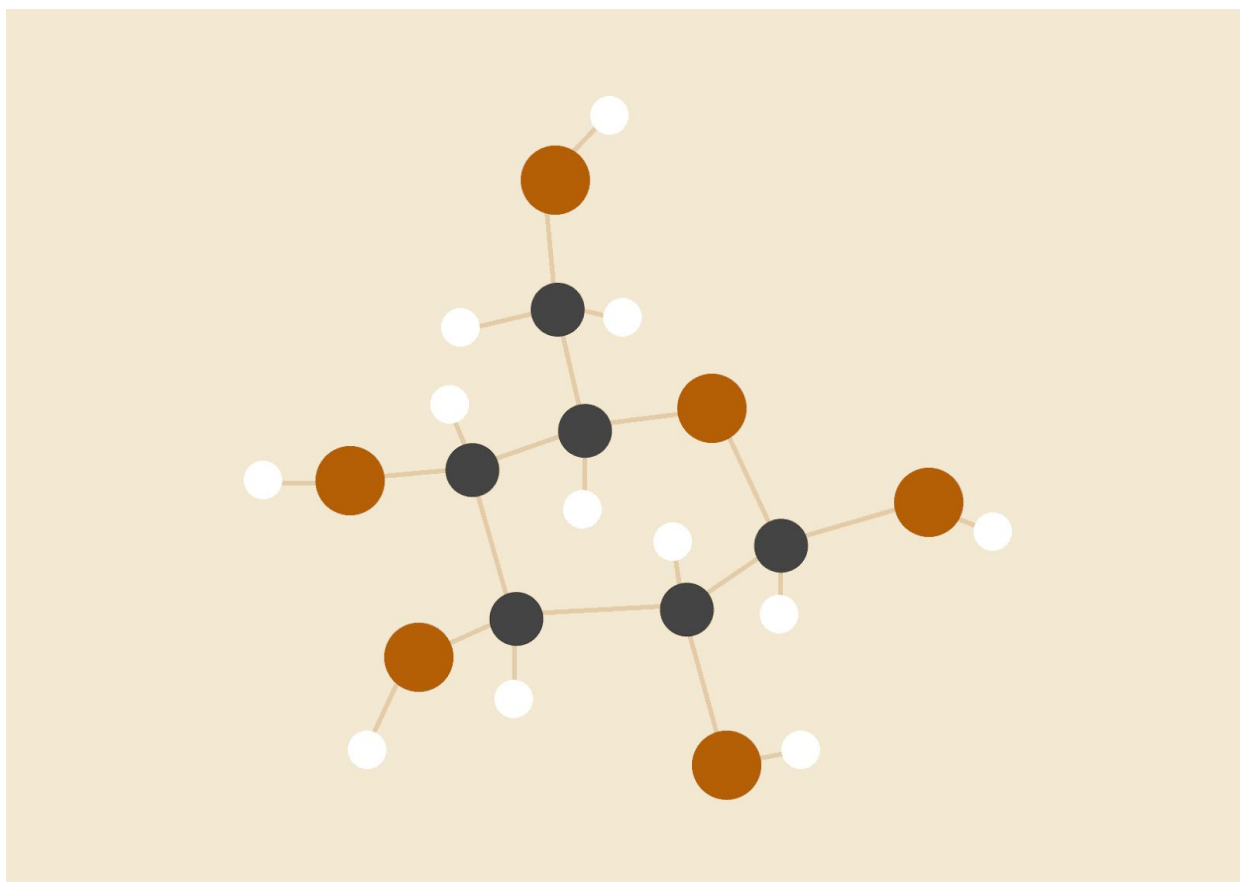


Μεταγλωττιστές 2019

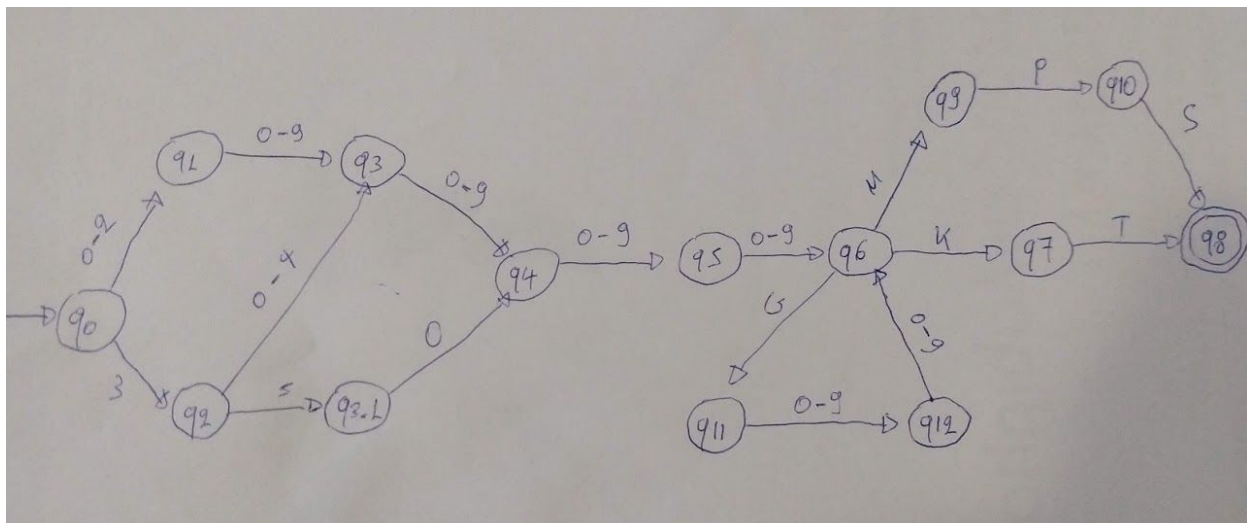
Προγραμματιστική Εργασία #1



Σπύρος Μπαξεβανάκης

Π2016086

ΑΥΤΟΜΑΤΟ



Τελική κατάσταση αποδοχής η q8.

ΠΙΝΑΚΑΣ ΜΕΤΑΒΑΣΕΩΝ

```
transitions = {
    's0': { 'D0': 's1', 'D02': 's1', 'D3': 's2' },
    's1': { 'D0': 's3', 'D02': 's3', 'D3': 's3', 'D04': 's3', 'D5': 's3', 'D09': 's3' },
    's2': { 'D0': 's3', 'D02': 's3', 'D3': 's3', 'D04': 's3', 'D5': 's3.1' },
    's3': { 'D0': 's4', 'D02': 's4', 'D3': 's4', 'D04': 's4', 'D5': 's4', 'D09': 's4' },
    's3.1': { 'D0': 's4' },
    's4': { 'D0': 's5', 'D02': 's5', 'D3': 's5', 'D04': 's5', 'D5': 's5', 'D09': 's5' },
    's5': { 'D0': 's6', 'D02': 's6', 'D3': 's6', 'D04': 's6', 'D5': 's6', 'D09': 's6' },
    's6': { 'M': 's9', 'G': 's11', 'K': 's7' },
    's9': { 'P': 's10' },
    's10': { 'S': 's8f' },
    's7': { 'T': 's8f' },
    's11': { 'D0': 's12', 'D02': 's12', 'D3': 's12', 'D04': 's12', 'D5': 's12', 'D09': 's12' },
    's12': { 'D0': 's6', 'D02': 's6', 'D3': 's6', 'D04': 's6', 'D5': 's6', 'D09': 's6' }
}
```

Τα διαφορετικά inputs έχουν χωριστεί στις εξής κατηγορίες: D0,D02,D3,D04,D5,D09. Όπου DXY είναι ψηφίο που ανήκει στο [X,Y] και DZ είναι το ψηφίο Z.

Πχ. D04 -> 0,1,2,3,4

Το κύριο μειονέκτημα αυτής της προσέγγισης είναι ότι

οι κατηγορίες επικαλύπτονται και έτσι είναι αδύνατο να τις ξεχωρίσουμε.

Πχ. Έστω το `c=4` και `state=s4`, τότε η `getchar` θα επιστρέψει `D04` ενώ το αυτόματο περιμένει `D09`.

Για να αντιμετωπιστεί αυτό το πρόβλημα, πρέπει στο πίνακα με τα βάσεων να αναφέρουμε και όλες τις επικαλυπτόμενες κατηγορίες.

Πχ. Για να εκφράσουμε την μετάβαση `s2->s3` πρέπει να γράψουμε `{D0,D02,D3,D04}->s3` ενώ στο αυτόματο αναγράφεται `D04`.

Αυτό το πρόβλημα γίνεται πιο αντιληπτό στο `D09`.

Επίσης πρέπει να σημειωθεί ότι η κατάσταση `s3.1` προέκυψε από σχεδιαστικό λάθος κατά την δημιουργία του αυτόματου.

ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΚΩΔΙΚΑ

Η μόνη αλλαγή που έγινε στον κώδικα είναι στα σημεία αναφοράς με την προσθήκη της αλλαγής από 2 tab με μέγεθος 4 σε 1 tab που αντιστοιχεί σε 4 spaces.

ΑΠΟΤΕΛΕΣΜΑΤΑ

1. **02015KT**
give some input>02015KT
token: WIND_TOKEN text: 02015KT
2. **33020KT**
give some input>33020KT
token: WIND_TOKEN text: 33020KT
3. **19020G26KT**
give some input>19020G26KT

token: WIND_TOKEN text: 19020G26KT

4. **19020G26MPSs**

give some input>19020G26MPSs

token: WIND_TOKEN text: 19020G26MPS

unrecognized input at position 0 of s

5. **35120KT**

give some input>35120KT

unrecognized input at position 2 of 35120KT