας.*

```

test.py (~/Desktop/sxolh/compilers 2k18) - gedit
# current state is not accepting
return 'ERROR_TOKEN',pos

# the transition table, as a dictionary
td = {
    'q0': {
        '0': 'q1', '1': 'q1', '2': 'q2', '3': 'q3', '4': 'q3', '5': 'q3', '6': 'q3', '7': 'q3',
        'q1': {
            '0': 'q3', '1': 'q3', '2': 'q3', '3': 'q3', '4': 'q3', '5': 'q3', '6': 'q3', '7': 'q3',
            'q2': {
                '0': 'q3', '1': 'q3', '2': 'q3', '3': 'q3', '4': 'q4', '5': 'q4',
                'q3': {
                    '0': 'q4', '1': 'q4',
                    'q4': {
                        '0': 'q5', '1': 'q5', '2': 'q5', '3': 'q5', '4': 'q5', '5': 'q5',
                    }
                }
            }
        }
    }

# the dictionary of accepting states and their
# corresponding token
ad = {
    'q6': 'NUM_TOKEN'

# get a string from input
text = input('give some input>')

# scan text until no more input
while text:
    # that is, while len(text)>0

    # get next token and position after last char recognized
    token, position = scan(text, td, ad)

    if token == 'ERROR_TOKEN':
        print('unrecognized token input at pos', position
+1, 'of', text)

```

```

afe@afe-Inspiron-5537: ~/Desktop/sxolh/compilers 2k18$ python3 test.py
give some input>12.00
token: NUM_TOKEN string: 12.00
afe@afe-Inspiron-5537:~/Desktop/sxolh/compilers 2k18$ python3 test.py
give some input>25:00
unrecognized token input at pos 2 of 25:00
afe@afe-Inspiron-5537:~/Desktop/sxolh/compilers 2k18$ python3 test.py
give some input>kalhspira
unrecognized token input at pos 1 of kalhspira
afe@afe-Inspiron-5537:~/Desktop/sxolh/compilers 2k18$ python3 test.py
give some input>1.45
token: NUM_TOKEN string: 1.45
afe@afe-Inspiron-5537:~/Desktop/sxolh/compilers 2k18$ python3 test.py
give some input>00:00
token: NUM_TOKEN string: 00:00
afe@afe-Inspiron-5537:~/Desktop/sxolh/compilers 2k18$ python3 test.py
give some input>12345
unrecognized token input at pos 3 of 12345
afe@afe-Inspiron-5537:~/Desktop/sxolh/compilers 2k18$ python3 test.py
give some input>:12
unrecognized token input at pos 1 of :12
afe@afe-Inspiron-5537:~/Desktop/sxolh/compilers 2k18$

```

- *Αναφορά σε πηγές που πιθανόν χρησιμοποιήσατε.*

Για την δημιουργία του DFA: <http://madebyevan.com/fsm/>