

Παρουσίαση εργαλείου FSM

& Μέρος A-2 της εργασίας εξαμήνου

Αναθέσεις ομάδων

- ▶ Μετά το πέρας του σημερινού μαθήματος θεωρούνται οριστικές (υπό Κ.Σ. !!!)
- ▶ Αν υπάρχει οποιοδήποτε πρόβλημα το αναφέρετε στο τέλος του σημερινού μαθήματος!
- ▶ Ο αύξων αριθμός στη λίστα του Wiki είναι ο αριθμός της ομάδας σας

Εισαγωγή στο εργαλείο (1)

- ▶ Το πρόγραμμα FSM (τα αρχικά από τη φράση **finite state machine** – μηχανή πεπερασμένων καταστάσεων) είναι ένας μεταφραστής που:
 - Δέχεται την περιγραφή της λειτουργίας ενός **διαγράμματος μετάβασης** (σε μια συγκεκριμένη γλώσσα κατάλληλη για δήλωση μεταβάσεων)
 - Επιβεβαιώνει ότι το **αυτόματο** είναι πράγματι **ντετερμινιστικό**
 - Προσομοιώνει τη λειτουργία του επάνω σε δεδομένα εισόδου που προσδιορίζονται από τον χρήστη
 - Απαντάει αν τα δεδομένα αναγνωρίζονται από το αυτόματο ή όχι

Εισαγωγή στο εργαλείο (2)

- ▶ Το εργαλείο FSM είναι ένα πρόγραμμα γραμμένο σε γλώσσα C
 - Το περιβάλλον λειτουργίας (Windows / Linux) είναι πρακτικά αδιάφορο
- ▶ Ο πηγαίος κώδικας είναι διαθέσιμος στο eClass: FSM source στον κατάλογο [3-ΕΡΓΑΣΤΗΡΙΟ](#) » [0-Υποστήριξη για Εργαλεία](#) » [0_1-ΚΕ και FSM](#)

Εισαγωγή στο εργαλείο (3)

- ▶ Το εκτελέσιμο σε Linux παράγεται με
 - `gcc -o fsm fsm.c`
- ▶ Σε Windows το εκτελέσιμο (exe) είναι διαθέσιμο ως FSM.zip στον κατάλογο 3-ΕΡΓΑΣΤΗΡΙΟ » 0-Υποστήριξη για Εργαλεία » 0_1-ΚΕ και FSM

Χρήση εργαλείου FSM

- ▶ Θα πρέπει να δημιουργηθεί ένα κατάλληλο αρχείο εισόδου με την περιγραφή ενός αυτόματου/ΔΜ
- ▶ Το αρχείο αυτό θα πρέπει να δοθεί ως παράμετρος στο εκτελέσιμο
- ▶ Πχ. Έστω αρχείο εισόδου test.fsm
 - Εκτέλεση σε Windows
 - fsm.exe test.fsm
 - Εκτέλεση σε Linux
 - ./fsm test.fsm

Προσοχή: Η εκτέλεση θα πρέπει να γίνει στον κατάλογο που περιέχει το εκτελέσιμο αρχείο!!!

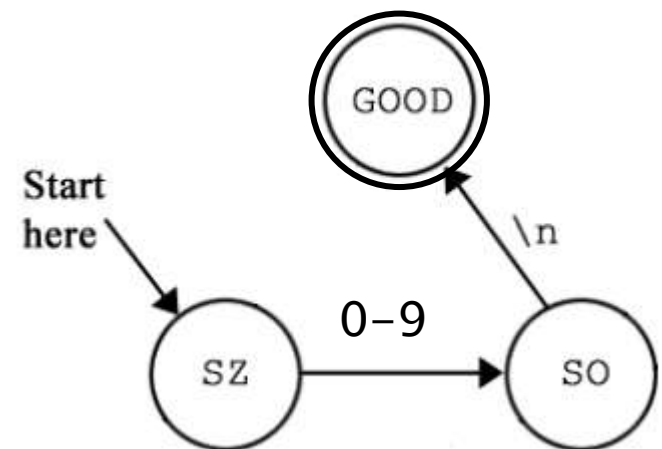
Εγχειρίδιο χρήσης FSM

- ▶ Περισσότερες πληροφορίες για το FSM και τη λειτουργία του μπορείτε να βρείτε στο εγχειρίδιο χρήσης του στο eClass:
 - [3-ΕΡΓΑΣΤΗΡΙΟ](#) » [0-Υποστήριξη για Εργαλεία](#) » [0_1-ΚΕ και FSM](#)
 - FSM manual

Δοκιμή του εργαλείου FSM (1 α)

- ▶ Ένα απλό αυτόματο που να δέχεται ένα μόνο αριθμητικό ψηφίο (0-9) και μετά τον χαρακτήρα νέας γραμμής (\n)
- ▶ Τίποτα άλλο!

???

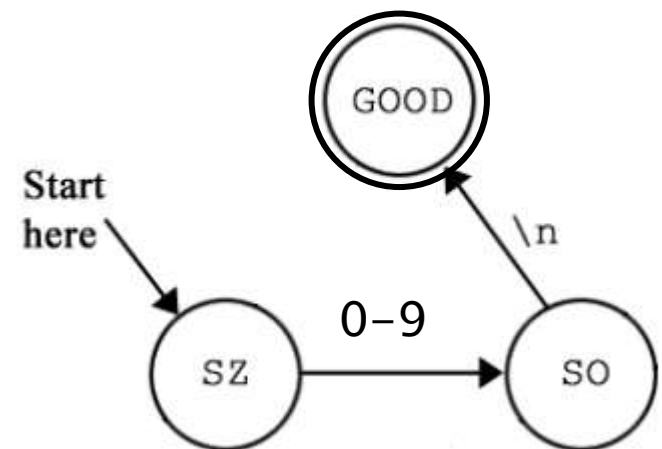


Δοκιμή του εργαλείου FSM (1 β)

- ▶ Ένα απλό αυτόματο που να δέχεται ένα μόνο αριθμητικό ψηφίο (0-9) και μετά τον χαρακτήρα νέας γραμμής (\n)
- ▶ Τίποτα άλλο!

START=SZ
SZ: 0-9 -> SO
SO: \n -> GOOD
GOOD(OK):

Προσοχή: Μετά το GOOD(OK): οπωσδήποτε ακολουθεί νέα γραμμή!!!



Δοκιμή του εργαλείου FSM (1 γ)

► fsm test.fsm

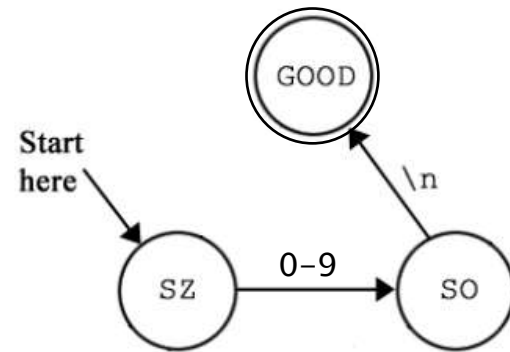
- 0 -> Πατάμε Enter -> Πατάμε Ctrl+Z -> Ξανά Enter
 - YES

'Η Ctrl+D σε Linux

► Ομοίως:

- 9
- 85
- a

???



START=SZ

SZ: 0-9 -> SO

SO: \\n -> GOOD

GOOD(OK):

Δοκιμή του εργαλείου FSM (1δ)

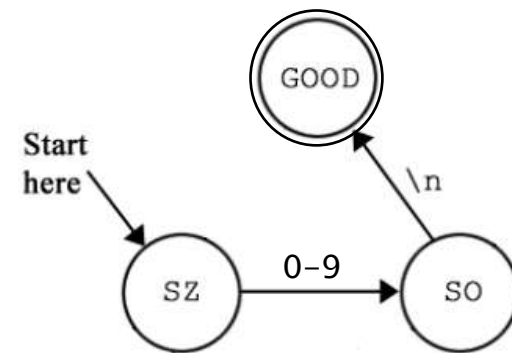
▶ fsm test.fsm

- 0 -> Πατάμε Enter -> Πατάμε Ctrl+Z -> Ξανά Enter
 - YES

'H Ctrl+D σε Linux

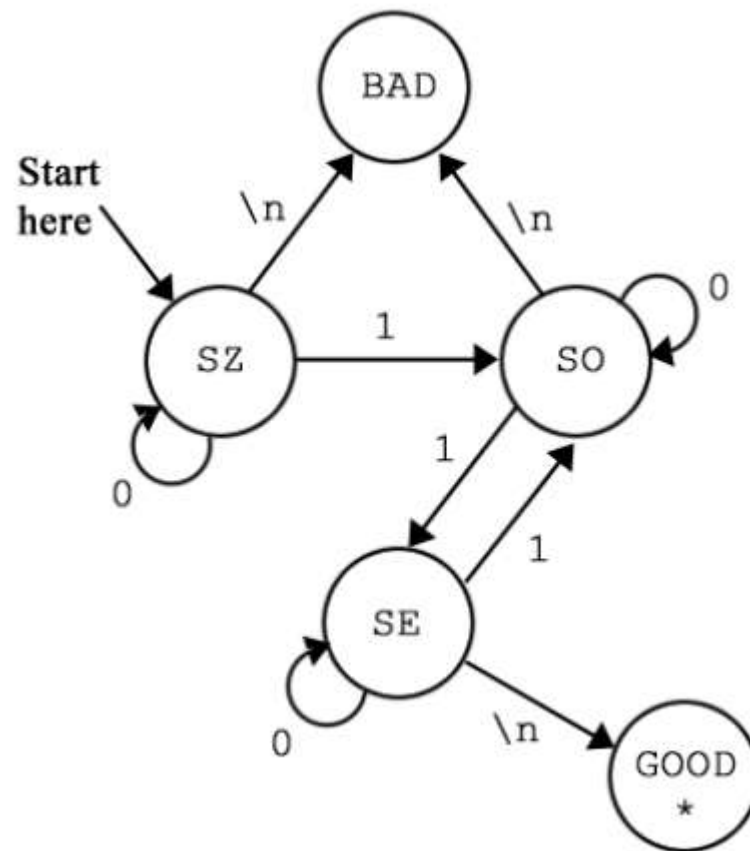
▶ Ομοίως:

- 9
 - YES
- 85
 - fsm: in test.fsm, state 'so' input 5 not accepted
- a
 - fsm: in test.fsm, state 'sz' input a not accepted



Παράδειγμα αρχείου εισόδου (2α)

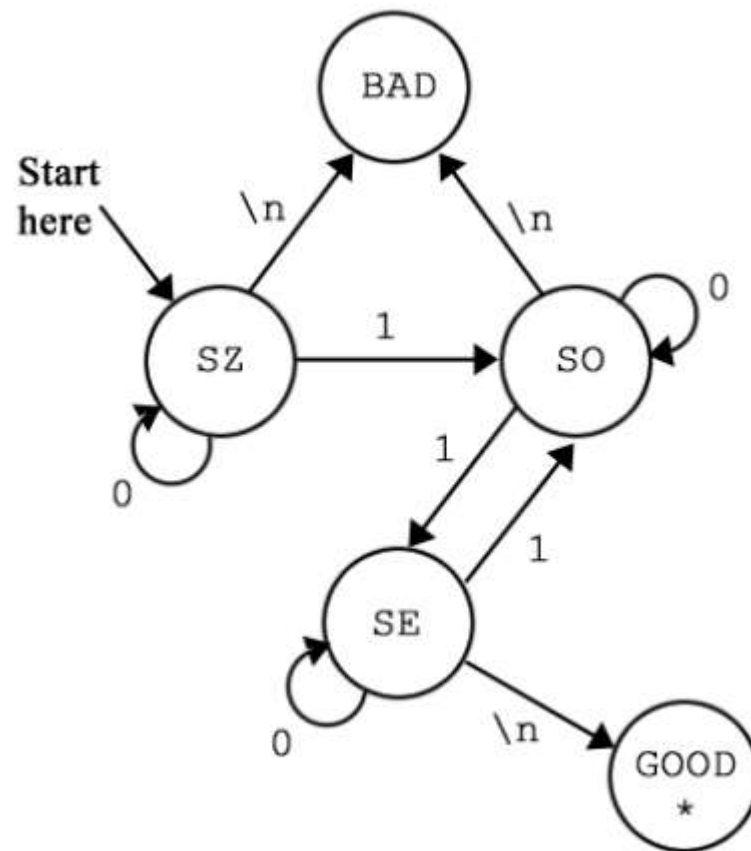
- ▶ Έστω το παρακάτω ΔΜ:



Παράδειγμα αρχείου εισόδου (2β)

- ▶ Κώδικας που περιγράφει το ΔΜ:

???



Παράδειγμα αρχείου εισόδου (2γ)

- Κώδικας που περιγράφει το ΔΜ:

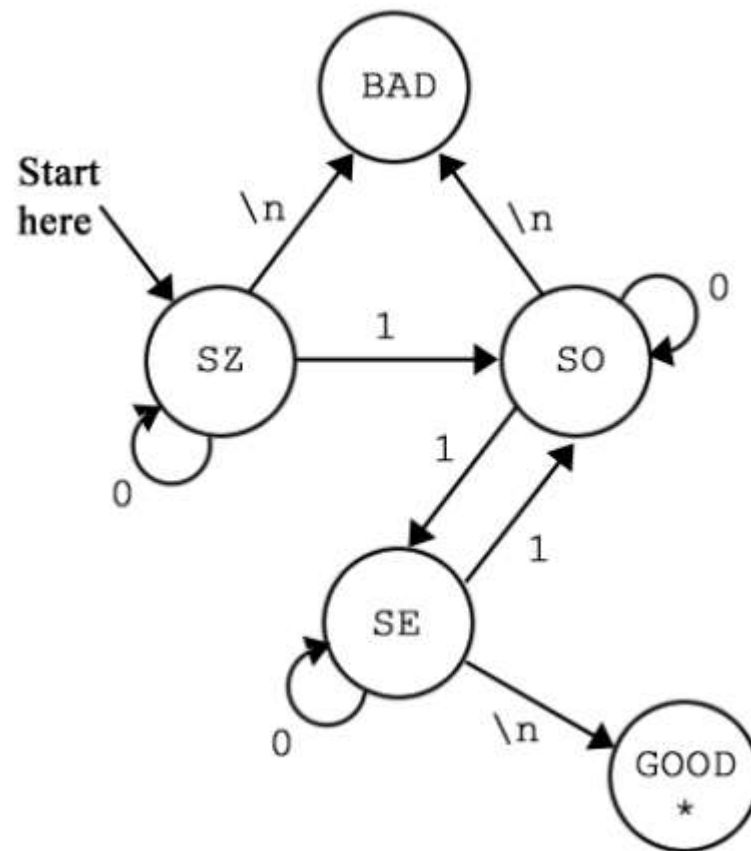
START=SZ

SZ: 0 -> SZ
 1 -> SO
 \n -> BAD

SO: 0 -> SO
 1 -> SE
 \n -> BAD

SE: 0 -> SE
 1 -> SO
 \n -> GOOD

GOOD(OK):



Εκτέλεση εργαλείου FSM (2δ)

- ▶ Αποθηκεύστε τον κώδικα ως απλό αρχείο κειμένου με όνομα **automato0-1.fsm**
- ▶ Εκτελέστε την εντολή
 - `fsm automato0-1.fsm` [Windows]
 - `./fsm automato0-1.fsm` [Linux]
- ▶ Το FSM θα επιβεβαιώσει ότι το αυτόματο/ΔΜ που περιγράφεται είναι συντακτικά σωστό και ντετερμινιστικό
- ▶ Ακολούθως θα περιμένει μια λέξη για να μας πει αν είναι αποδεκτή ή όχι. Πχ:
 - 1111 → YES
 - 10 → NO

Εκτέλεση εργαλείου FSM (2ε)

- ▶ Όταν τελειώσετε με το input, πατήστε Enter
 - Αν είστε σε Windows πατήστε μετά Control+Z
 - Αν είστε σε Linux πατήστε μετά Control+D
- ▶ Διακόπτες του εργαλείου fsm
 - -list Τυπώνει τους κανόνες
 - -trace Τυπώνει αναλυτική τη λειτουργία αναγνώρισης
- ▶ Πχ: fsm -trace automato0-1.fsm [Windows]
- ▶ ή ./fsm -trace automato0-1.fsm [Linux]

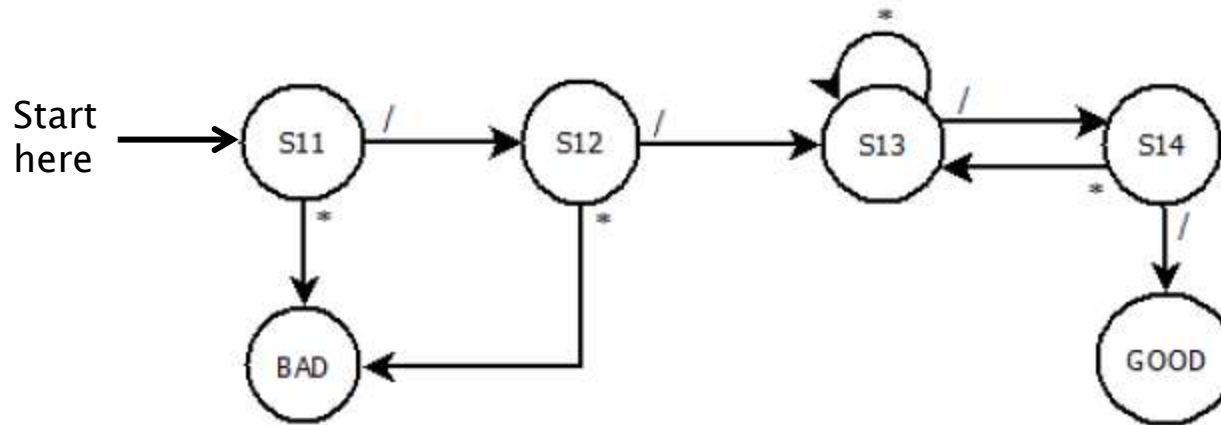
Παράδειγμα 1: Σχόλια

- ▶ Έστω C-like γλώσσα που τα σχόλια στον κώδικα ακολουθούν τους κανόνες:
 - Οτιδήποτε περικλείεται ανάμεσα σε `//` και `//` θεωρείται σχόλιο και αγνοείται
 - Μέσα στα σχόλια επιτρέπεται μόνο η εμφάνιση μοναδικής καθέτου / όσες φορές θέλουμε και όχι ομάδας δύο καθέτων και άνω

Αυτόματο / ΔΜ = ???

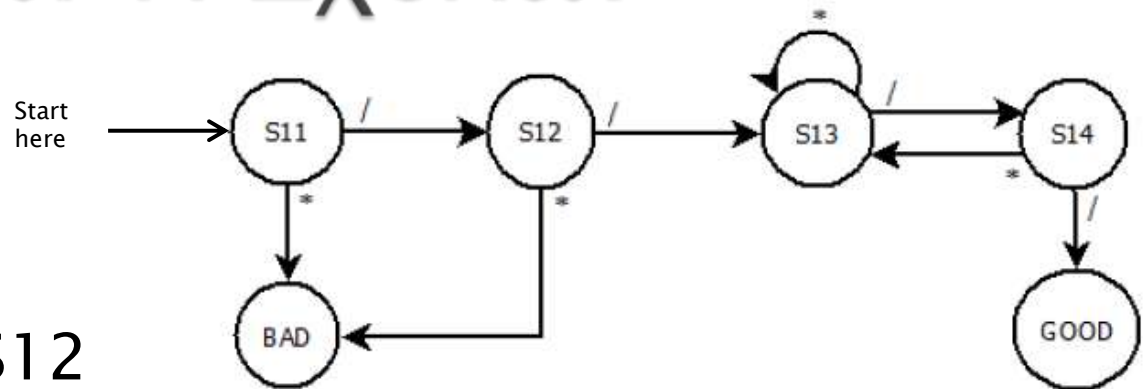
Παράδειγμα 1: Σχόλια

- ▶ Με βάση τους προηγούμενους κανόνες:



Κώδικας FSM = ???

Παράδειγμα 1: Σχόλια



► Κώδικας FSM:

S11:	/	->	S12
	*	->	BAD
S12:	/	->	S13
	*	->	BAD
S13:	/	->	S14
	*	->	S13
S14:	/	->	GOOD
	*	->	S13
GOOD(OK):	\n	->	GOOD

Παράδειγμα 1: Σχόλια

► Δοκιμαστικές εισοδοι και αποτελέσματα

- // comment //
- //hello world//
- //blah/ /
- /nope/
- blabla//

<-

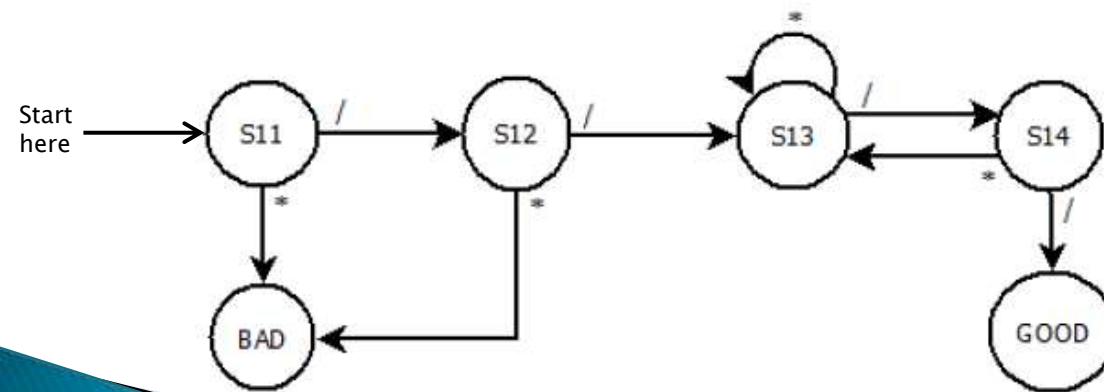
<-

<-

<-

<-

???

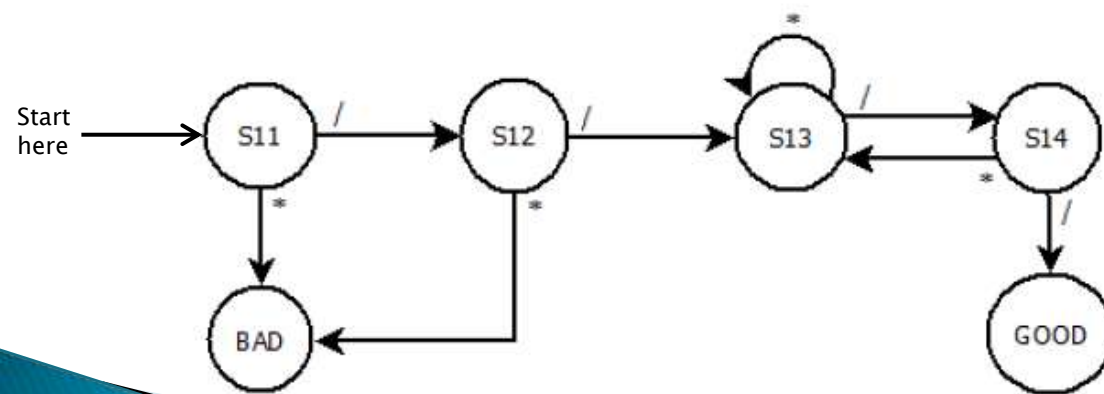


S11:	/	-> S12
	*	-> BAD
S12:	/	-> S13
	*	-> BAD
S13:	/	-> S14
	*	-> S13
S14:	/	-> GOOD
	*	-> S13
GOOD(OK):	\n	-> GOOD

Παράδειγμα 1: Σχόλια

► Δοκιμαστικές εισοδοι και αποτελέσματα

- // comment // <- YES
- //hello world// <- YES
- //blah/ / <- NO
- /nope/ <- NO (input not accepted)
- blabla// <- NO (input not accepted)



S11:	/	-> S12
	*	-> BAD
S12:	/	-> S13
	*	-> BAD
S13:	/	-> S14
	*	-> S13
S14:	/	-> GOOD
	*	-> S13
GOOD(OK):	\n	-> GOOD

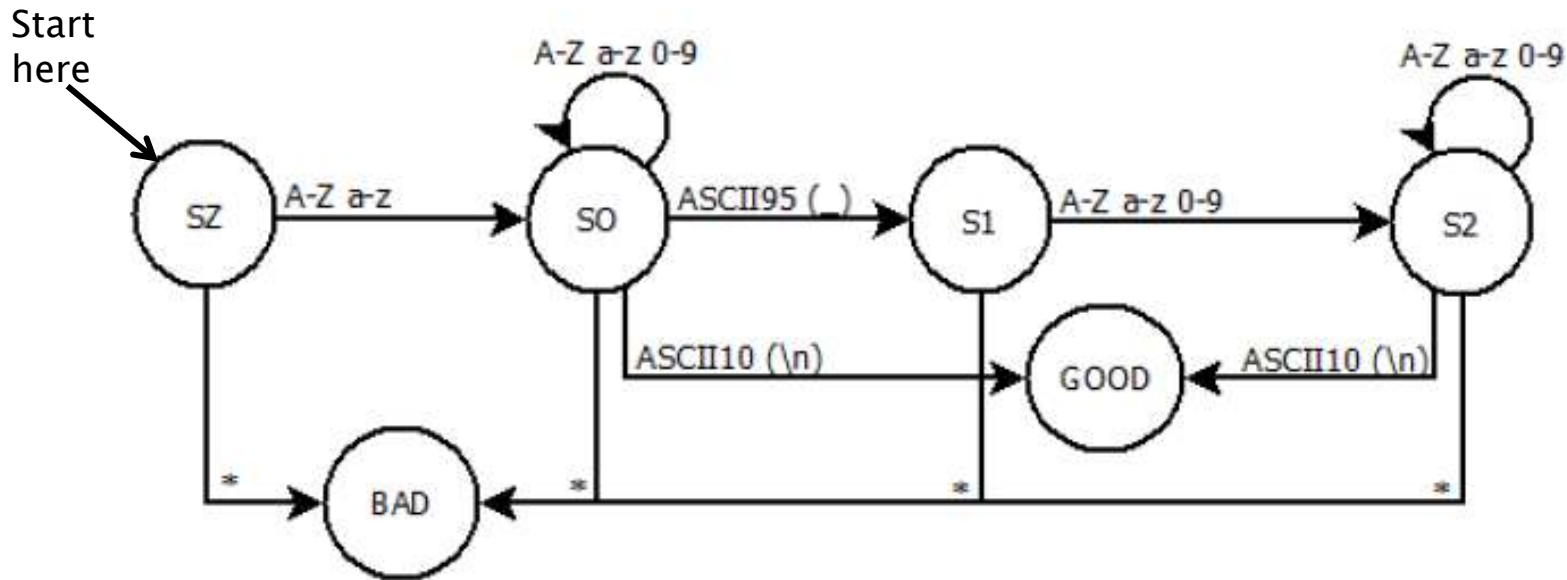
Παράδειγμα 2: Μεταβλητές

- ▶ Έστω C-like γλώσσα που τα ονόματα των μεταβλητών της:
 - Αρχίζουν από πεζό ή κεφαλαίο λατινικό χαρακτήρα
 - Μετά τον 1^ο χαρακτήρα μπορούν να περιέχουν πεζούς ή κεφαλαίους λατινικούς χαρακτήρες, αριθμητικά ψηφία ή μόνο μια κάτω παύλα
 - Η κάτω παύλα **ΔΕΝ** μπορεί να είναι στο τέλος του ονόματος

Αυτόματο / ΔΜ = ???

Παράδειγμα 2: Μεταβλητές

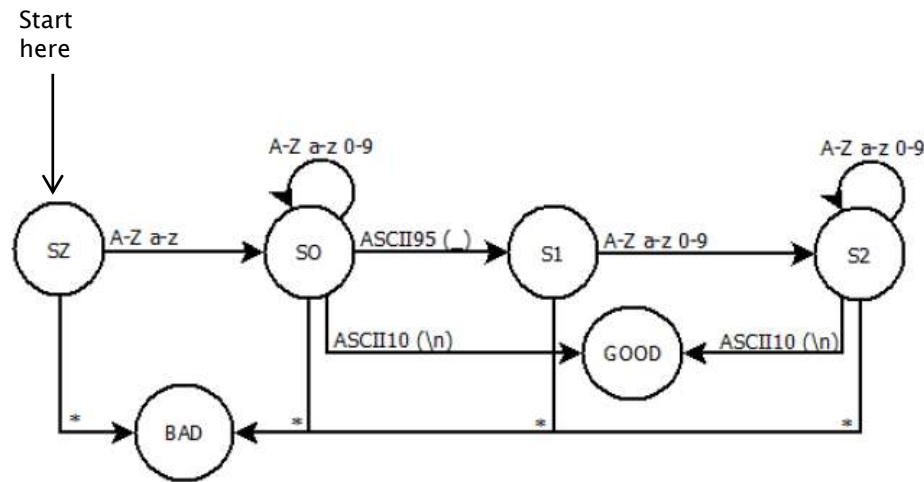
- ▶ Με βάση τους προηγούμενους κανόνες:



Κώδικας FSM = ???

Παράδειγμα 2: Μεταβλητές

► Κώδικας FSM:



SZ:	A-Z a-z	->	S0
	*	->	BAD
S0:	A-Z a-z 0-9	->	S0
	\n	->	GOOD
	-	->	S1
	*	->	BAD
S1:	A-Z a-z 0-9	->	S2
	*	->	BAD
S2:	A-Z a-z 0-9	->	S2
	\n	->	GOOD
	*	->	BAD

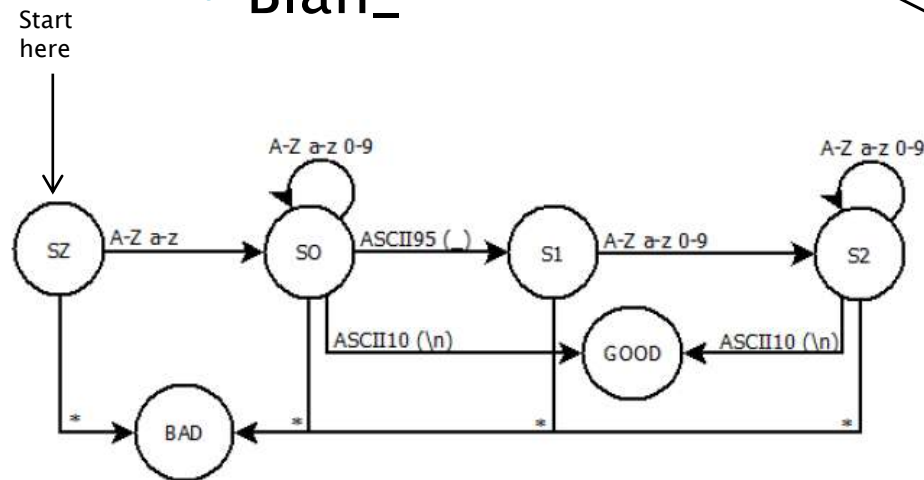
GOOD(OK):

Παράδειγμα 2: Μεταβλητές

► Δοκιμαστικές εισοδοι και αποτελέσματα

- Test <-
- test123 <-
- test123_blah <-
- _t1 <-
- Blah_ <-

???

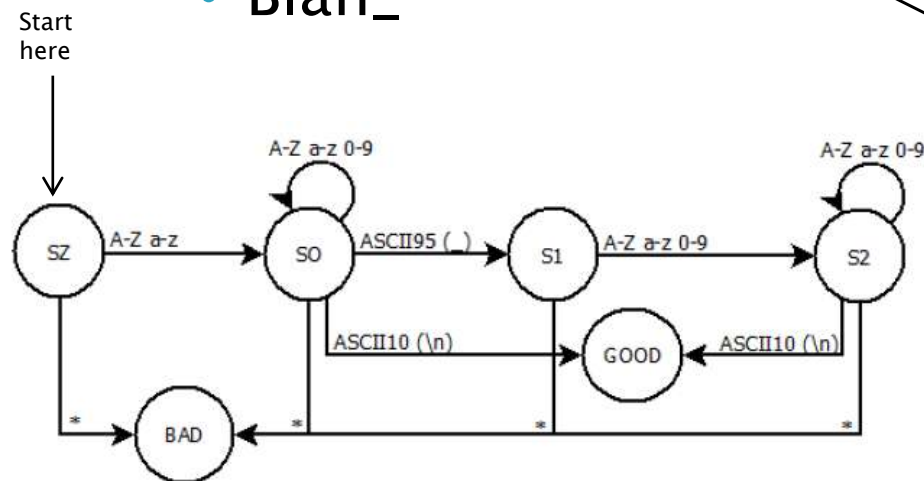


SZ:	A-Z a-z	-> S0
	*	-> BAD
S0:	A-Z a-z 0-9	-> S0
	\n	-> GOOD
	-	-> S1
	*	-> BAD
S1:	A-Z a-z 0-9	-> S2
	*	-> BAD
S2:	A-Z a-z 0-9	-> S2
	\n	-> GOOD
	*	-> BAD
GOOD(OK):		

Παράδειγμα 2: Μεταβλητές

► Δοκιμαστικές εισοδοι και αποτελέσματα

- Test <- YES
- test123 <- YES
- test123_blah <- YES
- _t1 <- NO (input not accepted)
- Blah_ <- NO



SZ:	A-Z a-z	-> S0
	*	-> BAD
S0:	A-Z a-z 0-9	-> S0
	\n	-> GOOD
	-	-> S1
	*	-> BAD
S1:	A-Z a-z 0-9	-> S2
	*	-> BAD
S2:	A-Z a-z 0-9	-> S2
	\n	-> GOOD
	*	-> BAD
GOOD(OK):		

Παράδειγμα 3

- ▶ Ένας λίγο διαφορετικός τρόπος χρήσης του εργαλείου FSM:

```
wantC: C -> wantA
        \n -> end   N
        * -> wantC
wantA: A -> wantT
        \n -> end   N
        * -> wantA
wantT: T -> happy
        \n -> end   N
        * -> wantT
happy: \n -> end   Y
        * -> happy
end: none -> telos \n
```

Άσκηση 1 (για εξάσκηση)

- ▶ Κώδικας FSM για αυτόματο που:
 - Δέχεται ακέραιους αριθμούς
 - Ο κάθε αριθμός ΔΕΝ μπορεί να ξεκινάει με 0
 - Ο αριθμός μπορεί να είναι το ίδιο το 0
 - Έγκυροι χαρακτήρες 0 έως 9

Άσκηση 2 (για εξάσκηση)

- ▶ Κώδικας FSM για αυτόματο που:
 - Δέχεται ακέραιους **και δεκαδικούς** αριθμούς
 - Ο κάθε αριθμός ΔΕΝ μπορεί να ξεκινάει με 0
 - Εξαιρούνται οι δεκαδικοί στο εύρος (0, 1)
 - Ο αριθμός μπορεί να είναι το ίδιο το 0
 - Έγκυροι χαρακτήρες 0 έως 9 και τελεία
 - Η τελεία . αποτελεί το διαχωριστικό ακεραίου και δεκαδικού μέρους

Εργασία εξαμήνου (1)

▶ Μέρος Α-2

- Κωδικοποίηση αυτομάτων πεπερασμένων καταστάσεων μέσω εργαλείου FSM

▶ Βασικά ζητούμενα

- Κανονικές εκφράσεις λεκτικών μονάδων της γλώσσας **Uni-C**
- Πεπερασμένα αυτόματα (ή ΔΜ)
 - Σημείωση: Το ενιαίο αυτόματο πρέπει να περιέχει ΜΟΝΟ επιτρεπτές εξόδους
- Παραλείψτε κρατημένες λέξεις & γενικά λέξεις για τις οποίες δεν χρειάζεται πρότυπο αναγνώρισης
 - Συμβουλευτείτε την περιγραφή της γραμματικής της Uni-C

Εργασία εξαμήνου (2)

- ▶ Βασικά ζητούμενα (συνέχεια)
 - Προσομοιώστε το ενιαίο αυτόματο σε έναν γενικό Πίνακα Μεταβάσεων (ΠΜ)
 - Στον ενιαίο ΠΜ αφήστε κενά τα κελιά που αντιστοιχούν σε ΜΗ επιτρεπτές μεταβάσεις
 - Με τη βοήθεια του εργαλείου FSM, κωδικοποιήστε τον γενικό ΠΜ για να ελέγξετε την ορθή αναγνώριση των λεκτικών μονάδων
 - Στο FSM κωδικοποιήστε τα ως μια κοινή μετάβαση BAD
 - Όλα τα επιμέρους αυτόματα/ΠΜ/κώδικες FSM θα πρέπει να υποβληθούν.
 - Ειδικά στην περίπτωση που δεν υπάρχουν τα αντίστοιχα ενιαία!

Εργασία εξαμήνου (3)

- ▶ Αναλυτικά η εκφώνηση της εργασίας:
 - 3-ΕΡΓΑΣΤΗΡΙΟ » 4-Υλικό για Εργασίες » ΕΚΦΩΝΗΣΕΙΣ ΕΡΓΑΣΙΩΝ
 - Εργασία Μεταγλωττιστών – Μέρος Α
- ▶ Μην αμελήσετε:
 - Να διαβάσετε καλά τα ζητούμενα στο παραπάνω έγγραφο (Μέρος Α-2)
 - Να δείτε τις οδηγίες για την τεκμηρίωση στο τέλος του εγγράφου
 - Να διαβάσετε προσεκτικά το έγγραφο με την περιγραφή της γραμματικής της γλώσσας Uni-C
 - Να σιγουρευτείτε ότι είναι πρώτο το όνομα του γενικού συντονιστή στην ομάδα σας στο Wiki

Εργασία εξαμήνου (4)

- ▶ Παραδοτέα (Υποβολή -> Εργασίες eClass)
 - **ΠΡΟΣΟΧΗ: Θα παραδοθεί μαζί με το μέρος A-3 !!!**
 - Ένα συμπίεσμένο αρχείο ZIP με όνομα που θα αρχίζει με τον κωδικό της ομάδας (α/α ομάδας), κάτω παύλα και το μέρος της εργασίας (A-2). Το ZIP θα περιλαμβάνει:
 - Ένα έγγραφο (Word/Writer/PDF) με όλα τα ζητούμενα και την τεκμηρίωσή τους
 - Όχι greeklish!
 - Όχι screenshots! (με εξαίρεση τα αυτόματα/ΔΜ)
 - Προσοχή στο μέγεθος του εγγράφου!
 - Το αρχείο FSM του ενιαίου αυτόματου και τα επιμέρους αρχεία FSM που έχετε δημιουργήσει

Εργασία εξαμήνου (5)

- ▶ Το έγγραφο τεκμηρίωσης (PDF ή Word/Writer) θα πρέπει να περιλαμβάνει:
 - Εξώφυλλο (με ονοματεπώνυμα μελών ομάδας, αριθμό ομάδας και τμήμα) – Πίνακα Περιεχομένων – Εισαγωγή
 - Κανονικές εκφράσεις (μόνο επιμέρους – όχι ενιαία)
 - Αυτόματα πεπερασμένων καταστάσεων ή ΔΜ (επιμέρους & ενιαίο)
 - Πίνακες μεταβάσεων (επιμέρους & ενιαίος)
 - Σχολιασμένος κώδικας FSM (επιμέρους & ενιαίος)
 - Θα πρέπει να βάλετε κώδικα & σχόλια σε ξεχωριστά αρχεία .FSM αντί για μέσα στο έγγραφο
 - Περιπτώσεις ελέγχου / **εξαντλητικές δοκιμαστικές εκτελέσεις** / αποτελέσματα (όχι screenshots – απλό copy-paste!) και σχολιασμός τους
 - Σχόλια πάνω στα παραπάνω υποβαλλόμενα (πχ. τρόπος λειτουργίας, προβλήματα που αντιμετωπίσατε, αν / πως τα λύσατε, **ελλείψεις, ορθή ή μη εκτέλεση κώδικα**, κλπ)
 - Ανάλυση αρμοδιοτήτων/ρόλων όλων των μελών της ομάδας