

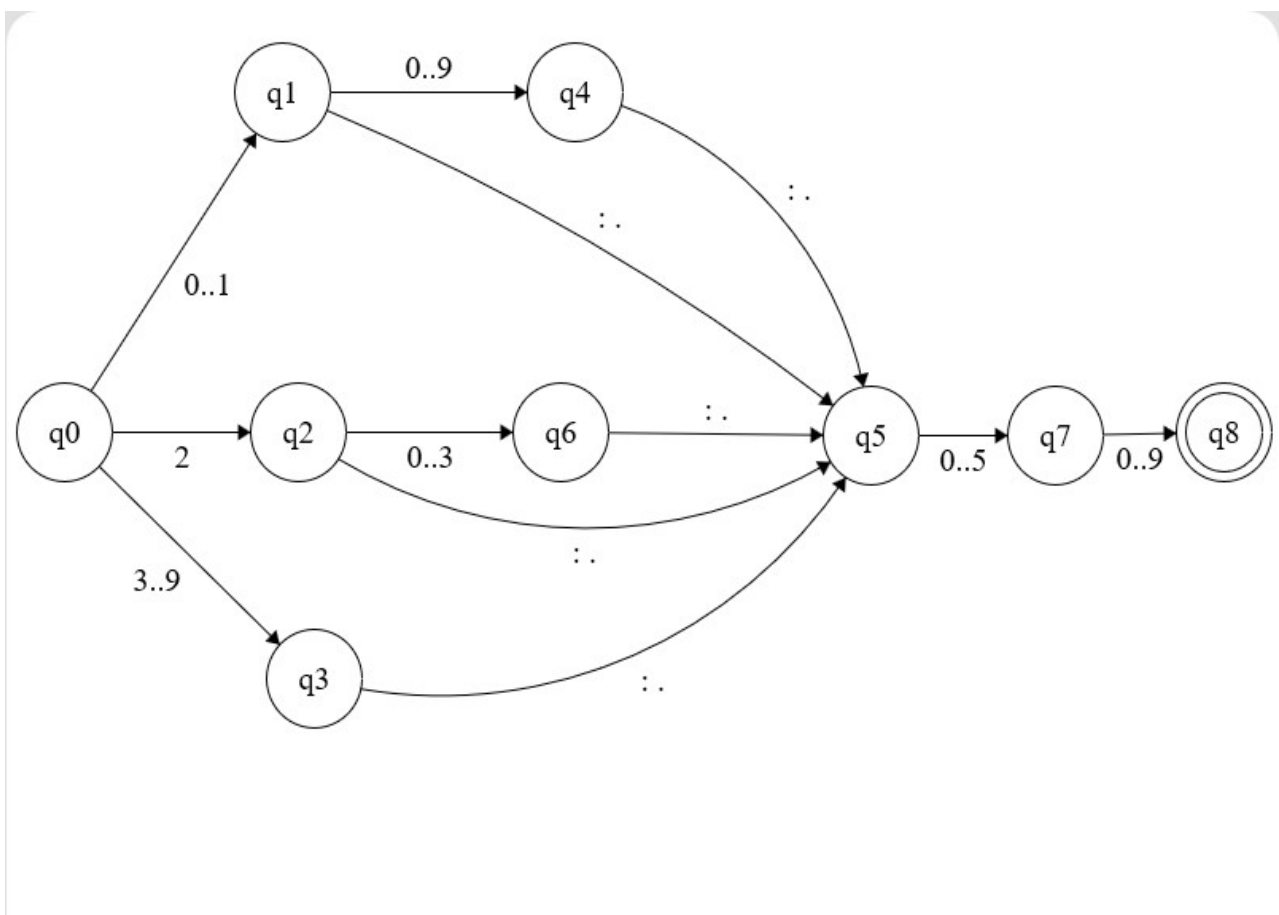
Μεταγλωττιστές 2018
Προγραμματιστική Εργασία #1

Ονοματεπώνυμο: Μαγουνάκη Ουρανία

A.M: Π2015140

Στην παρούσα εργασία κατασκευάστηκε λεκτικός αναλυτής ο οποίος αναγνωρίζει ενδείξεις ώρας σε **24ωρη** μορφή με χρήση της **Python3**.

Αρχικά, παρουσιάζεται το σχήμα του ντετερμινιστικού αυτομάτου πεπερασμένων καταστάσεων (**DFA**) που αντιστοιχεί στην ζητούμενη προδιαγραφή.



Ο πίνακας μεταβάσεων απεικονίζεται παρακάτω:

| δ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | : | . |
|----------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| q0 | {q1} | {q1} | {q2} | {q3} | {q3} | {q3} | {q3} | {q3} | {q3} | {q3} | \emptyset | \emptyset |
| q1 | {q4} | {q4} | {q4} | {q4} | {q4} | {q4} | {q4} | {q4} | {q4} | {q4} | {q5} | {q5} |
| q2 | {q6} | {q6} | {q6} | {q6} | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | {q5} | {q5} |
| q3 | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | {q5} | {q5} |
| q4 | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | {q5} | {q5} |
| q5 | {q7} | {q7} | {q7} | {q7} | {q7} | {q7} | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset |
| q6 | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | \emptyset | {q5} | {q5} |
| q7 | {q8} | {q8} | {q8} | {q8} | {q8} | {q8} | {q8} | {q8} | {q8} | {q8} | \emptyset | \emptyset |

Στον πίνακα μεταβάσεων η πρώτη στήλη έχει ως στοιχεία τις καταστάσεις του αυτομάτου, η πρώτη γραμμή έχει ως στοιχεία τους αριθμούς, την άνω κάτω τελεία (:) και την τελεία (.) ενώ τα υπόλοιπα κελιά έχουν ως στοιχείο την κατάσταση στην οποία μεταβαίνει το αυτόματο με είσοδο την κατάσταση που βρίσκεται στην ίδια γραμμή με αυτή του κελιού και το σύμβολο που βρίσκεται στην ίδια στήλη με αυτή του κελιού.

Το σύμβολο \emptyset σημαίνει ότι δεν υπάρχει μετάβαση.

Διευκρινίζεται ότι :

δ : συνάρτηση μετάβασης

q0: αρχική κατάσταση

F={q8}: τελική κατάσταση

Αρχικά, δημιουργήθηκε το λεξικό δ το οποίο περιέχει τις καταστάσεις μετάβασης, τους αριθμούς από το 0 έως 9 καθώς και τους χαρακτήρες (:) και (.) για την μετάβαση από την μία κατάσταση στην άλλη.

Η αρχική κατάσταση q0 του λεξικού διαιρείται σε 3 καταστάσεις, η πρώτη κατάσταση q1 περιλαμβάνει τα ψηφία 0 και 1 ενώ ακολουθούν είτε οι αριθμοί από 0 έως 9 που οδηγούν στην κατάσταση q4 είτε οι χαρακτήρες (.) ή (:) οι οποίοι οδηγούν στην κατάσταση q5.

Η δεύτερη μετάβαση στην q2 από την αρχική κατάσταση περιλαμβάνει μόνο το 2 καθώς ο συγκεκριμένος αριθμός ακολουθείται είτε από τους αριθμούς 0 μέχρι 3 (κατάσταση q6) είτε από τους χαρακτήρες (.) και (:) (κατάσταση q5) προκειμένου να είναι έγκυρη η ώρα. Η κατάσταση q6 μετά την εισαγωγή (:) ή (.) οδηγεί στην κατάσταση q5.

Επιπλέον, για την τρίτη μετάβαση στην κατάσταση q3 χρησιμοποιούνται οι αριθμοί από το 3 έως το 9 ενώ ακολουθείται έπειτα είτε (:) είτε (.) που οδηγούν στην κατάσταση q5. Η μετάβαση από την q5 στην q7 περιλαμβάνει τους αριθμούς από 0 έως 5 ενώ καταληκτικά η μετάβαση από την q7 στην q8 περιλαμβάνει τους αριθμούς από το 0 έως 9 αφού τα λεπτά είναι από 0 μέχρι 59.

Στην συνέχεια δημιουργήθηκε το λεξικό καταστάσεων αποδοχής στο οποίο περιλαμβάνεται η τελική κατάσταση q8. Όταν η ώρα που έχει πληκτρολογήσει ο χρήστης είναι αποδεκτή δηλαδή βρίσκεται στην κατάσταση αποδοχής τότε εκτυπώνεται “TIME_TOKEN”.

Η *getchar()* δέχεται ως είσοδο τους χαρακτήρες από τον χρήστη καθώς και την θέση και επιστρέφει τον χαρακτήρα στην συγκεκριμένη θέση .Σε περίπτωση που η θέση αυτή είναι μικρότερη του μηδενός ή ξεπερνάει τα όρια της λέξης τότε δεν επιστρέφει τίποτα.

Η *scan()* παίρνει ως είσοδο το κείμενο, τον πίνακα μεταβάσεων και τις καταστάσεις αποδοχής.

Αν η κατάσταση είναι κατάσταση αποδοχής τότε επιστρέφει “**TIME_TOKEN**” αλλιώς επιστρέφει “**ERROR_TOKEN**” και την θέση του μη αποδεκτού χαρακτήρα. Εν κατακλείδι, η συνάρτηση *input()* παίρνει ως είσοδο από τον χρήστη την ώρα.

Παρακάτω παρουσιάζονται ορισμένα αποτελέσματα μετά την εισαγωγή έγκυρων και μη μορφών ώρας:

```
File Edit View Terminal Tabs Help
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>1.00
token: TIME_TOKEN string: 1.00
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>2.00
token: TIME_TOKEN string: 2.00
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>23:59
token: TIME_TOKEN string: 23:59
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>23:45
token: TIME_TOKEN string: 23:45
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>9.00
token: TIME_TOKEN string: 9.00
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>8:25
token: TIME_TOKEN string: 8:25
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>98:00
ERROR_TOKEN
unrecognized input at pos 2 of 98:00
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>23:90
ERROR_TOKEN
unrecognized input at pos 4 of 23:90
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>iu:09
ERROR_TOKEN
unrecognized input at pos 1 of iu:09
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>23jfs
ERROR_TOKEN
unrecognized input at pos 3 of 23jfs
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>09:00
token: TIME_TOKEN string: 09:00
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>02:00
token: TIME_TOKEN string: 02:00
guestuser@lab3n12:~/Desktop$ python3 scanner.py
give some input>03.00
token: TIME_TOKEN string: 03.00
guestuser@lab3n12:~/Desktop$
```

```
File Edit Search View Document Help
td={
    'q0':{'0':'q1','1':'q1','2':'q2','3':'q3','4':'q3','5':'q3','6':'q3','7':'q3','8':'q3','9'
    'q1':{'0':'q4','1':'q4','2':'q4','3':'q4','4':'q4','5':'q4','6':'q4','7':'q4','8':'q4','9'
    'q2':{'0':'q6','1':'q6','2':'q6','3':'q6','4':'q5','5':'q5'},
    'q3':{'':'q5','':'q5'},
    'q4':{'':'q5','':'q5'},
    'q6':{'':'q5','':'q5'},
    'q5':{'0':'q7','1':'q7','2':'q7','3':'q7','4':'q7','5':'q7'},
    'q7':{'0':'q8','1':'q8','2':'q8','3':'q8','4':'q8','5':'q8','6':'q8','7':'q8','8':'q8','9'
    }
}
ad = {
    'q0':'TIME_TOKEN',
    }

def getchar(words,pos):
    """ returns char at pos of words, or None if out of bounds """

    if pos<0 or pos>=len(words):
        return None

    return words[pos]

def scan(text,transition_table,accept_states):
    """ Scans `text` while transitions exist in `transition_table`.
    After that, if in a state belonging to `accept_states`,
    returns the corresponding token, else ERROR_TOKEN.
    """
```