



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
TECHNICAL UNIVERSITY OF CRETE

ΉΜΜΥ

ΠΛΗ 402 - Θεωρία Υπολογισμού

3η Σειρά Ασκήσεων

Συγγραφέας:

Βασιλειάδης Σταύρος

ΑΜ:

2019030023

Διδάσκων:

Λαγουδάκης Μιχαήλ

30 Ιουνίου 2025

Περιεχόμενα

Περιεχόμενα	2
1 Κατάλογος συμβόλων και συντομογραφίες	i
2 Λύσεις Ασκήσεων	1
1 Άσκηση 1η - Μηχανές Turing:	2
1 - α Απάντηση Υποερωτήματος (α)	3
1 - β Απάντηση Υποερωτήματος (β)	8
2 Άσκηση 2η - Αυτόματα στοίβας:	17
2 - α Απάντηση Υποερωτήματος (β)	18
2 - β Απάντηση Υποερωτήματος (β)	21
3 Άσκηση 3η - Γλώσσες χωρίς συμφραζόμενα	24
3 - α Απάντηση Υποερωτήματος (α)	25
3 - β Απάντηση Υποερωτήματος (β)	28
4 Άσκηση 4η - Αναγνώριση γλωσσών χωρίς συμφραζόμενα:	29
4 - α Απάντηση Υποερωτήματος (α)	30
4 - β Απάντηση Υποερωτήματος (β)	34

ΜΕΡΟΣ 1

Κατάλογος συμβόλων και συντομογραφίες

ΜΕΡΟΣ 1. ΚΑΤΑΛΟΓΟΣ ΣΥΜΒΟΛΩΝ ΚΑΙ ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

- \mathcal{L} 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
- $\sim M$ Ισοδυναμία DFA κατά Myhill-Nerode
- Σ Αλφάβητο
- Q ή K Σύνολο καταστάσεων (αυτομάτου)
- s Αρχική κατάσταση (αυτομάτου)
- F Σύνολο τελικών καταστάσεων (αυτομάτου)
- Δ Σύνολο σχέσεων μεταβάσεων (αυτομάτου)
- w, x, y, z Συμβολοσειρά / Υποσυμβολοσειρές

- *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
- *M* Αυτόματο
- *M'* Πρότυπο/Ελάχιστο Αυτόματο
Εναλλακτικά *DFA* παραγόμενο ενός άλλου *DFA* / *NFA*.
- \mathbb{N}^0 Σύνολο φυσικών αριθμών (συμπεριλαμβανομένου και του μηδενός)
- \aleph_0 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 e Κενό σύμβολο

ΜΕΡΟΣ 1. ΚΑΤΑΛΟΓΟΣ ΣΥΜΒΟΛΩΝ ΚΑΙ ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

- \exists Υπάρχει
- \forall Για όλα
- \in Ανήκει στο
- \dots "και ούτω καθεξής", συνέχιση ακολουθίας βάσει εμφανούς μοτίβου
- $|$ ή $:$ ή \ni Έτσι Όστε
- $\mathcal{P}(Q)$ Δυναμοσύνολο συνόλου Q .
Για $Q = \{a, b, c\} \xrightarrow{\emptyset \subseteq \text{all sets}} \mathcal{P}(Q) = \{\emptyset, a, b, c, ab, ac, bc, abc\}$
Συνεπάγεται ότι $|\mathcal{P}(Q)| = 2^{|Q|}$
- \mathcal{P} ή Q Λογικές προτάσεις
- T / F (Λογικό) Αληθές / Ψευδές
- \cup Ένωση
- \cap Τομή
- $-$ ή $/$ Διαφορά
- \subseteq Υποσύνολο
- \subset Γνήσιο υποσύνολο
- \wedge Λογική Σύζευξη
- \vee Λογική Διάζευξη
- \neg Λογική Άρνηση
- Σ^* Σύνολο πεπερασμένου μήκους συμβολοσειρών αλφάβητου Σ
συμπεριλαμβανομένης της κενής συμβολοσειράς ε

ΜΕΡΟΣ 2

Λύσεις Ασκήσεων

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:

$$(\alpha) w \in \{a, b\}^* : \triangleright \sqsubseteq w \sqsubseteq \rightarrow \triangleright \sqsubseteq w^R w \sqsubseteq$$

$$(\beta) x, y, z \in \{0, 1\}^*, |x| = |y| = |z| = |w|, z = x \bar{\wedge} y, w = x \wedge y :$$

$$\triangleright \sqsubseteq x \sqsubseteq y \sqsubseteq \rightarrow \triangleright \sqsubseteq z \sqsubseteq w \sqsubseteq$$

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 ίδιας 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 σκεφθήκαμε εν προκειμένω για την επίλυση του συγκεκριμένου 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.

- Οι επιμέρους μηχανές υποθετικά έχουν ως εξής (αργότερα ίσως δούμε ότι ίσως χρειαζόμαστε 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 : Μετακίνηση στη τελική θέση που απαιτούν οι προδιαγραφές να βρεθεί η κεφαλή 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.

$$\text{Έλεγχος κενής λέξης : } M_1 : \quad M_4 \xleftarrow{\sqcup} \mathbf{R} \xrightarrow{x \in \{a,b\}} M_2$$

Δοκιμή:

- $|w| = 0 \Rightarrow \triangleright \sqcup \sqcup \xrightarrow{R} \triangleright \sqcup \sqcup \xrightarrow{\sqcup} M_4 \quad \checkmark$
- $|w| = n, n \geq 1 \Rightarrow \triangleright \sqcup w_1 \dots w_n \sqcup \xrightarrow{R} \triangleright \sqcup \underline{w_1} \dots w_n \sqcup \xrightarrow{x \in \{a,b\}} M_2 \quad \checkmark$

ΜΕΡΟΣ 2. ΑΥΤΣΕΙΣ ΑΣΚΗΣΕΩΝ

Η μηχανή M_2 έχει ως αποστολή την αντιγραφή συμβολοσειράς w δεξιά της ίδιας 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.

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

Δοκιμή:

$$\begin{aligned} \bullet \quad w = abb &\xrightarrow{M_1} \triangleright \sqcup \underline{a} b b \sqcup \xrightarrow{\$} \triangleright \sqcup \underline{\$} b b \sqcup \xrightarrow{R_{\sqcup}^2} \triangleright \sqcup \$ b b \sqcup \sqcup \xrightarrow{x} \triangleright \sqcup \$ b b \sqcup \underline{a} \\ &\xrightarrow{L_{\$}} \triangleright \sqcup \underline{\$} b b \sqcup a \xrightarrow{R} \triangleright \sqcup \$ \underline{b} b \sqcup a \xrightarrow{x \in \{a, b\}} M_2 \xrightarrow{\$ R_{\sqcup}^2 x L_{\$} R} \triangleright \sqcup \$ \$ \underline{b} \sqcup a b \\ &\xrightarrow{x \in \{a, b\}} M_2 \xrightarrow{\$ R_{\sqcup}^2 x L_{\$} R} \triangleright \sqcup \$ \$ \$ \underline{\sqcup} a b b \xrightarrow{\sqcup} M_3 \quad \checkmark \end{aligned}$$

Ακολουθεί μηχανή M_3 οποία έχει ως σκοπό αναστροφή αρχικής λέξης. Αρχικά όπως το είχαμε σχεφθεί 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 τέλει 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 :

$$M_4 \xleftarrow{\sqcup} \overset{\vee}{\mathbf{R}} \xrightarrow{x \in \{a, b\}} \sqcup L x L_{\$} x R_{\sqcup} \rightarrow M_3$$

Δοκιμή:

$$\begin{aligned}
 \bullet \quad w = abb &\xrightarrow{M_2} \triangleright \sqcup \$ \$ \$ \sqcup ab \xrightarrow{R} \triangleright \sqcup \$ \$ \$ \sqcup \underline{a} b \xrightarrow{x \in \{ab\}} \sqcup \\
 &\triangleright \sqcup \$ \$ \$ \sqcup \underline{b} b \xrightarrow{L} \triangleright \sqcup \$ \$ \$ \sqcup \sqcup b \xrightarrow{x} \triangleright \sqcup \$ \$ \$ \underline{a} \sqcup b \xrightarrow{L_\$} \\
 &\triangleright \sqcup \$ \$ \$ \underline{a} \sqcup b \xrightarrow{x} \triangleright \sqcup \$ \$ \underline{a} \sqcup b \xrightarrow{R_\sqcup} \triangleright \sqcup \$ \$ a \underline{a} \sqcup b \rightarrow M_3 \\
 &\xrightarrow{R} \triangleright \sqcup \$ \$ a \sqcup \underline{b} \xrightarrow{x \in \{ab\}} \sqcup \xrightarrow{\sqcup Lx L_\$ x R_\sqcup} \triangleright \sqcup \$ \$ a \sqcup \underline{b} \\
 &\triangleright \sqcup \$ b a a b \underline{b} \rightarrow M_3 \xrightarrow{R} \triangleright \sqcup \$ b a a b \sqcup \underline{b} \xrightarrow{x \in \{ab\}} \sqcup \triangleright \sqcup \$ b a a b \sqcup \underline{\sqcup} \\
 &\xrightarrow{\sqcup Lx L_\$ x R_\sqcup} \triangleright \sqcup b b a a b b \underline{\sqcup} \rightarrow M_3 \xrightarrow{R} \triangleright \sqcup b b a a b b \sqcup \underline{\sqcup} \xrightarrow{\sqcup} M_4 \quad \checkmark
 \end{aligned}$$

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

Μετακίνηση κεφαλής στο αριστερότατο κενό : M_4 :

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

Δοκιμή:

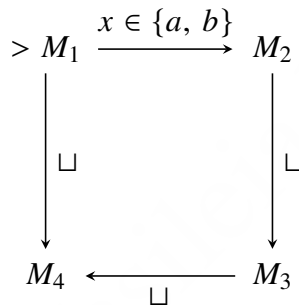
$$\begin{aligned}
 \bullet \quad w = abb &\xrightarrow{M_3} \triangleright \sqcup b b a a b b \sqcup \underline{\sqcup} \xrightarrow{L_{\triangleright}} \underline{\sqcup} \sqcup b b a a b b \sqcup \sqcup \xrightarrow{R} \\
 &\triangleright \underline{\sqcup} b b a a b b \sqcup \sqcup \quad \checkmark \\
 \bullet \quad |w| = 0 &\xrightarrow{M_1} \triangleright \sqcup \underline{\sqcup} \xrightarrow{L_{\triangleright}} \underline{\sqcup} \sqcup \sqcup \xrightarrow{R} \triangleright \underline{\sqcup} \sqcup \quad \checkmark
 \end{aligned}$$

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 \sqcup).

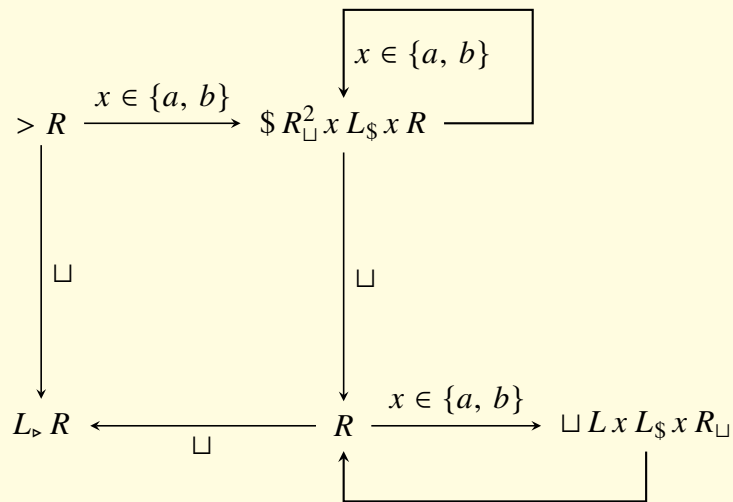
ΜΕΡΟΣ 2. ΑΥΤΣΕΙΣ ΑΣΚΗΣΕΩΝ

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.



Γραφική αναπαράσταση τελικής πρότυπης μηχανής Turing



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 τιμή).

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 **NAND** 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 πειράζει.

Έχουμε σκεφθεί 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 ακολουθηθεί τελικά, είναι να γράψουμε τα αποτελέσματα μετά το τρίτο εν σειρά κενό της ταινίας (δηλαδή πέρα της όλης αρχικής εισόδου που απαιτεί η μηχανή) και κατόπιν να τα μεταφέρουμε (ταυτόχρονα αντικαθιστώντας το σύμβολο στην θέση τους με κενό χαρακτήρα) επικαλύπτοντας τις αρχικές τιμές (σκεφτείτε το σαν *cut-paste*). Με αυτόν τον τρόπο απαλλασσόμαστε μέρους της ανάγκης εισαγωγής πολλών επιπλέον προσωρινών συμβόλων που θα εκτελούσαν χρέη δεικτών ώστε να γνωρίζουμε μέχρι που και τί έχουμε κάνει.

Η μέθοδος αυτή δεν θα προτιμηθεί διότι εάν μία τέτοια διαδικασία υλοποιηθεί, θα απαιτήσει πολλές παραπάνω και μεγαλύτερες μετακινήσεις της κεφαλής, προκαλώντας έτσι μεγαλύτερες καθυστερήσεις στην παραλαβή της εξόδου και αυξημένες φθορές στον όλο μηχανισμό (καθώς και επιπλέον 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 επιλεγεί, 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

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 κενή (ελέγχοντας αν υπάρχει έστω ένας χαρακτήρας στη λέξη x) 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. Τον λόγο για τον οποίο δεν ελέγχουμε και το y τον έχουμε ήδη αναφέρει και είναι ότι απλά, x και y έχουν ίδιο μήκος και άρα αν το ένα είναι κενό, θα είναι και το άλλο.

$$\text{Έλεγχος κενής λέξης : } M_S : \quad L \xleftarrow{\sqcup} \overset{\vee}{\mathbf{R}} \xrightarrow{0} M_{0LR} \\ \xrightarrow{1} M_{1LR}$$

Δοκιμή:

- $|x| = |y| = |z| = |w| = 0 \Rightarrow \triangleright \sqcup \sqcup \sqcup \xrightarrow{R} \triangleright \sqcup \sqcup \sqcup \xrightarrow{\sqcup} \triangleright \sqcup \sqcup \sqcup \quad \checkmark$
- $|x| = |y| = |z| = |w| = n, n \geq 1 \Rightarrow \triangleright \sqcup x_1 \dots y_n \sqcup y_1 \dots y_n \xrightarrow{R} \triangleright \sqcup x_1 \dots x_n \sqcup y_1 \dots y_n \xrightarrow{0} M_{0LR} \quad \checkmark$
 $\xrightarrow{1} M_{1LR} \quad \checkmark$

ΜΕΡΟΣ 2. ΑΎΣΣΕΙΣ ΑΣΚΗΣΕΩΝ

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

Υπολογισμός/Αντικατάσταση Αριστερά προς Δεξιά : M_{0LR} :

$$\begin{aligned} \overset{\vee}{\mathbf{T}} R_{\sqcup} &\longrightarrow \overset{\text{T,F}}{\bigcirc} R \xrightarrow{0,1} F R \xrightarrow{0} M_{0RL} \\ &\xrightarrow{1} M_{1RL} \\ &\xrightarrow{\sqcup} R_{\sqcup} \rightarrow M_T \end{aligned}$$

Δοκιμή:

- $|x| = |y| = |z| = |w| = 0 \Rightarrow \triangleright \sqcup \sqcup \sqcup \xrightarrow{R} \triangleright \sqcup \sqcup \sqcup \xrightarrow{\sqcup} M_4 \quad \checkmark$
- $|x| = |y| = |z| = |w| = n, n \geq 1 \Rightarrow \triangleright \sqcup x_1 \dots y_n \sqcup y_1 \dots y_n \sqcup \xrightarrow{R}$
 $\triangleright \sqcup \underline{x_1} \dots x_n \sqcup y_1 \dots y_n \xrightarrow{x \in \{0,1\}} M_2 \quad \checkmark$

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

Υπολογισμός/Αντικατάσταση Δεξιά προς Αριστερά : M_{0LR} :

$$\begin{aligned} \overset{\vee}{\mathbf{F}} L_{\sqcup} &\longrightarrow \overset{0,1}{\bigcirc} L \xrightarrow{F,T} R T R \xrightarrow{0} M_{0RL} \\ &\xrightarrow{1} M_{1RL} \\ &\xrightarrow{\sqcup} R_{\sqcup} \rightarrow M_T \end{aligned}$$

Δοκιμή:

- $|x| = |y| = |z| = |w| = 0 \Rightarrow \triangleright \sqcup \sqcup \sqcup \xrightarrow{R} \triangleright \sqcup \sqcup \sqcup \xrightarrow{\sqcup} M_4$ ✓
- $|x| = |y| = |z| = |w| = n, n \geq 1 \Rightarrow \triangleright \sqcup x_1 \dots y_n \sqcup y_1 \dots y_n \sqcup \xrightarrow{R}$
 $\triangleright \sqcup \underline{x_1} \dots x_n \sqcup y_1 \dots y_n \xrightarrow{x \in \{0,1\}} M_2$ ✓

- Ελάχιστη περίπτωση:

$$S \rightarrow aa$$

- Ελάχιστα b :

$$S \rightarrow aaR \mid aRa \mid Raa$$

$$R \rightarrow baaa \mid abaa \mid aaba \mid aaab \mid \varepsilon$$

- $|w| = \aleph_0$:

$$S \rightarrow aaR \mid aRa \mid Raa$$

$$R \rightarrow RbRaaa \mid aRbRaa \mid aaRbR \mid aaaRbR \mid \varepsilon$$

- Σύμπτυξη:

$$S \rightarrow RaRaR$$

$$M \rightarrow bRaaa \mid abRaa \mid aabRa \mid aaaRbR \mid \varepsilon$$

Για να καταλάβουμε για πιο λόγο έφυγε η αναδρομή πριν από το b σε όλες τις παραλλαγές του R εκτός την $aaaRbR$ και άρα να υπάρχει κάποια αιτιολόγηση θα πρέπει να δώσουμε κάποια βασικά παραδείγματα:

- $bb\ aaa\ aaa\ aa$: $S \rightarrow RaRaR \rightarrow (baaa)aRaR \rightarrow (bRaaa)aRaR \rightarrow (b(bRaaa)aaa)aRaR \rightarrow (b(b\epsilon aaa)aaa)aRaR \rightarrow (bb\ aaa\ aaa)a\epsilon a\epsilon \rightarrow bb\ aaa\ aaa\ aa$
- $a\ bb\ aaa\ aaa\ a$: $S \rightarrow RaRaR \rightarrow \epsilon aRa\epsilon \rightarrow a(bRaaa)a \rightarrow a(b(bRaaa)aaa)a \rightarrow a(b(b\epsilon aaa)aaa)a \rightarrow a\ bb\ aaa\ aaa\ a$
- $aa\ bbaaa\ aaa$: $S \rightarrow RaRaR \rightarrow \epsilon a\epsilon aR \rightarrow aa(bRaaa) \rightarrow aa(b(bRaaa)aaa) \rightarrow aa(b(b\epsilon aaa)aaa) \rightarrow aa\ bb\ aaa\ aaa$
- ...
- $aa\ aaa\ bb\ aaa$: $S \rightarrow RaRaR \rightarrow \epsilon a\epsilon aR \rightarrow aa(aaabR) \rightarrow aa(aaab(bRaaa)) \rightarrow aa(aaab(b\epsilon aaa)) \rightarrow aa\ aaa\ bb\ aaa$
- ...
- $aaa\ aaa\ bb\ aa$: $S \rightarrow RaRaR \rightarrow Ra\epsilon a\epsilon \rightarrow (aaab)aa \rightarrow (aaaRb)aa \rightarrow (aaa(aaabR)b)aa \rightarrow (aaa(aaab\epsilon b)b)aa \rightarrow aaa\ aaa\ bb\ aa$
- Χρειαζόμαστε $aaaRb$ και $aaabR$. Θυμόμαστε ότι R είναι αναδρομή στον ίδιο κανόνα. Παρατηρούμε ότι και στα δύο, είναι ίδια συμβολοσειρά, με αναδρομή πριν ή μετά το b και αντικαθιστούμε με $aaaRbR$.
- Για να σιγουρευτούμε, ότι δεν χρειάζονται κανόνες τύπου "... RbR ...", έχουμε αναλύσει όλες τις παραλλαγές για ορθές συμβολοσειρές με έως τρία b , καθώς και κάποιες χαρακτηριστικές μη ορθές. Δεν τις συμπεριλαμβάνουμε για προφανείς λόγους. Αυτό δεν σημαίνει ότι δεν γίνεται να μας έχει ξεφύγει κάτι. Υπάρχει και καλύτερος τρόπος ναδειχθεί γιατί λειτουργεί αυτή η γραμματική αλλά είναι επίσης φλύαρη. Παρακάτω θα δείξουμε εναλλακτική γραμματική.
- Τέλος αν πούμε ότι η CFG μας δεν περιέχει αριστερή αναδρομή.

Γενικά θα πρέπει να έχουμε κατά νου ότι είναι πάρα πολύ χρήσιμο έως και απαραίτητο, εκτός των ελάχιστων περιπτώσεων βάση συνόλου ($\pi x \in \mathbb{N}_0$ ή και ελάχιστων περιπτώσεων για μη μηδενικό σύνολο (ουσιαστικά λαμβάνοντας υπόψιν αποδοχή κενής συμβολοσειράς εάν επιτρέπεται και ελάχιστης μή κενής), καλό θα είναι να ελέγχουμε και για διπλές εμφανίσεις των ελάχιστων μή κενών πx $\{\forall w \neq \varepsilon \mid w_{min} = xy, \exists w_{double} = xxyy \cup xyxy \cup xyxx \cup yxxy \cup yxyx \cup yxxx\}$

Υπόψιν ότι εκτός κάποιων πολύ απλών περιπτώσεων, στις CFL δεν μπορούμε πάντα να είμαστε σίγουροι ότι η CFG μας καλύπτει όλες τις πιθανές ορθές συμβολοσειρές της γλώσσας ή ότι είναι ελάχιστη.

Θα δώσω επίσης άλλη μία λύση εν συντομία μία και έχει ήδη δειχθεί (απλά απλοποιήθηκε):

- Η παρακάτω απλή και εύκολη γραμματική παρόλο που καλύπτει την γλώσσα, χρησιμοποιεί αριστερή αναδρομή κάτι που ενώ δεν αναφέρεται ως περιορισμός στην άσκηση, θα πρέπει να γνωρίζουμε ότι αναλόγως τα εργαλεία που θα χρησιμοποιήσουμε για την υλοποίηση, ίσως να προκύψει θέμα "συμβατότητας" (και είναι ένας λόγος που την αποφύγαμε, επιπροσθέτως στο ότι αυτή που προαναφέραμε αποτελεί ελαχιστοποίηση αυτής και άνευ αριστερής αναδρομής). Το πως καταλήξαμε σε αυτή φαίνεται στα βήματα της προηγούμενης προ σύμπτυξης:

$$G_{2b} = (\{a, b\}, \{aa, \varepsilon\}, \{S \rightarrow AaAaA, A \rightarrow Raaa \mid aRaa \mid aaRa \mid aaaR \mid \varepsilon, R \rightarrow AbA\}, S)$$

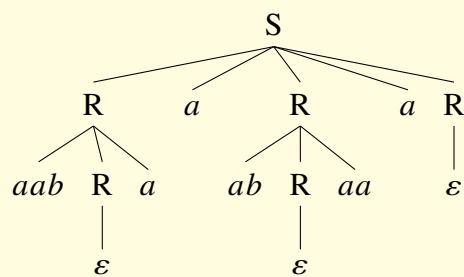
Γραμματική και συντακτικό δένδρο

Είμαστε έτοιμοι να προχωρήσουμε στην πλήρη περιγραφή της γραμματικής μας:

$$G_2 = (\{a, b\}, \{aa, \varepsilon\}, \{S \rightarrow RaRaR$$

$$R \rightarrow bRa aa \mid abRaa \mid aabRa \mid aaaRbR \mid \varepsilon\}, S)$$

Ήρθε η στιγμή να δώσουμε το συντακτικό δένδρο για συμβολοσειρά (τα κενά για ευκολότερη ανάγνωση) $w = aa \ b \ aaa \ b \ aaa$, $|w| = 10$, $|w|_a = 8$, $|w|_b = 2$:



Που δίνει: $aab \quad a \quad a \quad ab \quad aa \quad a$

Εναλλακτικά θα μπορούσαμε να το διαβάσουμε ως $(\varepsilon) \ a \ (ab(\varepsilon)aa) \ a \ (b(\varepsilon)aaa)$.

2 Άσκηση 2η - Αυτόματα στοίβας:

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

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

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

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

Από τον τύπο δυνάμεθα να εξάγουμε τα εξής δεδομένα, για συμβολοσειρά $w \in L_1$:

- $|w|_{\min} = 0 \implies n, k = 0 \implies w = \varepsilon, \quad |w|_{\max} = \aleph_0, \quad |w| = 2(n+k) = \text{Άρτιο}$
- $n+k-2nk=1 \implies |w|_{\min}=2 \implies w = ba \cup ab$
- $n, k=1 \implies |w|_{\min}=4 \implies w = baba, \quad L \subseteq \{xx^R | x \in \Sigma^*\}$
- 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.

Ως εκ τούτου έχουμε φαινομενικά τέσσερις ελάχιστες περιπτώσεις, που η συγκεκριμένη γλώσσα επιτρέπει: $w_0 = \varepsilon \vee w_1 = ab \vee w_2 = ba \vee w_3 = baba$ 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 $n, k \in \mathbb{N}_*$ τότε μπορούν όλοι αυτοί οι χαρακτήρες να είναι (μετρήσιμα) άπειροι αλλά υπό τον όρο ότι τηρείται το γενικό μοτίβο.

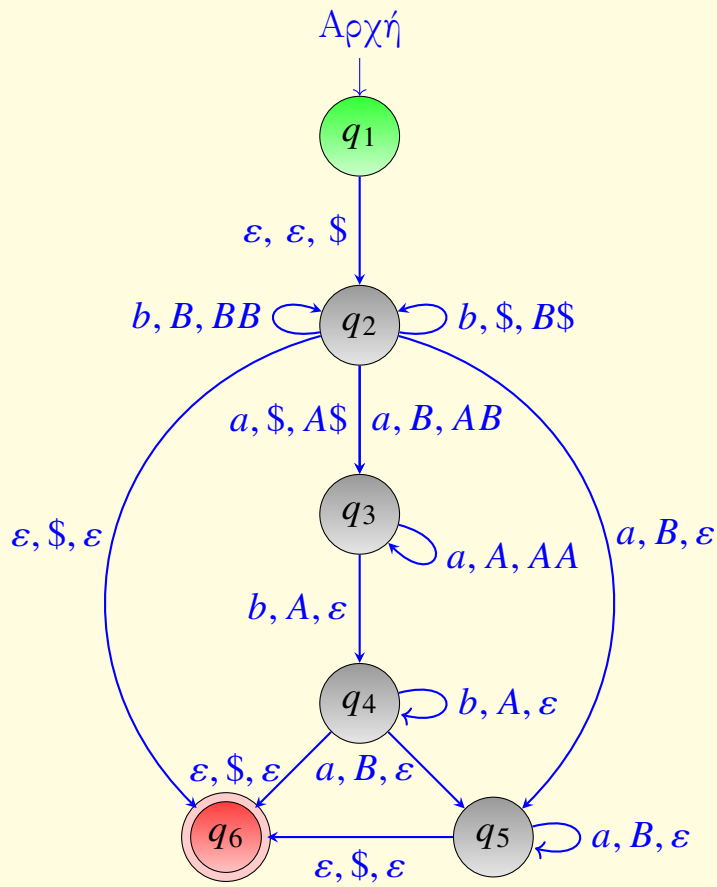
Άρα, υποθετικά, έχουμε τρεις αρχικούς κλάδους και σε έναν από αυτούς μία διάσπαση σε δύο. Θα δούμε ότι ακριβώς αυτό συμβαίνει και στην πράξη (τουλάχιστον στη συγκεκριμένη επίλυση) και ότι γενικά είναι καλή πρακτική. Ξεκινάμε κατασκευή αυτομάτου, αρχικά δίνοντας μαθηματική περιγραφή και κατόπιν αντίστοιχο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

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

- $K = \{q_1, q_2, q_3, q_4, q_5, q_6\}$
- $\Sigma = \{a, b\}$
- $\Gamma = \{A, B, \$\}$
- $\Delta =$

1. $(q_1, \varepsilon, \varepsilon) \rightarrow (q_2, \$)$	8. $(q_3, a, A) \rightarrow (q_3, AA)$
2. $(q_2, b, \$) \rightarrow (q_2, B\$)$	9. $(q_3, b, A) \rightarrow (q_4, \varepsilon)$
3. $(q_2, b, B) \rightarrow (q_2, BB)$	10. $(q_4, b, A) \rightarrow (q_1, \varepsilon)$
4. $(q_2, \varepsilon, \$) \rightarrow (q_6, \varepsilon)$	11. $(q_4, a, B) \rightarrow (q_5, \varepsilon)$
5. $(q_2, a, \$) \rightarrow (q_3, A\$)$	12. $(q_4, \varepsilon, \$) \rightarrow (q_6, \varepsilon)$
6. $(q_2, a, B) \rightarrow (q_3, AB)$	13. $(q_5, a, B) \rightarrow (q_5, \varepsilon)$
7. $(q_2, a, B) \rightarrow (q_5, \varepsilon)$	14. $(q_5, \varepsilon, \$) \rightarrow (q_6, \varepsilon)$
- $s = q_1$
- $F = q_6$

DFA $M_{\text{τελικό}}$



Υπολογισμός αποδοχής $bbbabaaa$

$(q_1, bbbabaaa, \epsilon) \vdash (q_2, bbbabaaa, \$) \vdash (q_2, bbabaaa, B\$) \vdash (q_2, babaaa, BB\$)$

$$\vdash (q_2, abaaa, BBB\$) \vdash \begin{cases} (q_3, baaa, ABBS\$) \\ (q_5, baaa, BB\$) \neq \text{Αποτυχία Κλώνου} \end{cases}$$

$\vdash (q_4, aaa, BBB\$) \vdash (q_5, aa, BB\$) \vdash (q_5, a, B\$) \vdash (q_5, \epsilon, \$)$

$\vdash (q_6, \epsilon, \epsilon)$ Επιτυχής Ανάγνωση

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 $w \in L_1$:

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

Έχουμε φαινομενικά τέσσερεις ελάχιστες περιπτώσεις, που η συγκεκριμένη γλώσσα επιτρέπει: $w_0 = \varepsilon \vee w_1 = abb \vee w_2 = bab \vee w_3 = bba$ 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.

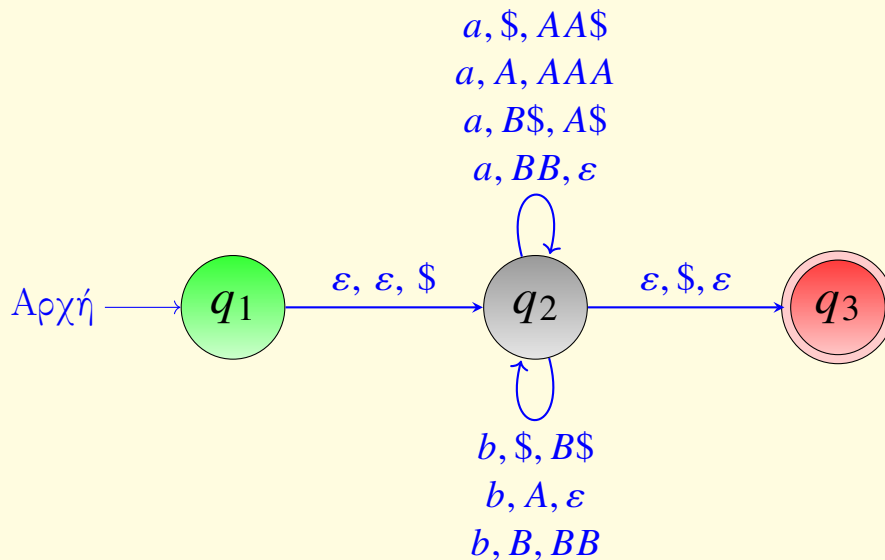
Άρα, υποθέτουμε ένα κόμβο για όλη την ανάγνωση βασιζόμενοι σε σύνθετες αναδρομές, συν επιπρόσθετους βοηθητικούς κόμβους (Εναρκτήριο με μετάβαση αρχικοποίησης σωρού με δείκτη τελευταίου κελιού και τελικό με μετάβαση προς αυτόν όπου αδειάζει τον δείκτη από τον σωρό). Ξεκινάμε την κατασκευή του αυτομάτου,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

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

- $K = \{q_1, q_2, q_3\}$
- $\Sigma = \{a, b\}$
- $\Gamma = \{A, B, \$\}$
- $\Delta =$

1. $(q_1, \varepsilon, \varepsilon) \rightarrow (q_2, \$)$	6. $(q_2, a, A) \rightarrow (q_2, AAA)$
2. $(q_2, b, \$) \rightarrow (q_2, B\$)$	7. $(q_2, a, BB) \rightarrow (q_2, \varepsilon)$
3. $(q_2, b, B) \rightarrow (q_2, BB)$	8. $(q_2, a, B\$) \rightarrow (q_2, A\$)$
4. $(q_2, b, A) \rightarrow (q_2, \varepsilon)$	9. $(q_2, \varepsilon, \$) \rightarrow (q_3, \varepsilon)$
5. $(q_2, a, \$) \rightarrow (q_2, AA\$)$	
- $s = q_1$
- $F = q_3$

DFA $M_{\text{τελικό}}$



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 Άσκηση 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

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

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

$$\bullet \forall (A, B) \ni A \subset B \rightarrow A \cap B = A, A \subset B \quad (3)$$

$$\bullet A \subset B \subset \dots \subset N \Rightarrow A \subset N \quad (4)$$

$$\bullet \stackrel{(2)(4)}{\Rightarrow} \mathcal{L}_{FIN} \subset \mathcal{L}_{DCF} \quad (5)$$

$$\bullet \mathcal{L}_{FIN} \cap \mathcal{L}_{DCF} \stackrel{(3)(5)}{=} \mathcal{L}_{FIN} \subset \mathcal{L}_{DCF} \subset \mathcal{L}_{CF} \stackrel{(4)}{\Rightarrow} \mathcal{L}_{FIN} \subset \mathcal{L}_{CF} \Rightarrow \mathcal{L}_{FIN} \in \mathcal{L}_{CF} \quad (6)$$

$$\bullet \mathcal{Q} \stackrel{(6)}{:=} \{\forall L_{DCF} \in \mathcal{L}_{DCF}, \forall L_{REG} \in \mathcal{L}_{REG} : L_{DCF} \cap L_{REG} \in \mathcal{L}_{CF}\} \quad (7)$$

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

Για να εξηγήσω την λογική, δείξαμε ότι:

1. Τομή μεταξύ ενός συνόλου με ενός υποσυνόλου (γνήσιου ή μη) παράγει μόνο το ίδιο το υποσύνολο. Αποδεικνύεται 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 συλλογίζόμενοι 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 $A \subset B \rightarrow A \cap B \Rightarrow A$ και $A \subset B \subset C \Rightarrow A \subset C$:

1. Ορισμός γνήσιου υποσυνόλου:

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

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

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

$$\begin{aligned} A \subset B \subset C &= [\forall x(x \in A \Rightarrow x \in B \Rightarrow x \in C)] \wedge [\exists y(y \in C \wedge y \notin B \Rightarrow y \notin A)] \Rightarrow \\ &\Rightarrow [\forall x(x \in A \Rightarrow x \in C)] \wedge [\exists y(y \in C \wedge y \notin A)] = A \subset C \end{aligned}$$

3. Ορισμός τομής:

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

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

$$A \cap B = C \rightarrow \forall x[(x \in A \wedge x \in B) \Leftrightarrow x \in C] \text{ αλλά } A \subset B \rightarrow \forall x(x \in A \Rightarrow x \in B)$$

οπότε στη συγκεκριμένη περίπτωση (και όλες τις αντίστοιχες) έχουμε:

$$\begin{aligned} A \cap B = C &\rightarrow [\forall x(x \in A \Rightarrow x \in B \Rightarrow x \in C)] \wedge [\exists y(y \notin A \wedge y \in B \Rightarrow y \notin C)] \Rightarrow \\ &[\forall x(x \in A \Rightarrow x \in C)] \wedge [\forall y(y \notin A \Rightarrow y \notin C)] \Rightarrow A = C \Rightarrow A \cap B = A \end{aligned}$$

3 - β Απάντηση Υποερωτήματος (β)

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

Αυτό μπορεί να γίνει μεταξύ άλλων με απόπειρα κατασκευής CFG της γλώσσας L , 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 $bbabbc$ 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 των δύο και συγκεκριμένα:

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

- Για $L_{LHS} : S \rightarrow \varepsilon \mid A, \quad A \rightarrow \varepsilon \mid aAbbb$
- Για $L_{RHS} : S \rightarrow \varepsilon \mid C, \quad C \rightarrow \varepsilon \mid bbCc$
- Άρα : $S \rightarrow \varepsilon \mid AC, \quad A \rightarrow \varepsilon \mid aAbbb, \quad C \rightarrow \varepsilon \mid bbCc$
- $G = (\{a, b, c\}, \{\varepsilon\},$
 $\{S \rightarrow \varepsilon \mid AC, A \rightarrow \varepsilon \mid aAbbb, C \rightarrow \varepsilon \mid bbCc\}, S)$

4 Άσκηση 4η - Αναγνώριση γλωσσών χωρίς συμφραζόμενα:

[30%] Έστω η γραμματική χωρίς συμφραζόμενα

$G = (V, \Sigma, R, S)$, όπου $V = \{S, A, L, D, a, b\}$, $\Sigma = \{a, b\}$ και

$R = \{S \rightarrow A, S \rightarrow L, A \rightarrow LbADa, A \rightarrow LaD, L \rightarrow Da, L \rightarrow e, D \rightarrow b\}$.

(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.

Δηλαδή $V \cap \Sigma = \emptyset$, 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 παραγωγής $S_0 \rightarrow S$

4. $\varepsilon \in L(G) \Leftrightarrow S \rightarrow \varepsilon$ ή $S_0 \rightarrow S | \varepsilon$ εάν το S εμφανίζεται στην παραγωγή κάποιου κανόνα.

Μετατροπή $CFG \rightarrow CNF$ 1/3

- Εισαγωγή νέας κατάστασης

1. Στη συγκεκριμένη γραμματική ισχύει $S \in S, S \rightarrow L \rightarrow \varepsilon \Leftrightarrow \varepsilon \in L(G)$ αλλά το S δεν υπάρχει στη δεξιά μεριά κανενός κανόνα παραγωγής και άρα δεν απαιτείτε κατασκευή νέου κανόνα παραγωγής $S_0 \rightarrow S \mid \varepsilon$.

- Απαλοιφή κενής παραγωγής

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

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

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

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

$$bADa \mid aD$$

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

$$3. L \rightarrow Da$$

$$4. D \rightarrow b$$

$$4. D \rightarrow b$$

- Απαλοιφή μοναδιαίων παραγωγών (μη τερματικών συμβόλων)

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

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

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

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

$$bADa \mid aD$$

$$3. L \rightarrow Da$$

$$3. L \rightarrow Da$$

$$4. D \rightarrow b$$

$$4. D \rightarrow b$$

- Απαλοιφή μη προσβάσιμων συμβόλων

1. Όλα τα σύμβολα είναι προσβάσιμα \rightarrow καμία αλλαγή.

Μετατροπή $CFG \rightarrow CNF$ 2/3

- Απομόνωση τερματικών συμβόλων

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

$$2. A \rightarrow \mathbf{LbADa} \mid \mathbf{LaD} \mid \mathbf{bADa} \mid \mathbf{aD}$$

$$3. L \rightarrow \mathbf{Da}$$

$$4. D \rightarrow b$$

$$1. S \rightarrow \mathbf{LDADD_a} \mid \mathbf{LD_aD} \mid \mathbf{DADD_a} \mid \mathbf{D_aD} \mid \mathbf{DD_a} \mid \varepsilon$$

$$2. A \rightarrow \mathbf{LDADD_a} \mid \mathbf{LD_aD} \mid \mathbf{DADD_a} \mid \mathbf{D_aD}$$

$$3. L \rightarrow \mathbf{DD_a}$$

$$4. D \rightarrow b$$

$$5. \mathbf{D_a} \rightarrow \mathbf{a}$$

- Απαλοιφή τριαδικών+ παραγωγών (1ο πέρασμα)

$$1. S \rightarrow \mathbf{LDADD_a} \mid \mathbf{LD_aD} \mid \mathbf{DADD_a} \mid D_aD \mid DD_a \mid \varepsilon$$

$$2. A \rightarrow \mathbf{LDADD_a} \mid \mathbf{LD_aD} \mid \mathbf{DADD_a} \mid D_aD$$

$$3. L \rightarrow DD_a$$

$$4. D \rightarrow b$$

$$5. D_a \rightarrow a$$

$$1. S \rightarrow \mathbf{LX_1X_2} \mid \mathbf{LX_3} \mid \mathbf{X_1X_2} \mid D_aD \mid DD_a \mid \varepsilon$$

$$2. A \rightarrow \mathbf{LX_1X_2} \mid \mathbf{LX_3} \mid \mathbf{X_1X_2} \mid D_aD$$

$$3. L \rightarrow DD_a$$

$$4. D \rightarrow b$$

$$5. D_a \rightarrow a$$

$$6. \mathbf{X_1} \rightarrow \mathbf{DA}$$

$$7. \mathbf{X_2} \rightarrow \mathbf{DD_a}$$

$$8. \mathbf{X_3} \rightarrow \mathbf{D_aD}$$

Μετατροπή $CFG \rightarrow CNF$ 3/3

- Απαλοιφή τριαδικών+ παραγωγών (2ο πέρασμα)

$$1. S \rightarrow \mathbf{LX_1X_2} \mid LX_3 \mid$$

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

$$2. A \rightarrow \mathbf{LX_1X_2} \mid LX_3 \mid$$

$$X_1X_2 \mid D_aD$$

$$3. L \rightarrow DD_a$$

$$4. D \rightarrow b$$

$$5. D_a \rightarrow a$$

$$6. X_1 \rightarrow DA$$

$$7. X_2 \rightarrow DD_a$$

$$8. X_3 \rightarrow D_aD$$

$$1. S \rightarrow \mathbf{LX_4} \mid LX_3 \mid$$

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

$$2. A \rightarrow \mathbf{LX_4} \mid LX_3 \mid$$

$$X_1X_2 \mid D_aD$$

$$3. L \rightarrow DD_a$$

$$4. D \rightarrow b$$

$$5. D_a \rightarrow a$$

$$6. X_1 \rightarrow DA$$

$$7. X_2 \rightarrow DD_a$$

$$8. X_3 \rightarrow D_aD$$

$$9. \mathbf{X_4 \rightarrow X_1X_2}$$

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

$$\{S \rightarrow \varepsilon, S \rightarrow LX_4, S \rightarrow X_1X_2, S \rightarrow LX_3, S \rightarrow D_aD, S \rightarrow DD_a,$$

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

$$L \rightarrow DD_a, D \rightarrow b, D_a \rightarrow a, X_1 \rightarrow DA, X_2 \rightarrow DD_a, X_3 \rightarrow D_aD, X_4 \rightarrow X_1X_2\}, S)$$

4 - β Απάντηση Υποερωτήματος (β)

Χρησιμοποιούμε το παρακάτω ως αναφορά για τους κανόνες παραγωγής καθώς με αρίθμηση για να μπορούμε να διαχωρίσουμε αυτούς που ξεκινάνε από ίδιο μη τερματικό σύμβολο:

$$1. S \rightarrow LX_4$$

$$10. A \rightarrow D_aD$$

$$2. S \rightarrow LX_3$$

$$11. L \rightarrow DD_a$$

$$3. S \rightarrow X_1X_2$$

$$12. D \rightarrow b$$

$$4. S \rightarrow D_aD$$

$$13. D_a \rightarrow a$$

$$5. S \rightarrow DD_a$$

$$14. X_1 \rightarrow DA$$

$$6. S \rightarrow \varepsilon$$

$$15. X_2 \rightarrow DD_a$$

$$7. A \rightarrow LX_4$$

$$16. X_3 \rightarrow D_aD$$

$$8. A \rightarrow LX_3$$

$$17. X_4 \rightarrow X_1X_2$$

$$9. A \rightarrow X_1X_2$$

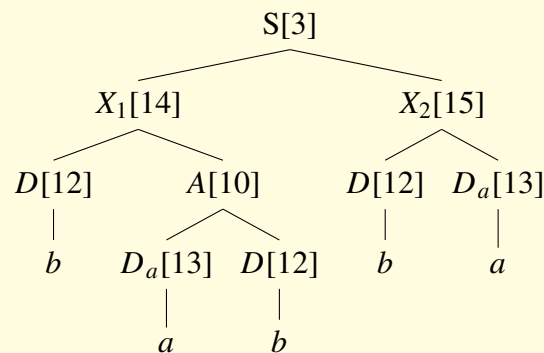
Πίνακας συντακτικής ανάλυσης για $w = babba$

- 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 κάποιο κανόνα παραγωγής, από το αρχικό (μη τερματικό προφανώς) σύμβολο. Εάν αυτό υπάρχει τότε $w \in G$, ειδάλλως $w \notin G$.

				a
			b	L[11], X ₂ [15], S[5]
		b	∅	∅
	a	S[4], A[10], X ₃ [16]	∅	∅
b	L[11], X ₂ [15], S[5]	X ₁ [14]	∅	A[9], S[3], X ₄ [17]

Δέντρο 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 βάσει 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.



Που δίνει:

$b \quad a \quad b \quad b \quad a$

