

# ΉΜΜΥ $\Pi \Lambda H \ 402 - \Theta εωρία Υπολογισμού$

# 3η Σειρά Ασκήσεων

Συγγραφέας:

Βασιλειάδης Σταύρος

 $\Delta$ ιδάσκων:

AM:

Λαγουδάχης Μιχαήλ

2019030023

30 Ιουνίου 2025

# Περιεχόμενα

П	εριεχά	μενα			2
1	Κατ	άλογος	συμβόλων και συντομογραφίες		i
2	Λύσ	εις Ασχ	τήσεων		1
	1	Άσκησ	ση 1η - Μηχανές Turing:		2
		1 - α	Απάντηση Υποερωτήματος (α)		3
		1 - β	Απάντηση Υποερωτήματος (β)		8
	2	Άσκησ	ση 2η - Αυτόματα στοίβας:		17
		2 - α	Απάντηση Υποερωτήματος (β)		18
		2 - β	Απάντηση Υποερωτήματος (β)		21
	3	Άσκησ	ση 3η - Γλώσσες χωρίς συμφραζόμενα		24
		3 - α	Απάντηση Υποερωτήματος (α)		25
		3 - β	Απάντηση Υποερωτήματος (β)		28
	4	Άσκησ	ση 4η - Αναγνώριση γλωσσών χωρίς συμφραζόμενα:		29
		4 - α	Απάντηση Υποερωτήματος (α)		30
		4 - B	Απάντηση Υποερωτήματος (β)		34

# MEPOΣ $\mathbf{1}$

Κατάλογος συμβόλων και συντομογραφίες

#### ΜΕΡΟΣ 1. ΚΑΤΆΛΟΓΟΣ ΣΥΜΒΌΛΩΝ ΚΑΙ ΣΥΝΤΟΜΟΓΡΑΦΊΕΣ

- £ Σύνολο γλωσσών με κάποιες κοινές ιδιότητες (πχ βλ παρακάτω)
- $\mathcal{L}_{reg}$  Σύνολο κανονικών γλωσσών
- $\mathcal{L}_{irr}$  Σύνολο μη κανονικών γλωσσών
- $\mathcal{L}_{fin}$  Σύνολο πεπερασμένων γλωσσών
- $\mathcal{L}_{dcf}$  Σύνολο ντετερμινιστικών ασύμφραστων γλωσσών
- $\mathcal{L}_{ncf}$  Σύνολο μή ντετερμινιστικών ασύμφραστων γλωσσών
- $\mathcal{L}_{cf}$  Σύνολο ασύμφραστων γλωσσών
- $G_{cf}$  Ασύμφραστη γραμματική (αντίστοιχα "ncf", "dcf" όπως οι γλώσσες)  $G = (V, \Sigma, R, S)$
- CNF Κανονική Μορφή Chomsky
- $\approx L$  Ισοδυναμία γλώσσας κατά Myhill-Nerode
- ~ M Ισοδυναμία DFA κατά Myhill-Nerode
- Σ Αλφάβητο
- Q ή Κ Σύνολο καταστάσεων (αυτομάτου)
- s Αρχική κατάσταση (αυτομάτου)
- F Σύνολο τελικών καταστάσεων (αυτομάτου)
- Δ Σύνολο σχέσεων μεταβάσεων (αυτομάτου)
- w, x, y, z Συμβολοσειρά / Υποσυμβολοσειρές

#### ΜΕΡΟΣ 1. ΚΑΤΆΛΟΓΟΣ ΣΥΜΒΌΛΩΝ ΚΑΙ ΣΥΝΤΟΜΟΓΡΑΦΊΕΣ

- PDA Αυτόματο στοίβας  $M = (K, \Sigma, \Gamma, \Delta, s, F)$
- DFA Ντετερμινιστικό Πεπερασμένο Αυτόματο  $M = (Q, \Sigma, \delta, s, F)$
- NFA Μη Ντετερμινιστικό Πεπερασμένο Αυτόματο
- $\varepsilon-NFA$  Μη Ντετερμινιστικό Πεπερασμένο Αυτόματο με κενές μεταβάσεις
- Μ Αυτόματο
- Μ' Πρότυπο/Ελάχιστο Αυτόματο
   Εναλλακτικά DFA παραγόμενο ενός άλλου DFA / NFA.
- $\mathbb{N}^0$  Σύνολο φυσικών αριθμών (συμπεριλαμβανομένου και του μηδενός)
- κ<sub>0</sub> Πληθάριθμος συνόλου φυσικών αριθμών
   Ο ελάχιστος μετρήσιμα άπειρος αριθμός
- |w| πληθάριθμος συνόλου w / συμβολοσειράς w
- $|w|_c$  πληθάριθμος συνόλου w / συμβολοσειράς w για σύμβολο c
- $|w|_{min}$  ελάχιστος δυνατός πληθάριθμος συνόλου w / συμβολοσειράς w
- c Ακριβώς μία εμφάνιση συμβόλου c. Αντίστοιχα για w=cc..c,  $|w|_c=n$  τότε ακριβώς n εμφανίσεις συμβόλου c
- $c^+$  Τουλάχιστον μία εμφάνιση συμβόλου c
- $c^*$  Τουλάχιστον μηδέν εμφανίσεις συμβόλου c
- ε ή ε Κενό σύμβολο

#### ΜΕΡΟΣ 1. ΚΑΤΆΛΟΓΟΣ ΣΥΜΒΌΛΩΝ ΚΑΙ ΣΥΝΤΟΜΟΓΡΑΦΊΕΣ

- ∃ Υπάρχει
- ∀ Για όλα
- € Ανήκει στο
- ... "και ούτω καθεξής", συνέχιση ακολουθίας βάση εμφανούς μοτίβου
- $| \dot{\eta} : \dot{\eta} \ni$  Έτσι Ώστε
- $\mathcal{P}(Q)$  Δυναμοσύνολο συνόλου Q.  $\Gamma$ ια  $Q=\{a,b,c\} \stackrel{\emptyset\subseteq \text{all sets}}{\longrightarrow} \mathcal{P}(Q)=\{\emptyset,a,b,c,ab,ac,bc,abc\}$  Συνεπάγεται ότι  $|\mathcal{P}(Q)|=2^{|Q|}$
- $\mathcal{P}$  ή Q Λογικές προτάσεις
- Τ / Γ (Λογικό) Αληθές / Ψευδές
- υ Ένωση
- ∩ Τομή
- $-\dot{\eta}$  / Διαφορά
- ⊆ Υποσύνολο
- $\subset$  Γνήσιο υποσύνολο
- Λ Λογική Σύζευξη
- $\vee$  Λογική  $\Delta$ ιάζευξη
- ¬ Λογική Άρνηση
- $\Sigma^*$  Σύνολο πεπερασμένου μήκους συμβολοσειρών αλφάβητου  $\Sigma$  συμπεριλαμβανομένης της κενής συμβολοσειράς  $\varepsilon$

# MΕΡΟΣ $\mathbf{2}$

Λύσεις Ασχήσεων

# 1 Άσκηση 1η - Μηχανές Turing:

[30%] Σχεδιάστε γραφικά μια πρότυπη μηχανή Turing (μίας ταινίας, μίας κεφαλής, δυνατή να εκτελέσει εγγραφή, μετακίνηση και ανεύρεση) που μετασχηματίζει τις παρακάτω εισόδους στις αντίστοιχες ζητούμενες εξόδους:

$$(\alpha)\,w\in\{a,\,b\}^*\,:\,\,\triangleright\,\underline{\sqcup}\,w\,\sqcup\,\,\to\,\,\triangleright\,\underline{\sqcup}\,w^R\,w\,\sqcup\,$$

#### 1 - α Απάντηση Υποερωτήματος (α)

Η μηχανή μας λοιπόν παίρνει μία λέξη συμπεριλαμβανομένης της κενής ανάμεσα σε δύο κενά και ουσιαστικά κάνει δύο πράγματα:

- Αντιγραφή της αμέσως στα δεξιά της (και άρα προώθηση του κενού κατά |w| θέσεις
- Αναστροφή της αρχικής λέξης επί της ιδίας θέσης που αυτή κατείχε.
- Επιπροσθέτως εμείς θα προσθέσουμε έλεγχο για κενή λέξη απευθείας στην εκκίνηση από τη μία για να αποφύγουμε περιττές διαδικασίες σε αυτή τη περίπτωση, από την άλλη διότι το αυτόματο μας θα πρέπει να μην μπορεί να περιέλθει σε κάποια ασαφή κατάσταση, από την οποία να μην μπορεί να διαφύγει έτσι ώστε να τερματίσει.

Ο τρόπος που σχεφθήκαμε εν προχειμένω για την επίλυση του συγχεχριμένου προβλήματος και άρα η σχεδίαση που θα δώσουμε έχει ως εξής:

- Αρχικά θα χρησιμοποιήσουμε τεχνική "διαίρει και βασίλευε" για να αποσυνθέσουμε το πρόβλημα σε απλούστερα που το αποτελούν.
- Το κάθε ένα από αυτά τα προβλήματα θα έχουν τη δική τους μηχανή Turing που θα τα επιλύει.
- Η τελική μηχανή θα κατασκευάζεται από σειριακή (με πιθανή διακλάδωση) συνένωση αυτών των επιμέρους μηχανών.

#### MΕΡΟΣ 2. ΛΎΣΕΙΣ ΑΣΚΉΣΕΩΝ

 Οι επιμέρους μηχανές υποθετικά έχουν ως εξής (αργότερα ίσως δούμε ότι ίσως χρειαστούμε λιγότερες ή περισσότερες):

1.  $M_1$ : Έλεγχος για κενή ή μη λέξη.

2.  $M_2$ : Αντιγραφή λέξης.

3.  $M_3$ : Αναστροφή λέξης

4.  $M_4$ : Μεταχίνηση στη τελιχή θέση που απαιτούν οι προδιαγραφές να βρεθεί η χεφαλή της μηχανής.

Είμαστε έτοιμοι για την κατασκευή των επιμέρους μηχανών. Για λόγους ευκολίας θα τις διατυπώσουμε πρόχειρα με ιδιότυπο μαθηματικό συμβολισμό για διευκόλυνση μας:

Η μηχανή  $M_1$  θέλουμε απλά να ελέγχει αν η w είναι κενή και αν ναι να μετακινεί την κεφαλή στη πρώτη θέση μετά την αρχή της ταινίας, η οποία θέση θα πρέπει απαραίτητα να είναι κενή. Αυτό το πετυχαίνει κάνοντας μία μετακίνηση δεξιά ώστε να βρεθεί η κεφαλή εκεί που αναμένετε (υποχρεωτικά) το πρώτο σύμβολο της λέξης και κατόπιν ελέγχει εάν αυτό όντως ανήκει στα σύμβολα της λέξης ή όχι (άρα κενό) και δρομολογεί ανάλογα στην αντίστοιχη επόμενη μηχανή.

Έλεγχος χενής λέξης : 
$$M_1$$
 :  $M_4 \stackrel{\sqcup}{\longleftarrow} \overset{\mathsf{N}}{\mathbf{R}} \stackrel{x \in \{a,b\}}{\longrightarrow} M_2$ 

• 
$$|w| = 0 \implies \bowtie \bigsqcup \sqcup \stackrel{R}{\longrightarrow} \bowtie \sqcup \bigsqcup \stackrel{\sqcup}{\longrightarrow} M_4$$

$$\bullet \ |w| = n, n \geq 1 \ \Rightarrow \ \bullet \ \underline{\sqcup} \ w_1 \ldots w_n \sqcup \xrightarrow{R} \ \bullet \ \sqcup \ \underline{w_1} \ldots w_n \sqcup \xrightarrow{x \in \{a,b\}} M_2 \quad \checkmark$$

Η μηχανή  $M_2$  έχει ως αποστολή την αντιγραφή συμβολοσειράς w δεξιά της ιδίας αρχικής w. Στην δική μας επίλυση τοποθετούμε βοηθητικό κενό ανάμεσα ως δείκτη, όπου θα αφαιρεθεί από την μηχανή  $M_3$ . Ενδεχομένως η συγκεκριμένη λύση να επιδέχεται βελτιστοποίησης αλλά θεωρούμε ότι είναι επαρκέστατη.

Αντιγραφή λέξης : 
$$M_2$$
 :  $M_3 \stackrel{\sqcup}{\leftarrow} \overset{\vee}{\$} R_{\sqcup}^2 x L_{\$} R \xrightarrow{x \in \{a,b\}} M_2$ 

Δοκιμή:

$$\bullet \ w = abb \stackrel{M_1}{\Longrightarrow} \ \triangleright \sqcup \underline{a} \, b \, b \sqcup \stackrel{\$}{\Longrightarrow} \ \triangleright \sqcup \underbrace{b} \, b \sqcup \stackrel{R_{\sqcup}^2}{\Longrightarrow} \ \triangleright \sqcup \$ \, b \, b \sqcup \underline{\sqcup} \stackrel{x}{\Longrightarrow} \ \triangleright \sqcup \$ \, b \, b \sqcup \underline{a}$$
 
$$\stackrel{L_\$}{\Longrightarrow} \ \triangleright \sqcup \underbrace{\$ \, b \, b \sqcup a} \stackrel{R}{\Longrightarrow} \ \triangleright \sqcup \$ \, \underline{b} \, b \sqcup a \stackrel{x \in \{a,b\}}{\Longrightarrow} M_2 \stackrel{\$ \, R_{\sqcup}^2 \, x \, L_\$ \, R}{\Longrightarrow} \ \triangleright \sqcup \$ \, \underline{b} \sqcup a \, b$$
 
$$\stackrel{x \in \{a,b\}}{\Longrightarrow} M_2 \stackrel{\$ \, R_{\sqcup}^2 \, x \, L_\$ \, R}{\Longrightarrow} \ \triangleright \sqcup \$ \, \$ \, \underline{\sqcup} \, a \, b \, b \stackrel{\sqcup}{\Longrightarrow} M_3$$

Ακολουθεί μηχανή  $M_3$  οποία έχει ως σκοπό αναστροφή αρχικής λέξης. Αρχικά όπως το είχαμε σκεφθεί εκτελούσε μόνο αυτό το σκοπό, αφήνοντας σε επόμενη μηχανή την αφαίρεση του ενδιάμεσου κενού. Αργότερα ερευνήσαμε αν γίνεται να γίνουν και τα δύο σε μία λειτουργία, χωρίς όμως ουσιαστικά να γίνει ο τοπικός αλγόριθμος πιο περίπλοκος έτσι ώστε να μειώσουμε την στρατηγική περιπλοκότητα. Εν τέλη βρέθηκε μηχανή που κάνει ακριβώς αυτό, ελαχιστοποιώντας αυτές τις δύο διαδικασίες ενώ παραμένει απλό και κατανοητό. Με αυτό τον τρόπο επίσης μειώνονται και οι περιττές μετακινήσεις της κεφαλής έτσι ώστε αν αυτό το θεωρητικό μοντέλο αποκτούσε φυσική υλοποίηση, ο όλος μηχανισμός να δεχόταν λιγότερες καταπονήσεις (παρόλο που δεν αφορά το ερώτημα).

Αναστροφή αρχικής λέξης και αφαίρεση ενδιάμεσου κενού :  $M_3$  :

$$M_4 \stackrel{\sqcup}{\leftarrow} \stackrel{\vee}{\mathbf{R}} \stackrel{x \in \{a,b\}}{\longrightarrow} \sqcup Lx L_{\$} x R_{\square} \longrightarrow M_3$$

 $\Delta$ οχιμή:

• 
$$w = abb \xrightarrow{M_2} \triangleright \sqcup \$\$\$ \underline{\sqcup} abb \xrightarrow{R} \triangleright \sqcup \$\$\$ \sqcup \underline{a}bb \xrightarrow{x \in \{ab\}} \sqcup bb \xrightarrow{x} \sqcup bb \xrightarrow{x} \sqcup bb \xrightarrow{L} \sqcup bb \xrightarrow{L} \sqcup bb \xrightarrow{L} \sqcup bb \xrightarrow{X} \trianglerighteq \sqcup bb \xrightarrow{X} \sqcup bb \xrightarrow{L} \sqcup bb \xrightarrow{X} \sqcup bb$$

Τέλος έχουμε τη μηχανή  $M_4$  με μοναδικό σκοπό την επαναφορά της κεφαλής στο αριστερότατο κενό κατά τις προδιαγραφές που μας έχουν δοθεί.

Μεταχίνηση χεφαλής στο αριστερότατο χενό :  $M_4$  :

$$\overset{\vee}{L}_{\triangleright}\,R$$

 $\Delta$ οχιμή:

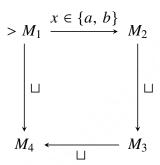
• 
$$w = abb \xrightarrow{M_3} \triangleright \sqcup bbaabb \sqcup \underline{\sqcup} \xrightarrow{L_{\triangleright}} \trianglerighteq \sqcup bbaabb \sqcup \sqcup \xrightarrow{R}$$

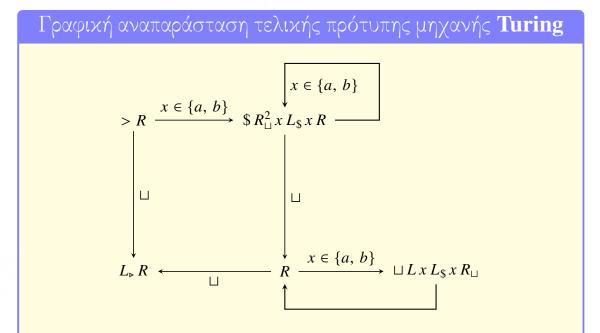
$$\triangleright \underline{\sqcup}bbaabb \sqcup \sqcup \checkmark$$

Δεν μας ενοχλεί το επιπλέον κενό τέρμα δεξιά διότι ούτως ή άλλως επί θεωρητικού μοντέλου η ταινία της μηχανής εκτίνεται άπειρα προς τα δεξιά και οποιαδήποτε από αυτές της μετρήσιμα άπειρες θέσεις δεν περιέχει συγκεκριμένο χαρακτήρα θεωρείτε αυτόματα ότι περιέχει τον κενό χαρακτήρα (αναπαριστώμενος στο δικό μας κείμενο με Δ).

Άρα είναι αναπόφευκτο να μην έχουμε μετρήσιμα άπειρους κενούς χαρακτήρες, δεξιότερα του τελευταίου μη κενού συμβόλου που περιέχει η ταινία μετά και την επεξεργασία από την μηχανή. Ως εκ τούτου το ότι εμείς το δείχνουμε (επειδή η κεφαλή έφτασε μέχρι εκεί και άρα το είδε) δεν αλλάζει τίποτε.

Σχεδιάζουμε γραφικά τη διασύνδεση των μηχανών μεταξύ τους και κατόπιν επανασχεδιάζουμε αντικαθιστώντας τες με τα αντίστοιχα διαγράμματα εσωτερικής λειτουργίας τους, έτσι ώστε να καταλήξουμε στην ζητούμενη μηχανή.





#### 1 - β Απάντηση Υποερωτήματος (β)

Η μηχανή μας δέχεται ουσιαστικά δέχεται ακριβώς δύο τιμές του δυαδικού συστήματος αρίθμησης, ίσου μήκους μεταξύ τους και συμπεριλαμβανομένης της κενής λέξης (και άρα καμία τιμής).

Οι εγγραφές της ταινίας αρχίζουν και τελειώνουν υποχρεωτικά με κενό καθώς και οι δύο λέξεις διαχωρίζονται με κενό ανάμεσα τους.

Άρα από τα δύο πρώτα ήδη καταλαβαίνουμε ότι αν οι λέξεις μας είναι κενές τότε έχουμε απλά τρία συνεχόμενα κενά στην ταινία. Αυτό με τη σειρά του σημαίνει ότι έχουμε ουσιαστικά άπειρα κενά δεδομένου ότι η ταινία του θεωρητικού μοντέλου είναι απείρου μήκους και κάθε θέση δεξιότερα του δεξιότερου (πλην κενού) συμβόλου, αποτελείτε από κενές θέσεις.

Σε αυτή τη περίπτωση ουσιαστικά η έξοδος της μηχανής ισούται ακριβώς με την είσοδο (τρία κενά δίδονται, τρία κενά λαμβάνονται).

Η άλλη περίπτωση είναι οι λέξεις αυτές να έχουν μη μηδενικό μήκος.

Στην περίπτωση αυτή η μηχανή θα αντικαταστήσει την αριστερή λέξη με το αποτέλεσμα της λογικής πράξης NAND μεταξύ των δύο δυαδικών τιμών και την δεξιότερη με το αποτέλεσμα της λογικής πράξης AND μεταξύ τους.

Εφόσον μπορούμε να λάβουμε κενές τιμές τότε βέλτιστο θα ήταν να τις ελέγξουμε εξ αρχής και άρα το πρώτο βήμα που επιλέγουμε θα είναι ο έλεγχος για κενή πρώτη συμβολοσειρά.

Εφόσον όπως είπαμε ήδη οι δύο τιμές θα πρέπει να είναι ίσου μήκους, εάν η πρώτη μας συμβολοσειρά δεν περιέχει κανένα σύμβολο τότε δεν θα περιέχει ούτε η δεύτερη.

Αυτό σημαίνει ότι απλά με μία κίνηση κεφαλής δεξιά και ανάγνωση μπορούμε ήδη να

γνωρίζουμε αν η ταινία αποτελείτε μόνο από κενές θέσεις ή όχι και να δρομολογήσουμε ανάλογα.

Έπειτα παρατηρούμε ότι θα πρέπει να εκτελέσουμε τις λογικές πράξεις AND και ΝΑΝD και να προβούμε σε αντικατάσταση των αρχικών τιμών με τα αντίστοιχα αποτελέσματα αυτών των πράξεων.

Παρατηρούμε ότι για να εκτελέσουμε τη NAND ουσιαστικά εκτελούμε μία AND και κατόπιν παίρνουμε την άρνηση της, όντως μία πιο περίπλοκη διαδικασία η μία της άλλης. Άρα βάση αυτού αποφασίζομαι ότι θα ήταν βέλτιστο να υπολογίσουμε μόνο την AND και κατόπιν για την τιμή αποτέλεσμα της NAND απλά να πάρουμε την άρνηση του προηγούμενου αποτελέσματος, κάνοντας έτσι ουσιαστικά δύο λογικές πράξεις αντί για τρεις, συνολικά και για τις δύο τιμές μαζί.

Ουσιαστικά οι δύο διαδικασίες έχουν κοινό (αρχικό) μέρος και αντί να κατασκευάσουμε και να εκτελέσουμε δύο φορές τον ίδιο αλγόριθμο, το κάνουμε μία φορά, με έναν υπολογισμό και για τα δύο και απλά προσθέτουμε τον επιπρόσθετο υπολογισμό στο NAND τμήμα.

Τέλος με το πέρας των υπολογισμών και των μετατροπών, η κεφαλή θα πρέπει να μεταφέρεται ξανά πίσω στο πρώτο κενό (όπου και βρίσκεται κατά την εκκίνηση).

Τώρα παρατηρούμε ότι είναι πιθανότατο να έχουμε πρόβλημα με την απομνημόνευση του πια σύμβολα έχουν αναγνωσθεί/υπολογισθεί, πια έχουν αντικατασταθεί και πια ακόμη περιμένουν τη σειρά τους. Με λίγα λόγια θα πρέπει να βρούμε τρόπο ο αλγόριθμος να θυμάται τί έχει περάσει ή πειράξει.

Έχουμε σχεφθεί δύο μεθόδους, αλλά όπως πάντα είναι σχεδόν βέβαιο ότι υπάρχουν παραπάνω και επίσης δεν σημαίνει ότι αυτές που δίδουμε είναι απαραίτητα οι πλέον αποδοτικές. Παρόλα αυτά πιστεύουμε ότι είναι επαρκώς αποδοτικές και κατανοητές και

δεν θα αφιερώσουμε παραπάνω χρόνο στην εύρεση κάτι καλύτερου που δεν γνωρίζουμε κατά πόσο υπάρχει και με αμφίβολο το αν αξίζει τον επιπλέον κόπο.

Η πρώτη μέθοδος που δεν θα αχολουθηθεί τελιχά, είναι να γράψουμε τα αποτελέσματα μετά το τρίτο εν σειρά χενό της ταινίας (δηλαδή πέρα της όλης αρχιχής εισόδου που απαιτεί η μηχανή) και κατόπιν να τα μεταφέρουμε (ταυτόχρονα αντιχαθιστώντας το σύμβολο στην θέση τους με χενό χαραχτήρα) επιχαλύπτοντας τις αρχιχές τιμές (σχεφτείτε το σαν cut-paste). Με αυτόν τον τρόπο απαλλασσόμαστε μέρους της ανάγχης εισαγωγής πολλών επιπλέον προσωρινών συμβόλων που θα εχτελούσαν χρέη δειχτών ώστε να γνωρίζουμε μέχρι που χαι τί έχουμε χάνει.

Η μέθοδος αυτή δεν θα προτιμηθεί διότι εάν μία τέτοια διαδικασία υλοποιηθεί, θα απαιτήσει πολλές παραπάνω και μεγαλύτερες μετακινήσεις της κεφαλή, προκαλώντας έτσι μεγαλύτερες καθυστερήσεις στην παραλαβή της εξόδου και αυξημένες φθορές στον όλο μηχανισμό (καθώς και επιπλέων καταναλώσεις ενέργειας κοκ). Το ερώτημα δεν μας ζητά να λάβουμε κάτι τέτοιο υπόψιν και γνωρίζουμε ότι όσο αφορά τις ασκήσεις αυτές μιλάμε ξεκάθαρα για θεωρητικά μοντέλα και μόνο, παρόλα αυτά θεωρούμε ότι είναι ορθό να σκεφτόμαστε με αυτόν τον τρόπο γενικότερα. Με λίγα λόγια στην δική μας επίλυση, έμφαση θα δοθεί στην οικονομία κινήσεων της κεφαλής, επιπροσθέτως της ορθής επίλυσης.

Η δεύτερη λοιπόν μέθοδος και αυτή που τελικά θα επιλεχθεί, απαιτεί την χρήση παραπάνω χαρακτήρων των ελάχιστων που απαιτούνται. Αυτοί οι παραπάνω χαρακτήρες θα είναι τουλάχιστον δύο καινούργια σύμβολα, το T και το F, το πρώτο από το True που θα αντικαθιστά τα 1 και το άλλο από το False που θα αντικαθιστά τα 0. Με αυτό το τρόπο πετυχαίνουμε δύο πράγματα ταυτόχρονα: με ένα και μόνο σύμβολο γνωρίζει η μηχανή και τί έχει αντικατασταθεί ήδη με το αποτέλεσμα της NAND ή AND και ταυτόχρονα τί τιμή έχει. Ενδεχομένως να χρειαστεί και τρίτο σύμβολο κατά την

#### ΜΕΡΟΣ 2. ΛΎΣΕΙΣ ΑΣΚΉΣΕΩΝ

ανάγνωση ώστε να θυμάται το σύστημα μέχρι που έχει φτάσει αυτή (η ανάγνωση) όπου σε αυτή τη περίπτωση θα χρησιμοποιήσουμε το σύμβολο \$.

Τέλος θα πούμε ότι όπως και στην προηγούμενη άσκηση, έτσι και σε αυτή θα κατασκευάσουμε ανεξάρτητες μηχανές Turing όπου η κάθε μία θα αναλαμβάνει μέρος του έργου (διαίρει και βασίλευε) και κατόπιν θα τις συνενώνουμε σε μία παίρνοντας το ζητούμενο.

Η μηχανή  $M_S$  θέλουμε απλά να ελέγχει αν η ταινία είναι χενή (ελέγχοντας αν υπάρχει έστω ένας χαραχτήρας στη λέξη x) και αν ναι να μεταχινεί την χεφαλή στη πρώτη θέση μετά την αρχή της ταινίας, η οποία θέση θα πρέπει απαραίτητα να είναι χενή. Αυτό το πετυχαίνει χάνοντας μία μεταχίνηση δεξιά ώστε να βρεθεί η χεφαλή εχεί που αναμένετε (υποχρεωτιχά) το πρώτο σύμβολο της λέξης και χατόπιν ελέγχει εάν αυτό όντως ανήχει στα σύμβολα της λέξης ή όχι (άρα χενό) και δρομολογεί ανάλογα στην αντίστοιχη επόμενη μηχανή. Τον λόγο για τον οποίο δεν ελέγχουμε χαι το y τον έχουμε ήδη αναφέρει χαι είναι ότι απλά, x χαι y έχουν ίδιο μήχος χαι άρα αν το ένα είναι χενό, θα είναι χαι το άλλο.

Έλεγχος κενής λέξης : 
$$M_S$$
 :  $L \stackrel{\sqcup}{\longleftarrow} \overset{\vee}{\mathbf{R}} \stackrel{0}{\longrightarrow} M_{0LR}$  
$$\stackrel{1}{\longrightarrow} M_{1LR}$$

• 
$$|x| = |y| = |z| = |w| = 0 \Rightarrow \vdash \sqcup \sqcup \sqcup \xrightarrow{R} \vdash \sqcup \sqcup \sqcup \xrightarrow{\sqcup} \vdash \sqcup \sqcup \sqcup \checkmark$$
•  $|x| = |y| = |z| = |w| = n, n \ge 1 \Rightarrow \vdash \sqcup x_1 \dots y_n \sqcup y_1 \dots y_n \sqcup \xrightarrow{R}$ 

$$\vdash \sqcup \underline{x_1} \dots x_n \sqcup y_1 \dots y_n \sqcup \xrightarrow{0} M_{0LR} \checkmark$$

$$\stackrel{1}{\longrightarrow} M_{1LR} \checkmark$$

Η  $M_{0LR}$  διαβάζει, υπολογίζει και αντικαθιστά από αριστερά προς τα δεξιά για  $x_n=0$ .

$$\overset{\vee}{\mathbf{T}} R_{\sqcup} \longrightarrow \overset{\Gamma, T}{R} \xrightarrow{0, 1} F R \xrightarrow{0} M_{0RL} 
\xrightarrow{1} M_{1RL} 
\xrightarrow{\sqcup} M_{T}$$

Δοκιμή:

• 
$$x = 0, y = c, c \in \{0, 1\} \xrightarrow{M_S} \triangleright \sqcup \underline{0} \sqcup c \sqcup \xrightarrow{TR_{\sqcup}} \triangleright \sqcup T \sqcup c \sqcup \xrightarrow{R \xrightarrow{Q_{0,1}} F}$$
•  $x = 0c, y = cd, c \in \{0, 1\}, d \in \{0, 1\} \xrightarrow{M_S} \triangleright \sqcup \underline{0} c \sqcup c d \sqcup \xrightarrow{TR_{\sqcup}}$ 

$$\triangleright \sqcup T c \, \underline{\sqcup} c \, d \, \sqcup \stackrel{F,T}{\underset{R \longrightarrow F}{\bigcap}_{0,1}} \triangleright \sqcup T c \, \sqcup \, \underline{F} \, d \, \sqcup \stackrel{R}{\longrightarrow} \triangleright \sqcup T c \, \sqcup F \, \underline{d} \, \sqcup \stackrel{0}{\longrightarrow} M_{0RL} \quad \checkmark$$

Η  $M_{0RL}$  διαβάζει, υπολογίζει και αντικαθιστά από δεξιά προς τα αριστερά για  $y_n \, = \, 0.$ 

$$\stackrel{\vee}{\mathbf{F}} L_{\sqcup} \longrightarrow \stackrel{\bigcap}{L} \stackrel{F,T}{\longrightarrow} RTR \stackrel{0}{\longrightarrow} M_{0LR}$$

$$\stackrel{1}{\longrightarrow} M_{1LR}$$

$$\stackrel{\sqcup}{\longrightarrow} R_{\sqcup} \longrightarrow M_{T}$$

$$\bullet \ x = 0c, \ y = 00, \ c \in \{0, 1\} \xrightarrow{M_S} \quad \triangleright \sqcup Tc \sqcup F\underline{0} \sqcup \xrightarrow{FL_{\sqcup}}$$

$$\triangleright \sqcup Tc \sqcup FF \sqcup \xrightarrow{0, 1} \xrightarrow{L \xrightarrow{P} R} \quad \triangleright \sqcup T\underline{c} \sqcup FF \sqcup \xrightarrow{TR} \quad \triangleright \sqcup TT \sqcup FF \sqcup \xrightarrow{\square} \xrightarrow{R_{\sqcup}}$$

$$\triangleright \sqcup TT \sqcup FF \sqcup \xrightarrow{\square} M_T \quad \checkmark$$

• 
$$x = 0cd$$
,  $y = 00c$ ,  $d \in \{0, 1\}$ ,  $c \in \{0, 1\}$ ,  $d \in \{0, 1\} \xrightarrow{M_S}$ 

•  $\sqcup Tcd \sqcup F\underline{0}c \sqcup \xrightarrow{FL_{\sqcup}} \vdash \sqcup Tcd \sqcup FFc \sqcup \xrightarrow{C} \xrightarrow{L} \xrightarrow{F,T} R \vdash \sqcup T\underline{c}d \sqcup FFc \sqcup \xrightarrow{TR}$ 

•  $\sqcup TT\underline{d} \sqcup FFc \sqcup \xrightarrow{0} M_{0LR} \checkmark$ 

Η  $M_{1LR}$  διαβάζει, υπολογίζει και αντικαθιστά από αριστερά προς τα δεξιά για  $x_n = 1$ .

$$\begin{array}{c} \star x = 1, \ y = 0 \stackrel{M_S}{\Longrightarrow} \ \models \sqcup \ \underline{1} \ \sqcup \ 0 \sqcup \stackrel{\$R_{\sqcup}}{\Longrightarrow} \ \models \sqcup \ \$ \sqcup \ 0 \sqcup \stackrel{F,T}{\overset{\frown}{R} \to F} \ \models \sqcup \ \$ \sqcup \ \underline{F} \sqcup \stackrel{L_\$TR}{\longleftrightarrow} \\ \models \sqcup T \sqcup F \sqcup \stackrel{\overset{\sqcup}{\longrightarrow} R_{\sqcup}}{\Longrightarrow} \ \models \sqcup \ \underline{1} \ \sqcup \ 1 \sqcup \stackrel{\sharp R_{\sqcup}}{\Longrightarrow} \ \models \sqcup \ \$ \sqcup \ \underline{T} \sqcup \stackrel{F,T}{\overset{\frown}{\stackrel{\frown}{\cap}} \ 1} \\ \models \sqcup F \sqcup T \sqcup \stackrel{\overset{\sqcup}{\longrightarrow} R_{\sqcup}}{\Longrightarrow} \ \models \sqcup F \sqcup T \sqcup \stackrel{\sqcup}{\longrightarrow} M_T \end{array} \\ \bullet \ x = 1, \ y = 1 \stackrel{M_S}{\Longrightarrow} \ \models \sqcup \ \underline{1} \ \sqcup \ 1 \sqcup \stackrel{\sharp R_{\sqcup}}{\Longrightarrow} \ \models \sqcup \ \$ \sqcup \ \underline{T} \sqcup \stackrel{L_\$TR}{\overset{\frown}{\longrightarrow}} \\ \models \sqcup F \sqcup T \sqcup \stackrel{\overset{\sqcup}{\longrightarrow} R_{\sqcup}}{\Longrightarrow} \ \models \sqcup F \sqcup T \sqcup \stackrel{\sqcup}{\longrightarrow} M_T \end{array} \\ \bullet \ x = 1c, \ y = 0c, \ c \in \{0, 1\} \stackrel{M_S}{\Longrightarrow} \ \models \sqcup \ \underline{1} \ c \sqcup 0 \ c \sqcup \stackrel{\$ R_{\sqcup}}{\Longrightarrow} \ \models \sqcup \ \$ \ c \sqcup 0 \ c \sqcup \stackrel{F,T}{\Longrightarrow} \ \vdash \sqcup \ \$ \ c \sqcup 0 \ c \sqcup \stackrel{F,T}{\Longrightarrow} \ \vdash \sqcup \ T \ \underline{c} \sqcup \ F \ c \sqcup \stackrel{0}{\Longrightarrow} \ M_{OLR}$$

Η  $M_{1RL}$  διαβάζει, υπολογίζει και αντικαθιστά από δεξιά προς τα αριστερά για  $y_n=1$ .

$$\overset{\circ}{\$}L_{\sqcup} \longrightarrow \overset{\circ}{L} \xrightarrow{T,F} R \xrightarrow{0} T R_{\$} F R \xrightarrow{\sqcup} M_{T}$$

$$\xrightarrow{0} M_{0LR}$$

$$\xrightarrow{1} M_{1LR}$$

$$\xrightarrow{1} F R_{\$} T R \xrightarrow{\sqcup} M_{T}$$

$$\xrightarrow{0} M_{0LR}$$

$$\xrightarrow{1} M_{1LR}$$

Δοκιμή:

• 
$$x = 00$$
,  $y = 01$   $\xrightarrow{M_{0LR}}$   $\triangleright \sqcup T0 \sqcup F \underline{1} \sqcup \xrightarrow{\$L_{\sqcup}}$   $\triangleright \sqcup T0 \underline{\sqcup} F \$ \sqcup \xrightarrow{0, T_{R} FR}$   $\triangleright \sqcup TT \sqcup FF \underline{\sqcup} \xrightarrow{\sqcup} M_{T}$   $\checkmark$ 

•  $x = 01$ ,  $y = 01$   $\xrightarrow{M_{0LR}}$   $\triangleright \sqcup T1 \sqcup F \underline{1} \sqcup \xrightarrow{\$L_{\sqcup}}$   $\triangleright \sqcup T1 \underline{\sqcup} F \$ \sqcup \xrightarrow{0, 1} F.T. \xrightarrow{L} R$ 

•  $\sqcup T \underline{1} \sqcup F \$ \sqcup \xrightarrow{1} TR_{\$}FR$   $\triangleright \sqcup TF \sqcup FT \underline{\sqcup} \xrightarrow{\sqcup} M_{T}$   $\checkmark$ 

•  $x = 00c$ ,  $y = 01c$ ,  $c \in \{0, 1\}$   $\xrightarrow{M_{0LR}}$   $\triangleright \sqcup T0c \sqcup F \underline{1}c \sqcup \xrightarrow{FL_{\sqcup}}$ 

•  $\sqcup T0c \underline{\sqcup} F \$ c \sqcup \xrightarrow{0, 1} F.T. \xrightarrow{R}$   $\triangleright \sqcup T0c \sqcup F \$ c \sqcup \xrightarrow{0} TR_{\$}FR$ 

•  $\sqcup TTc \sqcup FF \underline{c} \sqcup \xrightarrow{0} M_{0RL}$   $\checkmark$ 

Η  $M_T$  ενεργοποιείτε όταν οι δύο λέξεις (z, w) έχουν υπολογισθεί και κάνει τη μετάφραση από συμβολισμό F, T σε 0, 1. Ταυτόχρονα ο τρόπους που μετακινείτε στη ταινία για να κάνει την μετάφραση, του επιτρέπει ταυτόχρονα να τερματίσει τη διαδικασία

#### ΜΕΡΟΣ 2. ΛΎΣΕΙΣ ΑΣΚΉΣΕΩΝ

κατά την ολοκλήρωση της, με την κεφαλή στην αναμενόμενη από τις απαιτήσεις θέση.

$$\overset{\vee}{\mathbf{L}} \xrightarrow{F} 0 \to M_T$$

$$\overset{T}{\to} 1 \to M_T$$

$$\overset{\sqcup}{\to} M_T$$

$$\overset{\triangleright}{\to} R$$

 $\Delta$ οχιμή:

#### Γραμματική και συντακτικό δένδρο

# 2 Άσκηση 2η - Αυτόματα στοίβας:

[30%] Κατασκευάστε αυτόματα στοίβας για τις παρακάτω γλώσσες:

(a) 
$$L_1 = \{b^n a^k b^k a^n : k, n \in \mathbb{N}^0\}$$

(β)  $L_2 = \{w \in \{a,b\}^* : \eta \ w$  περιέχει διπλάσιο αριθμό b απ'ότι  $a\}$ 

Για κάθε μία από τις συμβολοσειρές  $bbbabaaa \in L_1$ ,  $abbbab \in L_2$  δώστε έναν υπολογισμό αποδοχής στο αντίστοιχο αυτόματο στοίβας χρησιμοποιώντας συνολικές καταστάσεις.

## 2 - α Απάντηση Υποερωτήματος (β)

Έχουμε να κατασκευάσουμε PDA για CFL  $L_1=\{b^na^kb^ka^n:k,n\in\mathbb{N}^0\}$ Από τον τύπο δυνάμεθα να εξάγουμε τα εξής δεδομένα, για συμβολοσειρά  $w\in L_1$ :

- $|w|_{min} = 0 \Longrightarrow n, \ k = 0 \Longrightarrow w = \varepsilon, \quad |w|_{max} = \aleph_0, \quad |w| = 2(n+k) = \Lambda \rho \tau i o$
- $n + k 2nk = 1 \Longrightarrow |w|_{min} = 2 \Longrightarrow w = ba \cup ab$
- $n, k = 1 \Longrightarrow |w|_{min} = 4 \Longrightarrow w = baba, \qquad L \subseteq \{xx^R | x \in \Sigma^*\}$
- Για μεγαλύτερα n, k έχουμε συμβολοσειρές όπως bbabaa, baabba, bbaabbaa...

Από αυτά κάποια δεν μας λένε κάτι ιδιαίτερα χρήσιμο, άλλα όμως μας φανερώνουν την οδό που θα πρέπει να ακολουθήσουμε. Για παράδειγμα, κατά αντιστοιχία με τις αντίστοιχες τεχνικές που ακολουθήσαμε στις λύσεις ασκήσεων κανονικών εκφράσεων, τα διάφορα ελάχιστα μας δίνουν την βάση πάνω στην οποία θα χτιστεί το (PD) αυτόματο.

 $\Omega$ ς εχ τούτου έχουμε φαινομενικά τέσσερεις ελάχιστες περιπτώσεις, που η συγκεκριμένη γλώσσα επιτρέπει:  $w_0 = \varepsilon \lor w_1 = ab \lor w_2 = ba \lor w_3 = baba$  αλλά προσοχή, το w = baba αποτελεί παρεμβολή του  $w_1$  εντός του  $w_2$  και όχι  $(ba)^*$ 

Αυτό με τη σειρά του δείχνει ότι ουσιαστικά, το  $w_3$  αποτελεί επέκταση της  $w_2$  μετά αρχικού b και άρα δεν είναι μέρος της καθολικής βάσης του αυτομάτου. Θα μας χρειαστεί όμως, αφού δείχνει ότι όταν ο αλγόριθμος θα μπαίνει στη διαδρομή που διαβάζει ba, θα διακλαδώνεται σε δύο εναλλακτικές βάση του αν αυτό που έπεται του b είναι a ή aba.

Τέλος αφού  $n, k \in \mathbb{N}_{r}$  τότε μπορούν όλοι αυτοί οι χαρακτήρες να είναι (μετρήσιμα) άπειροι αλλά υπό τον όρο ότι τηρείται το γενικό μοτίβο.

Άρα, υποθετικά, έχουμε τρεις αρχικούς κλάδους και σε έναν από αυτούς μία διάσπαση σε δύο. Θα δούμε ότι αχριβώς αυτό συμβαίνει και στην πράξη (τουλάχιστον στη συγκεκριμένη επίλυση) και ότι γενικά είναι καλή πρακτική. Ξεκινάμε κατασκευή αυτομάτου, αρχικά δίνοντας μαθηματική περιγραφή και κατόπιν αντίστοιχο διάγραμμα:

#### **PDA** $M = (K, \Sigma, \Gamma, \Delta, s, F)$

- $K = \{q1, q2, q3, q4, q5, q6\}$
- $\Sigma = \{a, b\}$
- $\Gamma = \{A, B, \$\}$
- ∆ =

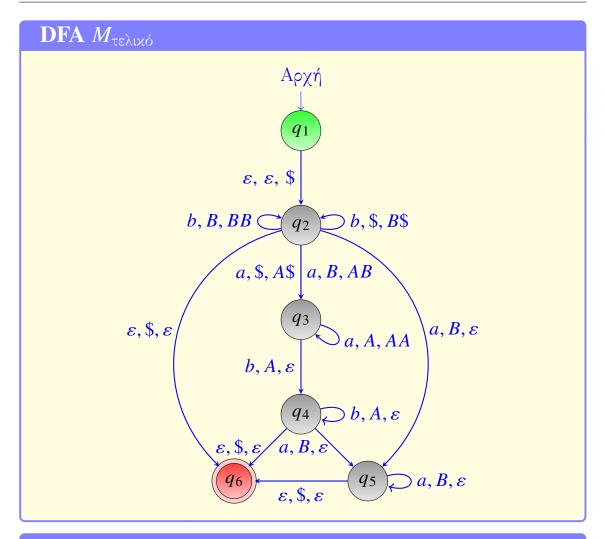
  - 1.  $(q1, \varepsilon, \varepsilon) \rightarrow (q2, \$)$  8.  $(q3, a, A) \rightarrow (q3, AA)$
  - 2.  $(q2, b, \$) \rightarrow (q2, B\$)$  9.  $(q3, b, A) \rightarrow (q4, \varepsilon)$
  - 3.  $(q2, b, B) \rightarrow (q2, BB)$  10.  $(q4, b, A) \rightarrow (q1, \varepsilon)$

  - 4.  $(q2, \varepsilon, \$) \rightarrow (q6, \varepsilon)$  11.  $(q4, a, B) \rightarrow (q5, \varepsilon)$

  - 5.  $(q2, a, \$) \to (q3, A\$)$  12.  $(q4, \varepsilon, \$) \to (q6, \varepsilon)$

  - 6.  $(q2, a, B) \rightarrow (q3, AB)$  13.  $(q5, a, B) \rightarrow (q5, \varepsilon)$

  - 7.  $(q2, a, B) \rightarrow (q5, \varepsilon)$  14.  $(q5, \varepsilon, \$) \rightarrow (q6, \varepsilon)$
- $\bullet$   $s = q_1$
- $F = q_6$



# Υπολογισμός αποδοχής bbbabaaa

$$(q1,bbbabaaa, \varepsilon)$$
  $\vdash$   $(q2,bbbabaaa,\$)$   $\vdash$   $(q2,bbabaaa,B\$)$   $\vdash$   $(q3,baaa,ABBB\$)$   $\vdash$   $(q2,abaaa,BBB\$)$   $\vdash$   $(q5,baaa,BB\$)$   $\vdash$   $(q5,aa,BB\$)$   $\vdash$   $(q6,\varepsilon,\varepsilon)$   $\vdash$   $(q6,\varepsilon,\varepsilon)$   $\vdash$   $(q6,baaa,\$)$   $\vdash$   $(q2,babaaa,BB\$)$   $\vdash$   $(q3,baaa,BB\$)$   $\vdash$   $(q5,a,B\$)$   $\vdash$   $(q5,\varepsilon,\$)$ 

#### 2 - β Απάντηση Υποερωτήματος (β)

Έχουμε να κατασκευάσουμε PDA για CFL  $L_2 = \{w \in \{a, b\}^* : |w|_b = 2|w|_a\}$ Από τον τύπο δυνάμεθα να εξάγουμε τα εξής δεδομένα, για συμβολοσειρά  $w \in L_1$ :

- $|w|_{min} = 0 \Longrightarrow w_{min} = \varepsilon$ ,  $|w|_{max} = \aleph_0$ ,  $|w| \equiv |w|_a \pmod{2}$
- $\{w \in \{a, b\}^+ \mid |w| = 3, |w|_a = 1, |w|_b = 2\} \Longrightarrow |w| = 3|w|_a$
- |w| = 3 : abb, bab, bba
- |w| > 3: aabbbb, bbbbaa, abbbba, babbab, ababbb, bbbaba, bbaabb, ...
- Κανένας περιορισμός οποιουδήποτε συμβόλου ως προς την θέση.

Έχουμε φαινομενικά τέσσερεις ελάχιστες περιπτώσεις, που η συγκεκριμένη γλώσσα επιτρέπει:  $w_0 = \varepsilon \lor w_1 = abb \lor w_2 = bab \lor w_3 = bba$  αλλά όπως ήδη είπαμε δεν υπάρχει περιορισμός ως προς την θέση αντίθετα με το προηγούμενο πρόβλημα στην άσκηση 2.1 (υποερώτημα (α)).

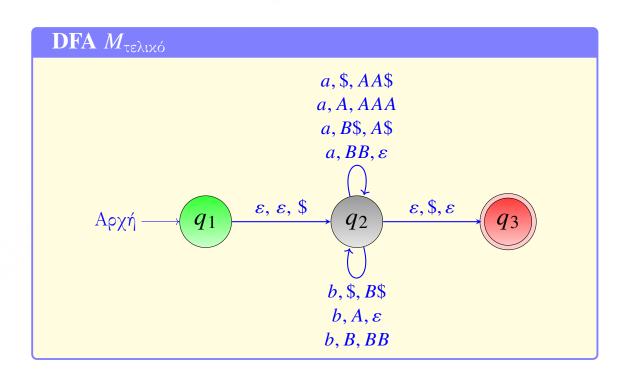
Αυτό με τη σειρά του δείχνει ότι πιθανώς να μην χρειαστεί κάποια διακλάδωση και να μπορούν να γίνουν όλες οι διαδικασίες αναδρομικά στον ίδιο κόμβο.

Άρα, υποθέτουμε ένα κόμβο για όλη την ανάγνωση βασιζόμενοι σε σύνθετες αναδρομές, συν επιπρόσθετους βοηθητικούς κόμβους (Εναρκτήριο με μετάβαση αρχικοποίησης σωρού με δείκτη τελευταίου κελιού και τελικό με μετάβαση προς αυτόν όπου αδειάζει τον δείκτη από τον σωρό). Ξεκινάμε την κατασκευή του αυτομάτου, αρχικά δίνοντας την μαθηματική περιγραφή και κατόπιν το αντίστοιχο διάγραμμα:

# **PDA** $M = (K, \Sigma, \Gamma, \Delta, s, F)$

- $K = \{q1, q2, q3\}$
- $\Sigma = \{a, b\}$
- $\Gamma = \{A, B, \$\}$
- Δ =
  - 1.  $(q1, \varepsilon, \varepsilon) \rightarrow (q2, \$)$
- 6.  $(q2, a, A) \rightarrow (q2, AAA)$
- 2.  $(q2, b, \$) \rightarrow (q2, B\$)$
- 7.  $(q2, a, BB) \rightarrow (q2, \varepsilon)$
- 3.  $(q2, b, B) \rightarrow (q2, BB)$
- 4.  $(q2, b, A) \rightarrow (q2, \varepsilon)$
- 8.  $(q2, a, B\$) \rightarrow (q2, A\$)$
- 5.  $(q2, a, \$) \rightarrow (q2, AA\$)$
- 9.  $(q2, \varepsilon, \$) \rightarrow (q3, \varepsilon)$

- $s = q_1$
- $F = q_3$



# Υπολογισμός αποδοχής *abbbab*

$$(q1,abbbab,\varepsilon)$$
  $\vdash$   $(q2,abbbab,\$)$   $\vdash$   $(q2,bbbab,AA\$)$   $\vdash$   $(q2,bbab,A\$)$   $\vdash$   $(q2,bab,\$)$   $\vdash$   $(q2,ab,B\$)$   $\vdash$   $(q2,b,A\$)$   $\vdash$   $(q2,\varepsilon,\$)$   $\vdash$   $(q3,\varepsilon,\varepsilon)$  Επιτυχής Ανάγνωση

# 3 Άσχηση 3η - Γλώσσες χωρίς συμφραζόμενα

(α) [5%] Αποφανθείτε αν ο παρακάτω ισχυρισμός είναι σωστός ή λανθασμένος και αιτιολογήστε:

Η τομή μίας ντετερμινιστικής γλώσσας χωρίς συμφραζόμενα με μία πεπερασμένη γλώσσα είναι πάντα γλώσσα χωρίς συμφραζόμενα.

(β) [5%] Αποδείξτε εάν η παρακάτω γλώσσα είναι ή δεν είναι γλώσσα χωρίς συμφραζόμενα:

$$L = \{a^m b^n c^k : n, k, m \in \mathbb{N}_0, n = 3m + 2k\}$$

#### 3 - α Απάντηση Υποερωτήματος (α)

Η πρόταση "η τομή μίας ντετερμινιστικής γλώσσας χωρίς συμφραζόμενα με μία πεπερασμένη γλώσσα είναι πάντα γλώσσα χωρίς συμφραζόμενα" είναι αληθής όπως θα αποδείξουμε παρακάτω:

#### Απόδειξη

• 
$$\mathcal{P} := \{ \forall L_{DCF} \in \mathcal{L}_{DCF}, \ \forall L_{REG} \in \mathcal{L}_{REG} : L_{DCF} \cap L_{REG} \in \mathcal{L}_{CF} \}$$
 (1)

• 
$$\mathcal{L}_{FIN} \subset \mathcal{L}_{REG} \subset \mathcal{L}_{DCF} \subset \mathcal{L}_{NCF} = \mathcal{L}_{CF}$$
 (2)

• 
$$\forall (A, B) \ni A \subset B \to A \cap B = A, A \subset B$$
 (3)

• 
$$A \subset B \subset ... \subset N \Rightarrow A \subset N$$
 (4)

• 
$$\stackrel{(2)(4)}{\Longrightarrow} \mathcal{L}_{FIN} \subset \mathcal{L}_{DCF}$$
 (5)

• 
$$\mathcal{L}_{FIN} \cap \mathcal{L}_{DCF} \stackrel{(3)(5)}{=} \mathcal{L}_{FIN} \subset \mathcal{L}_{DCF} \subset \mathcal{L}_{CF} \stackrel{(4)}{\Rightarrow} \mathcal{L}_{FIN} \subset \mathcal{L}_{CF} \Rightarrow \mathcal{L}_{FIN} \in \mathcal{L}_{CF}$$

**(6)** 

• 
$$Q := \{ \forall L_{DCF} \in \mathcal{L}_{DCF}, \ \forall L_{REG} \in \mathcal{L}_{REG} : L_{DCF} \cap L_{REG} \in \mathcal{L}_{CF} \}$$
 (7)

• 
$$\mathcal{P}^{(1)} = Q^{(7)} = \text{True} \Rightarrow \mathcal{P} = \text{True}$$

Για να εξηγήσω την λογική, δείξαμε ότι:

- Τομή μεταξύ ενός συνόλου με ενός υποσυνόλου (γνησίου ή μη) παράγει μόνο το ίδιο το υποσύνολο. Αποδειχνύεται μέσω της απόδειξης για του νόμου της απορρόφησης μεταξύ υποσυνόλου και συνόλου. Θα παραθέσω τις όποιες αποδείξεις παρακάτω.
- 2. Το χρησιμοποιήσαμε για να δείξουμε ότι τομή πεπερασμένων γλωσσών και ντετερμινιστικών χωρίς συμφραζόμενα γλωσσών, παράγει αποκλειστικά πεπερασμένες γλώσσες. Ο λόγος είναι ότι οι πεπερασμένες είναι υποσύνολο των DCFL.
- 3. Κατόπιν δείξαμε ότι λόγο του ότι αν έχουμε μία αλυσίδα συνόλων με κάθε επόμενο να είναι υπερσύνολο του κάθε προηγούμενου (έστω και γνήσιο) τότε μπορούμε να πούμε ότι οποιοδήποτε μικρότερο σύνολο είναι εντός οποιουδήποτε μεγαλύτερου. Αποδεικνύεται, διαισθητικά, μέσω του ιδίου παίρνοντας οποιοδήποτε στοιχείο από το οποιοδήποτε μικρότερο σύνολο και δείχνοντας ότι ανήκει σε όλα τα υπερσύνολα.
- 4. Τα συνδυάσαμε συλλογιζόμενοι ότι τομή πεπερασμένης με DCFL παράγει πεπερασμένη και αφού η πεπερασμένη είναι υποσύνολο κανονικής, που είναι υποσύνολο DCFL, που είναι υποσύνολο CFL, τότε η πεπερασμένη γλώσσα που παίρνουμε ως αποτέλεσμα τομής πεπερασμένης με DCFL, είναι επίσης υποσύνολο CFL γλωσσών και άρα κάθε στοιχείο της είναι και στοιχείο των CFL γλωσσών.
- 5. Άρα η τομή πεπερασμένης γλώσσας και ντετερμινιστικής γλώσσας χωρίς συμφραζόμενα παράγει πάντα πεπερασμένη γλώσσα και αφού οι πεπερασμένες γλώσσες είναι υποσύνολα των γλωσσών χωρίς συμφραζόμενα, τότε οποιαδήποτε πεπερασμένη γλώσσα, συμπεριλαμβανομένης και αυτής που παράγει τομή πεπερασμένης με DCFL, ανήκει στις γλώσσες χωρίς συμφραζόμενα.

Απόδειξη  $A \subset B \to A \cap B \Rightarrow A$  και  $A \subset B \subset C \Rightarrow A \subset C$ :

1. Ορισμός γνήσιου υποσυνόλου:

$$A \subset B \stackrel{def}{=} \left[ \forall x (x \in A \Rightarrow x \in B) \right] \wedge \left[ \exists y (y \in B \wedge y \notin A) \right]$$

2. Απόδειξη  $A \subset B \subset C \Rightarrow A \subset C$ :

$$B \subset C = \forall x (x \in B \Rightarrow x \in C), \qquad A \subset B = \forall x (x \in A \Rightarrow x \in B)$$

$$A \subset B \subset C = \left[ \forall x (x \in A \Rightarrow x \in B \Rightarrow x \in C) \right] \land \left[ \exists y (y \in C \land y \notin B \Rightarrow y \notin A) \right] \Rightarrow$$

$$\Rightarrow \left[ \forall x (x \in A \Rightarrow x \in C) \right] \land \left[ \exists y (y \in C \land y \notin A) \right] = A \subset C$$

3. Ορισμός τομής:

$$A \cap B \stackrel{def}{=} \{x \mid x \in A, x \in B\}$$

4. Απόδειξη  $A \subset B \rightarrow A \cap B = A$ :

$$A \cap B = C \longrightarrow \forall x \big[ (x \in A \land x \in B) \Leftrightarrow x \in C \big]$$
 αλλά  $A \subset B \longrightarrow \forall x (x \in A \Rightarrow x \in B)$  οπότε στη συγκεκριμένη περίπτωση (και όλες τις αντίστοιχες) έγουμε:

$$A \cap B = C \to \left[ \forall x (x \in A \Rightarrow x \in B \Rightarrow x \in C) \right] \land \left[ \exists y (y \notin A \land y \in B \Rightarrow y \notin C) \right] \Rightarrow$$
$$\left[ \forall x (x \in A \Rightarrow x \in C) \right] \land \left[ \forall y (y \notin A \Rightarrow y \notin C) \right] \Rightarrow A = C \Rightarrow A \cap B = A$$

#### 3 - β Απάντηση Υποερωτήματος (β)

Ζητείται να αποδείξουμε άν η  $L=\{a^mb^nc^k\ :\ n,\ k,\ m\in\mathbb{N}_0,\ n=3m+2k\}$  είναι CF.

Αυτό μπορεί να γίνει μεταξύ άλλων με απόπειρα κατασκευής CFG της γλώσσαςL, ενώ μία άλλη μέθοδος είναι μέσω λήμματος άντλησης κα.

#### Απόδειξη με απόπειρα κατασκευής CFG

- Προτεραιότητα συμβόλων:  $w = \{xyz \mid x = \{a\}^m, y = \{b\}^{3m+2k}, z = \{c\}^k\}.$
- Λόγο συγκεκριμένης προτεραιότητας τα σύμβολα δεν αναμιγνύονται πχ δεν υπάρχει bbabbcb παρόλο που η αναλογία συμβόλων είναι ορθή). Ακριανά σύμβολα (a, c) καθορίζουν το πλήθος των μεσαίων (b). Άρα μπορούμε να σπάσουμε όλες τις συμβολοσειρές της γλώσσας σε δύο μέρη: ένα αριστερό:

$$L_{LHS} = \{ w = \{ a^m b^{3m} \}^* : m \in \mathbb{N}_0 \}$$

και ένα δεξιό:

$$L_{RHS} = \{ w = \{ b^{2k} c^k \}^* : k \in \mathbb{N}_0 \}$$

Η λύση θα είναι η σύνθεση αυτών τον δύο και συγκεκριμένα:

$$\{w \in L, w_1 \in L_{LHS}, w_2 \in L_{RHS} \mid w = w_1 w_2\}$$

- Fix  $L_{LHS}$  :  $S \to \varepsilon \mid A$ ,  $A \to \varepsilon \mid aAbbb$
- Fix  $L_{RHS}: S \to \varepsilon \mid C, \quad C \to \varepsilon \mid bbCc$
- 'Apa :  $S \to \varepsilon \mid AC$ ,  $A \to \varepsilon \mid aAbbb$ ,  $C \to \varepsilon \mid bbCc$
- $G = (\{a, b, c\}, \{\epsilon\}, \{S \to \epsilon \mid AC, A \to \epsilon \mid aAbbb, C \to \epsilon \mid bbCc\}, S)$

4 Άσχηση 4η - Αναγνώριση γλωσσών χωρίς συμφραζόμενα:

[30%] Έστω η γραμματική χωρίς συμφραζόμενα

$$G=(V, \Sigma, R, S), \text{ όπου } V=\{S, A, L, D, a, b\}, \Sigma=\{a, b\} \text{ και}$$
 
$$R=\{S\rightarrow A, S\rightarrow L, A\rightarrow LbADa, A\rightarrow LaD, L\rightarrow Da, L\rightarrow e, D\rightarrow b\}.$$

- (α) Μετατρέψτε τη γραμματική G σε κανονική μορφή Chomsky.
- (β) Εφαρμόστε τον αλγόριθμο δυναμικού προγραμματισμού για συντακτική ανάλυση στη συμβολοσειρά w=babba. Δώστε τον πλήρη πίνακα N καθώς και όλα τα συντακτικά δέντρα της w.

#### 4 - α Απάντηση Υποερωτήματος (α)

Ζητούμενο η μετατροπή σε κανονική μορφή Chomsky (CNF) CFG's:

$$G=(V, \Sigma, R, S)$$
, όπου  $V=\{S, A, L, D, a, b\}$ ,  $\Sigma=\{a, b\}$  και 
$$R=\{S\rightarrow A, S\rightarrow L, A\rightarrow LbADa, A\rightarrow LaD, L\rightarrow Da, L\rightarrow e, D\rightarrow b\}.$$

Καταρχήν να θυμηθούμε τον ορισμό του τί εννοούμε "κανονική μορφή Chomsky":

Ξεκινώντας να αναφέρω ότι, ότι γράφουμε από εδώ και έπειτα εκτός την δοθέντα
 CFG, θα είναι με βάση της εναλλακτικής μαθηματικής περιγραφής CFG / CNF που θεωρεί ότι το V ως σύνολο μη τερματικών συμβόλων.

 $\Delta$ ηλαδή  $V \cap \Sigma = \emptyset$ , όπως περιγράφεται σε εξίσου μεγάλο μέρος της βιβλιογραφίας (πχ Sipser). Υπό αυτό το πρίσμα το V μεταφράζεται ως "(V)ariables" σε αντίθεση με όταν ισχύει  $V \cap \Sigma = \Sigma$  όπου μεταφράζεται ως "(V)ocabulary". Αυτό διαφέρει από την περιγραφή που δίδεται στο μάθημα, αλλά δεν είναι λιγότερο ή περισσότερο ορθή, απλά διαφορετική.

- $A, B, C \in V$
- $a \in \Sigma$
- Επιτρέπονται μόνο οι παρακάτω τύπου παραγωγές:
  - 1.  $A \rightarrow BC$
  - $2. A \rightarrow a$
  - 3. Σε ειδικές περιπτώσεις απαιτείται δημιουργία νέου εναρκτήριου κανόνα  $\pi \text{αραγωγής } S_0 \to S$
  - 4.  $\varepsilon \in L(G) \Leftrightarrow S \to \varepsilon$  ή  $S_0 \to S \mid \varepsilon$  εάν το S εμφανίζεται στην παραγωγή χάποιου χανόνα.

# Μετατροπή $CFG \rightarrow CNF$ 1/3

- Εισαγωγή νέας κατάστασης
  - 1. Στη συγκεκριμένη γραμματική ισχύει  $S\in S, S\to L\to \varepsilon\Leftrightarrow \varepsilon\in L(G)$  αλλά το S δεν υπάρχει στη δεξιά μεριά κανενός κανόνα παραγωγής και άρα δεν απαιτείτε κατασκευή νέου κανόνα παραγωγής  $S_0\to S\,|\,\varepsilon.$
- Απαλοιφή κενής παραγωγής

1. 
$$S \rightarrow A \mid L$$

2. 
$$A \rightarrow LbADa \mid LaD$$

3. 
$$L \rightarrow Da \mid \varepsilon$$

4. 
$$D \rightarrow b$$

1. 
$$S \rightarrow A \mid L \mid \varepsilon$$

2. 
$$A \rightarrow LbADa \mid LaD \mid$$

$$bADa \mid aD$$

3. 
$$L \rightarrow Da$$

4. 
$$D \rightarrow b$$

• Απαλοιφή μοναδιαίων παραγωγών (μη τερματικών συμβόλων)

1. 
$$S \rightarrow A \mid L \mid \varepsilon$$

2. 
$$A \rightarrow LbADa \mid LaD \mid$$

$$bADa \mid aD$$

3. 
$$L \rightarrow Da$$

4. 
$$D \rightarrow b$$

1. 
$$S \rightarrow LbADa \mid LaD \mid bADa \mid$$

$$aD \mid Da \mid \varepsilon$$

2. 
$$A \rightarrow LbADa \mid LaD \mid bADa \mid$$

3. 
$$L \rightarrow Da$$

4. 
$$D \rightarrow b$$

- Απαλοιφή μη προσβάσιμων συμβόλων
  - 1. Όλα τα σύμβολα είναι προσβάσιμα  $\to$  καμία αλλαγή.

# Μετατροπή $CFG \rightarrow CNF$ 2/3

• Απομόνωση τερματικών συμβόλων

1. 
$$S \rightarrow LbADa \mid LaD \mid bADa \mid$$
  
 $aD \mid Da \mid \varepsilon$ 

- 2.  $A \rightarrow LbADa \mid LaD \mid bADa \mid$  aD
- 3.  $L \rightarrow Da$
- 4.  $D \rightarrow b$

- 1.  $S \rightarrow LDADD_a \mid LD_aD \mid$  $DADD_a \mid D_aD \mid DD_a \mid \varepsilon$
- 2.  $A \rightarrow LDADD_a \mid LD_aD \mid$  $DADD_a \mid D_aD$
- 3.  $L \rightarrow DD_a$
- 4.  $D \rightarrow b$
- 5.  $D_a \rightarrow a$
- Απαλοιφή τριαδικών+ παραγωγών (1ο πέρασμα)

1. 
$$S \rightarrow LDADD_a \mid LD_aD \mid$$

$$DADD_a \mid D_aD \mid DD_a \mid \varepsilon$$

- 2.  $A \rightarrow LDADD_a \mid LD_aD \mid$   $DADD_a \mid D_aD$
- 3.  $L \rightarrow DD_a$
- 4.  $D \rightarrow b$
- 5.  $D_a \rightarrow a$

- 1.  $S \rightarrow LX_1X_2 \mid LX_3 \mid$   $X_1X_2 \mid D_aD \mid DD_a \mid \varepsilon$ 
  - 2.  $A \rightarrow LX_1X_2 \mid LX_3 \mid$  $X_1X_2 \mid D_aD$
  - 3.  $L \rightarrow DD_a$
  - 4.  $D \rightarrow b$
  - 5.  $D_a \rightarrow a$
  - 6.  $X_1 \rightarrow DA$
  - 7.  $X_2 \rightarrow DD_a$
  - 8.  $X_3 \rightarrow D_a D$

## Μετατροπή $CFG \rightarrow CNF$ 3/3

• Απαλοιφή τριαδικών+ παραγωγών (2ο πέρασμα)

$$1. S \rightarrow LX_1X_2 \mid LX_3 \mid$$

$$X_1X_2 \mid D_aD \mid DD_a \mid \varepsilon$$

1. 
$$S \rightarrow LX_4 \mid LX_3 \mid$$
  
 $X_1X_2 \mid D_aD \mid DD_a \mid \varepsilon$ 

2. 
$$A \rightarrow LX_1X_2 \mid LX_3 \mid$$

$$X_1X_2 \mid D_aD$$

3. 
$$L \rightarrow DD_a$$

4. 
$$D \rightarrow b$$

5. 
$$D_a \rightarrow a$$

6. 
$$X_1 \rightarrow DA$$

7. 
$$X_2 \rightarrow DD_a$$

8. 
$$X_3 \rightarrow D_a D$$

2. 
$$A \rightarrow LX_4 \mid LX_3 \mid$$

$$X_1X_2 \mid D_aD$$

3. 
$$L \rightarrow DD_a$$

4. 
$$D \rightarrow b$$

5. 
$$D_a \rightarrow a$$

6. 
$$X_1 \rightarrow DA$$

7. 
$$X_2 \rightarrow DD_a$$

8. 
$$X_3 \rightarrow D_a D$$

9. 
$$X_4 \rightarrow X_1 X_2$$

•  $G_{CNF} = (\{S, A, L, D, D_a, X_1, X_2, X_3, X_4\}, \{a, b\},$ 

$$\{S \to \varepsilon, S \to LX_4, S \to X_1X_2, S \to LX_3, S \to D_aD, S \to DD_a, S \to DD_a\}$$

$$A \rightarrow LX_4, A \rightarrow X_1X_2, A \rightarrow LX_3, A \rightarrow D_aD,$$

$$L \rightarrow DD_a,\, D \rightarrow b,\, D_a \rightarrow a,\, X_1 \rightarrow DA,\, X_2 \rightarrow DD_a,\, X_3 \rightarrow D_aD,\, X_4 \rightarrow DD_a,\, X_5 \rightarrow DD_a,\, X_6 \rightarrow DD_a,\, X_8 \rightarrow DD_a,\, X_$$

$$X_1X_2$$
},  $S$ )

#### MΕΡΟΣ 2. ΛΎΣΕΙΣ ΑΣΚΉΣΕΩΝ

#### 4 - β Απάντηση Υποερωτήματος (β)

Χρησιμοποιούμαι το παραχάτω ως αναφορά για τους χανόνες παραγωγής χαθώς με αρίθμηση για να μπορούμε να διαχωρίσουμε αυτούς που ξεχινάνε από ίδιο μη τερματιχό σύμβολο:

1. 
$$S \rightarrow LX_4$$

10. 
$$A \rightarrow D_a D$$

2. 
$$S \rightarrow LX_3$$

11. 
$$L \rightarrow DD_a$$

3. 
$$S \rightarrow X_1 X_2$$

12. 
$$D \rightarrow b$$

4. 
$$S \rightarrow D_a D$$

13. 
$$D_a \rightarrow a$$

5. 
$$S \rightarrow DD_a$$

14. 
$$X_1 \rightarrow DA$$

6. 
$$S \rightarrow \varepsilon$$

7. 
$$A \rightarrow LX_4$$

15. 
$$X_2 \rightarrow DD_a$$

8. 
$$A \rightarrow LX_3$$

16. 
$$X_3 \rightarrow D_a D$$

9. 
$$A \rightarrow X_1X_2$$

17. 
$$X_4 \to X_1 X_2$$

# Πίνακας συντακτικής ανάλυσης για w=babba

- Κατασκευάζουμε πίνακα |w| στηλών και για κάθε στήλη  $n = \in [1, |w|_{max}]$  ξεκινώντας να μετράμε από αριστερά ως 1η, έχουμε n κελιά.
- Στο κορυφαίο κελί κάθε στήλης (δηλαδή το n), τοποθετούμε το αντίστοιχο σύμβολο της λέξης (δηλαδή το  $w_n$ ).
- Κάθε ένα από αυτά τα σύμβολα ουσιαστικά αποτελεί κανόνα παραγωγής τερματικού συμβόλου.
- Ανά στήλη, σε κελί  $k, k \in [2, n]$  με  $k_{max}$  να βρίσκεται στη βάση του πίνακα και  $k_{min}$  ακριβώς στο κελί κάτω από αυτό στην κορυφή της στήλης του, τοποθετούμε τους κανόνες παραγωγής (τα μη τερματικά σύμβολα) τα οποία δύνανται να οδηγήσουν σε αριστερότερο τμήμα της w μήκους υποσυμβολοσειράς x με  $|x| = k, x = \sum_{i=1}^k w_i$
- Όταν ολοκληρώσουμε περιμένουμε να βρούμε στη κάτω δεξιά γωνία κάποια κανόνα παραγωγής, από το αρχικό (μη τερματικό προφανώς) σύμβολο. Εάν αυτό υπάρχει τότε  $w \in G$ , ειδάλλως  $w \notin G$ .

			a				
ь				b	L[11], X <sub>2</sub> [15], S[5]		
			b	Ø	0		
		a	S[4], A[10], X <sub>3</sub> [16]	Ø	0		
	b	L[11], X <sub>2</sub> [15], S[5]	<i>X</i> <sub>1</sub> [14]	Ø	A[9], S[3], X <sub>4</sub> [17]		

Που δίνει:

# $\Delta$ έντρο συντακτικής ανάλυσης για w=babba

- Κατασκευάζουμε δέντρο συντακτικής ανάλυσης βάση αντίστροφης χρήσης
   των αποτελεσμάτων του πίνακα, δηλαδή ξεκινώντας από την κάτω δεξιά
   γωνία και συγκεκριμένα την αντίστοιχη εκδοχή αρχικού σύμβολου εφόσον
   η w όντως ανήκει στη γραμματική μας.
- Σε κάθε βήμα ακολουθούμε κλάδο βάση κανόνων παραγωγής αυτής της εκδοχής συμβόλου και ακολουθείτε από την εκάστοτε εκδοχή επόμενων με τερματικών συμβόλων που έχουν βρεθεί στο αντίστοιχο βήμα στο πίνακα, έως να φτάσουμε διαδοχικά στα κατάλληλα τερματικά σύμβολα που κατασκευάζουν τη λέξη που μας δόθηκε.

