Κεφάλαιο 3

Κανονικές εκφράσεις και γλώσσες

3.1 Κανονικές εκφράσεις και οι γλώσσες τους

Θα δώσουμε κατ΄ αρχήν ένα αναδρομικό ορισμό για το τι είναι μια *κανονική έκφραση* (regular expression) και το ποια γλώσσα παριστάνει.

Ορισμός 27. (Κανονικές εκφράσεις και οι γλώσσες τους)
Οι παρακάτω λέξεις ορίζονται να είναι *κανονικές εκφράσεις* με τις αντίστοιχες γλώσσες. (Με $L(\cdot)$ συμβολίζουμε τη *γλώσσα μιας έκφρασης*.)

- $\emptyset$, και $L(\emptyset) = \emptyset$
- ϵ, και $L(\epsilon) = \{\epsilon\}$
- α, για κάθε $\alpha \in \Sigma$, και $L(\alpha) = \{\alpha\}$.

Επίσης, αν r και s είναι κανονικές εκφράσεις τότε και οι ακόλουθες λέξεις είναι επίσης κανονικές εκφράσεις.

1. (r), και $L((r)) = L(r)$.
2. $(r + s)$, και $L((r + s)) = L(r) \cup L(s)$,
3. (rs), και $L((rs)) = L(r)L(s)$,
4. (r^*), και $L((r^*)) = L(r)^*$.

Μια γλώσσα L λέγεται κανονική αν υπάρχει κανονική έκφραση r με $L = L(r)$.

Παρατήρηση 10. Αν μπορούμε να παραλείψουμε τις παρανθέσεις χωρίς να αλλάξουμε το νόημα τότε το κάνουμε αυτό. Λαμβάνουμε υπόψιν ότι τη μεγαλύτερη προτεραιότητα έχει ο εκθέτης $*$, μετά έρχεται η συγκόλληση και τέλος η πρόσθεση. Για παράδειγμα η κανονική έκφραση $01^* + 10^*$ ερμηνεύεται ως $(0(1^*)) + (1(0^*))$. Αν r είναι μια κανονική έκφραση τότε χρησιμοποιούμε επίσης τη συντομογραφία r^+ αντί για rr^*.

Παράδειγμα 32. Τα παρακάτω είναι παραδείγματα κανονικών εκφράσεων και των γλωσσών τους.

Κανονική έκφραση	**Αντίστοιχη γλώσσα**
00	$\{00\}$
$(0+1)^*$	Όλες οι λέξεις πάνω από το $\Sigma = \{0, 1\}$
$(0+1)^*00(0+1)^*$	Όλες οι λέξεις πάνω από το $\Sigma = \{0, 1\}$ που έχουν δύο διαδοχικά μηδενικά
$(1+10)^*$	Ότι αρχίζει με 1 και δεν έχει διαδοχικά 0
$(0+\epsilon)(1+10)^*$	Ότι δεν έχει διαδοχικά 0
$0^*1^*2^*$	Λέξεις όπου τα 0 έρχονται πριν τα 1 κι αυτά πριν τα 2.

Άσκηση 41. Γράψτε κανονικές εκφράσεις για τις ακόλουθες γλώσσες:

1. Λέξεις που δεν περιέχουν τη μορφή 101.
2. Λέξεις όπου όλες οι εμφανίσεις διαδοχικών 0 συμβαίνουν πριν από οποιαδήποτε εμφάνιση διαδοχικών 1.

Άσκηση 42. Περιγράψτε τις γλώσσες που ορίζονται από τις κανονικές εκφράσεις:

1. $(11+0)^*(00+1)^*$
2. $(1+01+001)^*(\epsilon+0+00)^*$
3. $(00+11+(01+10)(00+11)^*(01+10))^*$

Άσκηση 43. Δείξτε τις παρακάτω ταυτότητες όπου r, s, t είναι οποιεσδήποτε κανονικές εκφράσεις. Η ισότητα παρακάτω δε σημαίνει ότι οι δύο εκφράσεις είναι ίσες (και δεν είναι, μια και οι εκφράσεις είναι λέξεις, και ως τέτοιες διαφέρουν) αλλά ότι οι αντίστοιχες γλώσσες είναι ίσες.

1. $r+s=s+r$
2. $(r+s)+t=r+(s+t)$
3. $(rs)t=r(st)$
4. $r(s+t)=rs+rt$
5. $(r+s)t=rt+st$
6. $\emptyset^*=\epsilon$
7. $(r^*)^*=r^*$
8. $(\epsilon+r)^*=r^*$
9. $(r^*s^*)^*=(r+s)^*$

3.2 Κανονικότητα γλωσσών των αυτομάτων

Η βασική πρόταση είναι η ακόλουθη.

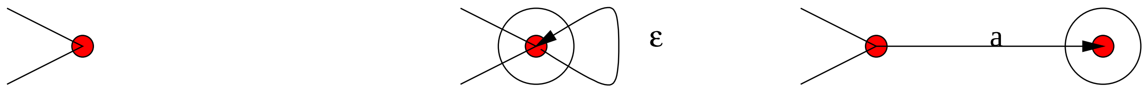
Θεώρημα 5. (Κανονικότητα γλωσσών των αυτομάτων)
Μια γλώσσα είναι κανονική αν και μόνο αν αναγνωρίζεται από κάποιο αυτόματο.

Παρατήρηση 11. Παρατηρείστε ότι στο Θεώρημα 5 δεν ξεκαθαρίζουμε για τι είδους αυτόματο μιλάμε. Έχουμε όμως ήδη αποδείξει ότι κάθε ϵ-NFA είναι ισοδύναμο με κάποιο NFA και κάθε NFA με κάποιο DFA. Αυτό σημαίνει ότι μια γλώσσα αναγνωρίζεται από ένα αυτόματο κάποιου είδους από τα τρία αν και μόνο αν αναγνωρίζεται από αυτόματο οποιουδήποτε είδους.

Απόδειξη. Θα δείξουμε ότι (α) Κάθε κανονική γλώσσα αναγνωρίζεται από κάποιο ϵ-NFA, και (β) Η γλώσσα που αναγνωρίζει κάθε DFA είναι κανονική.

(α) Χρησιμοποιούμε επαγωγή ως προς το μήκος της κανονικής έκφρασης r που παράγει τη γλώσσα. Αν πρόκειται για μια από τις εκφράσεις $\emptyset$, ϵ ή a, με $a \in \Sigma$, πολύ εύκολα φτιάχνουμε αυτόματα που τις αναγνωρίζουν όπως φαίνεται στο παρακάτω Σχήμα 13.

Αν τώρα έχουμε μια έφραση x του τύπου $r+s$, rs η r^*, τότε τα μήκη των r και s είναι αυτστηρά μικρότερα του $|x|$, άρα μπορούμε να υποθέσουμε επαγωγικά ότι έχουμε κάποια αυτόματα M και N που αναγνωρίζουν τις γλώσσες $L(r)$ και $L(s)$. Χρησιμοποιούμε τα M και N σα μαύρα κουτιά και μας ενδιαφέρει μόνο να 'βλέπουμε' απ΄ έξω τις αρχικές και τελικές τους καταστάσεις.



(a) (b) (c)

Σχήμα 13. Τα ϵ-NFA για τις εκφράσεις $\emptyset$ (a), ϵ (b) και a (c)

Στο Σχήμα 14 βλέπουμε στα (a) και (b) τα αυτόματα M και N που αντιστοιχούν στις εκφράσεις r και s. Στα (c), (d) και (e) βλέπουμε πως αυτά συνδυάζονται ώστε να φτιάξουν αυτόματα για τις γλώσσες $r+s$, rs και r^*.

Στο (c) ορίζουμε μια νέα αρχική κορυφή και την ενώνουμε με ϵ-ακμές με τις δύο αρχικές κορυφές των M και N. Οι τελικές καταστάσεις παραμένουν οι ίδιες.

Στο (d) αρχική κορυφή είναι αυτή του M του οποίου οι τελικές καταστάσεις συνδέονται με ϵ-ακμές με την αρχική του N. Τελικές καταστάσεις του συμπλέγματος είναι αυτές του N.

Στο (e) οι τελικές καταστάσεις του M συνδέονται με ϵ-ακμές με την αρχική κατάσταση του M. Αρχικές και τελικές καταστάσεις παραμένουν οι ίδιες.

(β) Έστω DFA $M = (Q, \Sigma, \delta, q_1, F)$, όπου $Q = \{q_1, \ldots, q_n\}$. Ορίζουμε για $i, j = 1, \ldots, n$, $k = 0, \ldots, n$, τις γλώσσες $R^k_{i,j}$ να είναι εκείνες οι λέξεις του Σ^* που είναι τέτοιες ώστε αν ξεκινήσουμε από την κορυφή q_i και τις ακολουθήσουμε τότε φτάνουμε στην κορυφή q_j **χωρίς να χρησιμοποιήσουμε κορυφή q_l με $l > k$.**

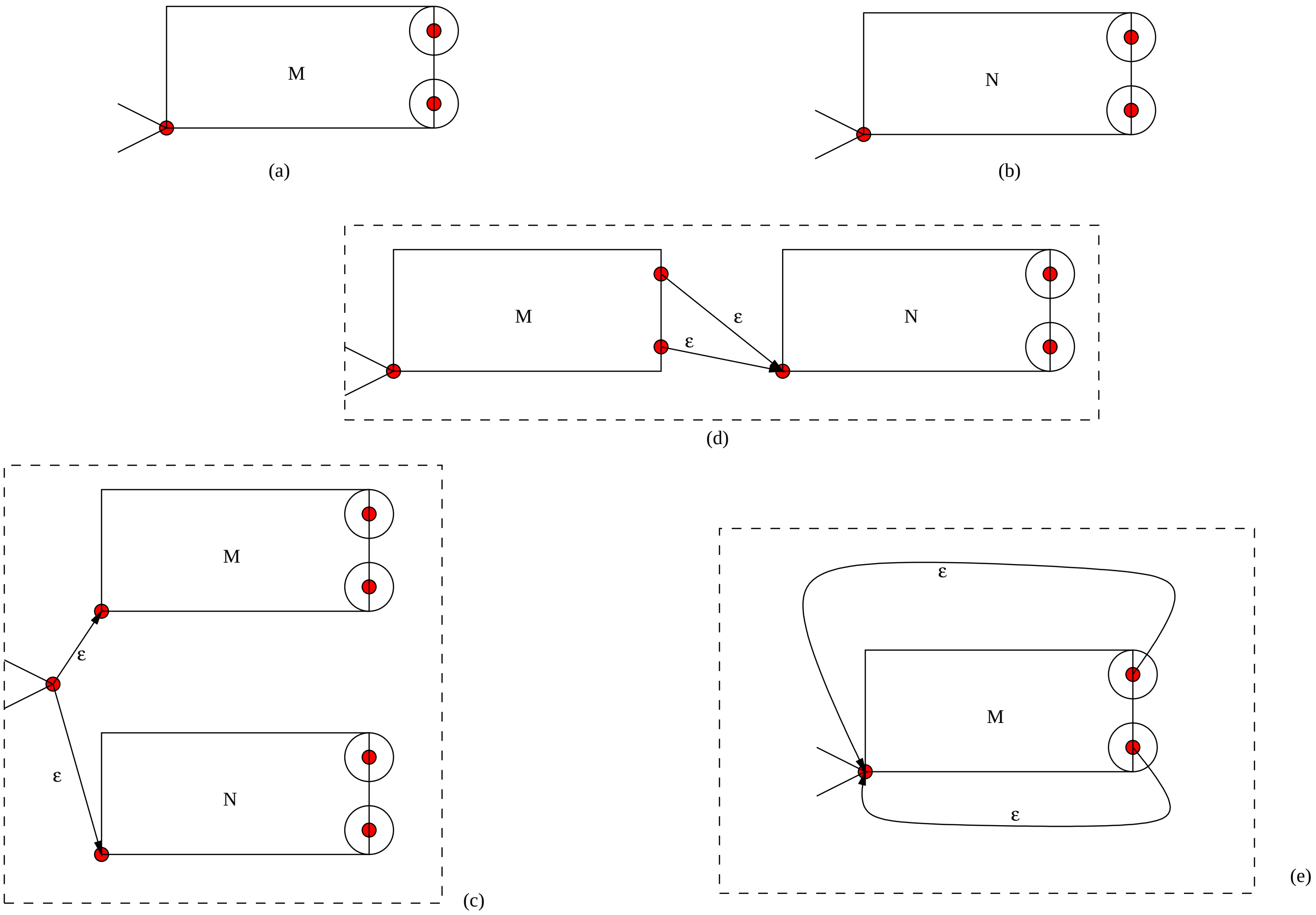
Είναι φανερό ότι
$$L(M) = \bigcup_{q_j \in F} R^n_{1,j}. \tag{3.1}$$
Με άλλα λόγια, αποδεκτές γίνονται εκείνες οι λέξεις που μας επιτρέπουν να φτάσουμε, ξεκινώντας από την αρχική κορυφή q_1, σε κάποια από τις κορυφές $q_j \in F$, χωρίς κανένα περιορισμό ως προς τις ενδιάμεσες κορυφές (επειδή ο άνω δείκτης είναι n, και, ούτως ή άλλως, δεν υπάρχουν κορυφές q_l, με $l > n$.

Θα δείξουμε με επαγωγή ως προς το k ότι οι γλώσσες $R^k_{i,j}$ είναι όλες κανονικές. Άρα κανονική είναι και η $L(M)$ αφού με βάση την (3.1) είναι **πεπερασμένη** ένωση από κανονικές γλώσσες, και η ένωση δύο κανονικών γλωσσών είναι εξ ορισμού κανονική.

Για $k = 0$ τώρα, παρατηρούμε ότι η απαίτηση, στον ορισμό της $R^k_{i,j}$ όσον αφορά το ποιες κορυφές δεν πρέπει να χρησιμοποιηθούν είναι ιδιαίτερα αυστηρή, αφού η συνθήκη $l > 0$ ισχύει για κάθε κορυφή $q_l \in Q$. Άρα έχουμε
$$R^0_{i,j} = \begin{cases} \{\alpha \in \Sigma : \quad \delta(q_i, \alpha) = q_j\} & (i \neq j) \\ \{\alpha \in \Sigma : \quad \delta(q_i, \alpha) = q_i\} \cup \{\epsilon\} & (i = j) \end{cases}$$
Αυτό σημαίνει ότι οι γλώσσες $R^0_{i,j}$ είναι πεπερασμένα σύνολα από σύμβολα του Σ ή το γράμμα ϵ. Κάθε ένα όμως από αυτά είναι κανονική γλώσσα άρα είναι τέτοια και η $R^0_{i,j}$.

Όσον αφορά το επαγωγικό βήμα, αν υποθέσουμε ότι οι $R^{k-1}_{i,j}$ είναι όλες κανονικές τότε το ίδιο συνάγουμε και για τις $R^k_{i,j}$ αφού παρατηρήσουμε ότι ισχύει η αναδρομική σχέση
$$R^k_{i,j} = R^{k-1}_{i,k} (R^{k-1}_{k,k})^* R^{k-1}_{k,j} \cup R^{k-1}_{i,j}. \tag{3.2}$$
Γιατί όμως ισχύει η (3.2);

Είναι φανερό ότι η γλώσσα $R^k_{i,j}$ είναι υπερσύνολο της $R^{k-1}_{i,j}$, αφού ο περιορισμός $l \leq k$, στον ορισμό της $R^k_{i,j}$, γίνεται ασθενέστερος (ισχύει πιο συχνά) όσο μεγαλώνει το k. Ποιες είναι όμως εκείνες οι λέξεις που ανήκουν στο σύνολο $R^k_{i,j}$ αλλά όχι στο $R^{k-1}_{i,j}$, οι λέξεις με άλλα λόγια της (συνολοθεωρητικής) διαφοράς $R^k_{i,j} \setminus R^{k-1}_{i,j}$;Είναι ακριβώς εκείνες οι λέξεις που οδηγούν από την κατάσταση q_i στην q_j, χωρίς να 'πατούν' σε κορυφή q_l, $l > k$, αλλά που πατούν τουλάχιστον μια φορά στην κορυφή q_k όως φαίνεται στο Σχήμα 15.



Σχήμα 14. (α) Αυτόματο για r, (β) για s, (ς) για $r+s$, (δ) για rs, (ε) για r^*

Μια τέτοια λέξη αντιστοιχεί σε ένα μονοπάτι πάνω στο DFA που σίγουρα ‘πατάει’ πάνω στην κορυφή q_k, ενδεχομένως και πάνω από μία φορά (στο Σχήμα 15 πατάει δύο φορές). Αν ονομάσουμε w μια τέτοια λέξη, και ονομάσουμε σ το πρόθεμα της w που αντιστοιχεί στο μονοπάτι από το q_i στο q_k, που δεν πατάει στην q_k, και τ το επίθεμα της w γαι το μονοπάτι $q_k \to q_j$ που δεν πατάει στην q_k, τότε η w γράφεται

$$w = \sigma v_1 \cdots v_t \tau,$$

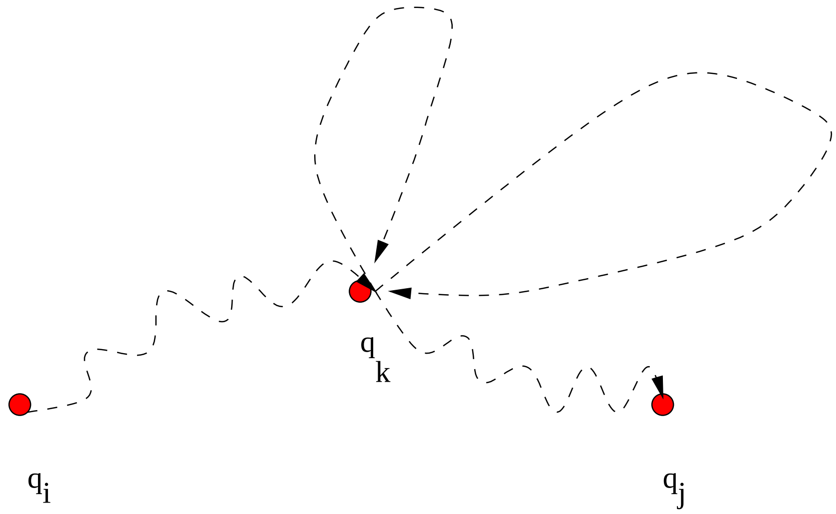
όπου οι λέξεις $v_1, \ldots, v_t$ αντιστοιχούν σε μονοπάτια που αρχίζουν και τελειώνουν από το q_k, χωρίς να πατούν στο q_k. Είναι δηλ. $\sigma \in R_{i,k}^{k-1}$, $v_l \in R_{k,k}^{k-1}$, $l = 1, \ldots, t$, και $\tau \in R_{k,j}^{k-1}$, έχουμε άρα δείξει τον εγκλεισμό

$$R_{i,j}^k \setminus R_{i,j}^{k-1} \subseteq R_{i,k}^{k-1}(R_{k,k}^{k-1})^* R_{k,j}^{k-1}.$$

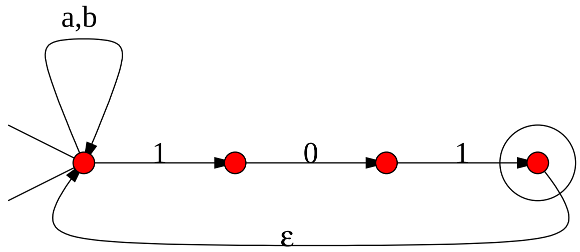
Ο αντίστροφος εγκλεισμός είναι ακόμη πιο εύκολος και παραλείπεται.
□

Άσκηση 44. Κατασκευάστε DFA για τις κανονικές εκφράσεις:

1. $10 + (0+11)0^*1$
2. $01\left(((10)^* + 111)^* + 0\right)^* 1$
3. $((0+1)(0+1))^* + ((0+1)(0+1)(0+1))^*$



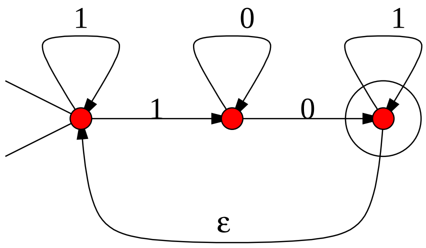
Σχήμα 15. Ένα μονοπάτι στο DFA που αντιστοιχεί σε λέξη του $R^k_{i,j} \setminus R^{k-1}_{i,j}$



Σχήμα 16. Το Σχήμα για την Άσκηση 45

Άσκηση 45. Δώστε μια κανονική έκφραση για τη γλώσσα που αναγνωρίζει το αυτόματο του Σχήματος 16.

Άσκηση 46. Δώστε μια κανονική έκφραση για τη γλώσσα που αναγνωρίζει το αυτόματο του Σχήματος 17.



Σχήμα 17. Το Σχήμα για την Άσκηση 46

3.3 Κλειστότητα κανονικών γλωσσών κάτω από απλές πράξεις

Από τον ορισμό των κανονικών εκφράσεων και των γλωσσών είναι φανερό ότι αν L_1 και L_2 είναι κανονικές γλώσσες τότε κανονική είναι και η $L_1 \cup L_2$ όπως και η γλώσσα L_1^*. Δεν είναι όμως καθόλου προφανές από τον ορισμό των κανονικών εκφράσεων το αν η συμπληρωματική γλώσσα

$$L_1^c = \{w \in \Sigma^* : \quad w \notin L_1\}$$

είναι κανονική. Γίνεται όμως και αυτό φανερό αν χρησιμοποιήσουμε το Θεώρημα 5 που λέει ότι μια γλώσσα είναι κανονική αν και μόνο αν υπάρχει ένα DFA που την αναγνωρίζει. Έστω λοιπόν ότι το DFA M αναγνωρίζει τη γλώσσα L_1. Φτιάνχουμε ένα DFA M' από το M κρατώντας τις ίδιες καταστάσεις, αλφάβητο, αρχική κατάσταση και ακμές του M αλλά κάνουμε όλες τις τελικές καταστάσεις του M μη τελικές και όλες τις μη τελικές καταστάσεις του M τις κάνουμε τελικές στο M'. (Εδώ πρέπει να είμαστε λίγο προσεκτικοί και να δουλεύουμε στο μορφή του M που δε χρησιμοποιεί συντομογραφία, αλλά που ικανοποιεί την αρχική απαίτηση για DFA, ότι δηλ. για κάθε κατάσταση και για κάθε γράμμα του αλφαβήτου υπάρχει ακριβώς μια ακμή.) Είναι φανερό τώρα ότι μια λέξη γίνεται δεκτή στο M' αν και μόνο αν *δε γίνεται δεκτή στο* M, άρα το M' είναι ένα DFA για τη γλώσσα L_1^c.

Είναι τώρα εύκολο να δείξουμε ότι και η γλώσσα $L_1 \cap L_2$ είναι κανονική. Αρκεί να δείξουμε ότι το συμπλήρωμά της είναι, και

$$(L_1 \cap L_2)^c = L_1^c \cup L_2^c,$$

και από αυτά που είπαμε προηγουμένως προκύπτει ο ισχυρισμός.

Ορισμός 28. *Αντικατάσταση* (substitution) σε ένα αλφάβητο Σ είναι μια οποιαδήποτε αντιστοίχιση των γραμμάτων του αλφαβήτου σε *γλώσσες* πάνω από το Σ, μια οποιαδήποτε συνάρτηση δηλ.

$$f : \Sigma \to 2^{\Sigma^*},$$

μια συνάρτηση δηλ. που παίρνει τιμές τυχόντα υποσύνολα του Σ^*. Αν για κάθε $\alpha \in \Sigma$ η γλώσσα $f(a)$ είναι κανονική, τότε και η αντικατάσταση ονομάζεται κανονική.

Έχοντας μια αντικατάσταση $f : \Sigma \to 2^{\Sigma^*}$ μπορούμε να επεκτείνουμε το πεδίο ορισμού της από γράμματα του Σ σε λέξεις του Σ^*. Αν $w = \alpha_1 \cdots \alpha_n$, $\alpha_i \in \Sigma$, είναι μια λέξη, τότε ορίζουμε (χρησιμοποιούμε το ίδιο σύμβολο f για να συμβολίσουμε και την επεκτεταμένη συνάρτηση)

$$f(w) = f(\alpha_1) \cdots f(\alpha_n),$$

η γλώσσα δηλ. που προκύπτει από τη συγκόλληση των γλωσσών $f(\alpha_1), \ldots, f(\alpha_n)$.

Αν L είναι μια γλώσσα ορίζουμε ως συνήθως

$$f(L) = \bigcup_{w \in L} f(w).$$

Θεώρημα 6. *Αν L κανονική, και η αντικατάσταση f είναι κανονική, τότε η $f(L)$ είναι κανονική γλώσσα.*

Σκίτσο απόδειξης. Αφού η L είναι κανονική τότε προκύπτει ως γλώσσα μιας κανονικής έκφρασης r. Είναι εύκολο να δούμε (αλλά δεν το κάνουμε με λεπτομέρεια εδώ) ότι η γλώσσα $f(L)$ είναι ακριβώς αυτή που προκύπτει αν τροποποιήσουμε την έκφραση r αντικαθιστώντας κάθε γράμμα α που εμφανίζεται στην r με μια κανονική έκφραση για τη γλώσσα $f(\alpha)$. Άρα η γλώσσα $f(L)$ δίδεται από κανονική έκφραση, άρα είναι κανονική.
□

Άσκηση 47. Αποδείξτε το Θεώρημα 6 ακολουθώντας τη 'συνταγή' του σκίτσου απόδειξης πο δόθηκε. Μπορείτε για παράδειγμα να χρησιμοποιήσετε επαγωγή ως προς το μήκος μιας κανονικής έκφρασης της κανονικής γλώσσας L.

Ορισμός 29. Μια αντικατάσταση f λέγεται *ομομορφισμός* αν για κάθε $\alpha \in \Sigma$ γλώσσα $f(\alpha)$ είναι μονοσύνολο.

Επειδή κάθε πεπερασμένο σύνολο λέξεων είναι κανονική γλώσσα (από τον ορισμό του τι είναι κανονική γλώσσα) από το προηγούμενο θεώρημα έπεται ότι οι ομομορφισμοί διατηρούν την κανονικότητα των γλωσσών.

Ορισμός 30. Αν f είναι ομομορφισμός και L είναι μια γλώσσα, η *αντίστροφη ομομορφική εικόνα* της L είναι η γλώσσα

$$f^{-1}(L) = \{w \in \Sigma^* : \quad f(w) \in L\}.$$

Θεώρημα 7. *Αν f ομομορφισμός και L κανονική γλώσσα τότε και η $f^{-1}(L)$ είναι κανονική.*

Απόδειξη. Έστω M ένα DFA που αναγνωρίζει την L. Φτιάχνουμε DFA M' ως εξής: κρατάμε ίδιες καταστάσεις και αρχικές και τελικές καταστάσεις με το M και ορίζουμε μόνο νέα συνάρτηση μετάβασης ως εξής: αν q είναι μια κατάσταση του M και $\alpha \in \Sigma$ ορίζουμε

$$\delta_{M'}(q, \alpha) = \delta_M(q, f(\alpha)).$$

Μεταβαίνουμε δηλ. στο M' σε εκείνη την κατάσταση όπου θα καταλήγαμε αν στο M ξεκινάγαμε από την κατάσταση q και ακολουθούσαμε τη *λέξη* $f(\alpha)$. Είναι φανερό ότι η λέξη w γίνεται αποδεκτή από το M' αν και μόνο αν η λέξη $f(w)$ γίνεται αποδεκτή από το M, άρα το DFA αναγνωρίζει τη γλώσσα $f^{-1}(L)$, οπότε αυτή είναι κανονική.
□

Άσκηση 48. Αν L είναι κανονική γλώσσα τότε και η γλώσσα

$$L^R = \left\{x \in \Sigma^* : \quad x^R \in L\right\}$$

είναι κανονική. Με x^R συμβολίζουμε τη λέξη x με τα γράμματά της σε ανάποδη σειρά. Π.χ. $011^R = 110$. (*Υπόδειξη:* Δουλέψτε πάνω σε μια κανονική έκφραση για την L.)

3.4 Το Λήμμα Άντλησης και μη κανονικές γλώσσες

Υπάρχουν γλώσσες που δεν είναι κανονικές. Αυτό είναι προφανές για πολύ γενικούς λόγους, που δεν έχουν τόσο πολύ να κάνουν με τη φύση των κανονικών γλωσσών, παρά μόνο με το γεγονός ότι κάθε κανονική γλώσσα επιδέχεται μια *πεπερασμένη περιγραφή*. Αυτή μπορεί να είναι είτε μια κανονική έκφραση, είτε η δομή ενός DFA που την αναγνωρίζει, για παράδειγμα.

Πόσες όμως διαφορετικές πεπερασμένες περιγραφές υπάρχουν;

Είναι φανερό ότι αυτές είνάι άπειρες το πλήθος (αφού π.χ. υπάρχουν οσοδήποτε μεγάλες κανονικές εκφράσεις), αλλά αυτό δεν είναι επαρκής πληροφορία για το πλήθος τους. Είναι πολύ πιο ακριβές το ότι υπάρχουν *αριθμήσιμες* το πλήθος πεπερασμένες περιγραφές, μπορούν δηλ. όλες οι πεπερασμένες περιγραφές να απαριθμηθούν: η περιγραφή 1, η περιγραφή 2, ..., η περιγραφή N, ..., με τρόπο ώστε να μη μείνει καμιά περιγραφή απ΄ έξω. Αυτός είναι ο ορισμός ενός αριθμήσιμου συνόλου. Ένα σύνολο δηλ. λέγεται αριθμήσιμο αν υπάρχει μια 1-1 απεικόνιση του συνόλου αυτού στους φυσικούς αριθμούς (βεβαιωθείτε ότι το καταλαβαίνετε αυτό).

Το γιατί το σύνολο όλων των πεπερασμένων εκφράσεων (πάνω από ένα σταθερό αλφάβητο Σ) είναι αριθμήσιμο είναι απλό. Αφού το Σ είναι πεπερασμένο υπάρχουν πεπερασμένο το πλήθος εκφράσεις που μπορούν να φτιάξουμε με μήκος n, για κάθε n. Για την ακρίβεια, υπάρχουν το πολύ $|\Sigma|^n$ τέτοιες εκφράσεις. Για να απαριθμήσουμε λοιπόν το σύνολο όλων των πεπερασμένων εκφράσεων, αριθμούμε πρώτα τις εκφράσεις μήκους 1, μετά αυτές μήκους 2, και συνεχίζουμε κατ΄ αυτόν τον τρόπο, ώστε δε μένει τίποτα αμέτρητο.

Είναι αρκετά πιο δύσκολο (και παραλείπουμε την απόδειξη) να δούμε ότι το σύνολο όλων των υποσυνόλων οποιουδήποτε άπειρου συνόλου *δεν είναι αριθμήσιμο*. Έτσι, το σύνολο όλων των γλωσσών πάνω από το Σ, δηλ. το σύνολο όλων των υποσυνόλων του άπειρου συνόλου Σ^* δεν είναι αριθμήσιμο. Άρα υπάρχει κάποια γλώσσα που δεν έχει αντίστοιχη πεπερασμένη περιγραφή, οπότε δεν είναι κανονική.

Το παραπάνω είναι ένα πολύ γενικό επιχείρημα, όπως είπαμε, και δεν εξαρτάται σχεδόν καθόλου από το τι εννοούμε κανονική γλώσσα, παρά μόνο ότι, ό,τι κι αν εννοούμε, η κανονική γλώσσα μπορεί να περιγραφεί πλήρως με μια πεπερασμένη ακολουθία από γράμματα, διαλεγμένα από ένα πεπερασμένο αλφάβητο. Με τον ίδιο τρόπο φαίνεται, π.χ. ότι υπάρχουν συναρτήσεις $f : \mathbf{N} \to \mathbf{N}$ που δεν είναι υπολογίσιμες από πρόγραμμα, ουσιαστικά ότι κι αν εννούμε με αυτό. Ας πούμε για παράδειγμα ότι θεωρούμε μια τέτοια συνάρτηση υπολογίσιμη αν υπάρχει ένα πρόγραμμα σε κάποια γλώσσα προγραμματισμού, ας πούμε την C που υπολογίζει τη συνάρτηση αυτή. Μα ένα πρόγραμμα αποτελεί πεπερασμένη περιγραφή της συνάρτησης, άρα, επειδή το πλήθος των συναρτήσεων $f : \mathbf{N} \to \mathbf{N}$ δεν είναι αριθμήσιμο, υπάρχει κάποια στην οποία δεν αντιστοιχεί πρόγραμμα, που δεν είναι δηλ. υπολογίσιμη.

Δε βοηθάει όμως αυτό το επιχείρημα στο να αποδείξει κανείς ότι συγκεκριμένες γλώσσες δεν είναι κανονικές. Αν και γνωρίζουμε δηλ. ότι οι περισσότερες γλώσσες (υπό μία έννοια) δεν είναι κανονικές, είναι παρ' όλα αυτά δύσκολο να δείξουμε αυτό για μια συγκεκριμένη γλώσσα που μας δίδεται, π.χ. για την

$$L_1 = \{0^n 1^n : \quad n = 0, 1, 2, \ldots\}.$$

Αυτή η γλώσσα απαρτίζεται από όλες τις λέξεις που έχουν αρχικά μια ομάδα από μηδενικά και μετά μια ίση σε μήκος ομάδα από άσσους, και τίποτε άλλο. Θα δούμε σε λίγο ένα τρόπο απόδειξης του ότι η L_1 δεν είναι κανονική. Διαισθητικά όμως ο λόγος είναι ο εξής (το παρακάτω *δεν* αποτελεί απόδειξη): ένα αυτόματο είναι ουσιαστικά ισοδύναμο με ένα υπολογιστή σαν κι αυτούς που ξέρουμε με μόνο τον περιορισμό ότι ο υπολογιστής αυτός έχει πεπερασμένη και σταθερή μνήμη, δεν εξαρτάται δηλ. η ποσότητα της μνήμης του από το πρόβλημα που έχει να λύσει. Αν είχαμε ένα πρόγραμμα που τρέχει σε ένα τέτοιο υπολογιστή και το οποίο αναγνωρίζει τη γλώσσα L_1, αυτό το πρόγραμμα θα είχε πρόσβαση σε μνήμη της οποίας το μήκος δε μπορεί να εξαρτάται από το μήκος της λέξης προς αναγνώριση. Όμως, δε μπορούμε να φανταστούμε ένα πρόγραμμα (αλγόριθμο) που θα αποφασίζει αν μια λέξη ανήκει στην L_1 ή οχι χωρίς να 'θυμάται' κάπου το πόσα μηδενικά έχει δει. Κι αυτό δε γίνεται αν ο αριθμός των μηδενικών είνάι πολύ μεγάλος, γιατί όσο αυξάνει το πλήθος των μηδενικών τόσο αυξάνει, χωρίς άνω φράγμα, το πλήθος των ψηφίων (η μνήμη) που χρειαζόμαστε για να αποθηκεύσουμε τον αριθμό αυτό.

Είναι δύσκολο να μετατρέψουμε το παραπάνω διαισθητικό επιχείρημα σε απόδειξη, οπότε η μέθοδος που ακολουθούμε για να δείξουμ εότι η L_1 δενείναι κανονική είναι αρκετά διαφορετική. Χρησιμοποιούμε το ακόλουθο πολύ χρήσιμο λήμμα.

Θεώρημα 8. *(Λήμμα Άντλησης – Pumping Lemma)*
Έστω ότι η L είναι κανονική. Τότε υπάρχει φυσικός αριθμός n (ο οποίος δε χρειάζεται να είναι μεγαλύτερος από τον ελάχιστο αριθμό καταστάσεων ενός DFA που αναγνωρίζει την L) ώστε για κάθε λέξη $z \in L$ με $|z| \geq n$, μπορούμε να γράψουμε

$$z = uvw, \quad (u, v, w \in \Sigma^*, \ |v| \geq 1, |uv| \leq n),$$

ούτως ώστε για κάθε $i = 0, 1, 2, \ldots$, η λέξη $uv^iw \in L$ (αυτή είναι η λέξη που αρχίζει με u ακολουθείται από την v οποιοδήποτε πεπερασμένο αριθμό από φορές – ακόμη και 0–και τελειώνει με w).

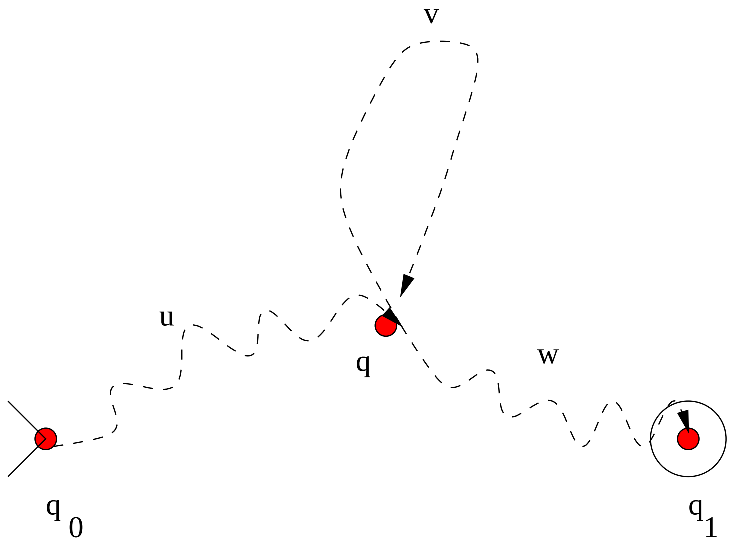
Απόδειξη: Έστω M ένα DFA με ελάχιστο αριθμό καταστάσεων που αναγνωρίζει τη γλώσσα L, και έστω n το πλήθος των καταστάσεων του M. Αν η λέξη z αναγνωρίζεται τότε ξεκινώντας από την αρχική κορυφή q_0 καταλήγουμε διαβάζοντας την z, με $|z| = m \geq n$, σε κάποια τελική κορυφή q_1 διανύοντας ένα μονοπάτι πάνω στο M:

$$q_0 = q_{i_1} \to q_{i_2} \to \cdots \to q_{i_{m+1}} = q_1$$

όπου το πλήθος των ακμών είναι m (μια ακμή για κάθε γράμμα της z) και άρα το πλήθος των κορυφών είναι $m + 1 > n$. Συμπεραίνουμε ότι υπάρχουν κάποιες κορυφές που εμφανίζονται τουλάχιστον δύο φορές στο μονοπάτι. Από αυτές τις κορυφές ονομάζουμε q αυτή που πρωτοεμφανίζεται για δεύτερη φορά στο μονοπάτι και ονομάζουμε v τη λέξη που μεσολαβεί μέχρι την επόμενη εμφάνιση της q, ενώ u ονομάζομε το πρόθεμα της z μέχρι την πρώτη εμφάνιση της q και w ονομάζομε το επίθεμα της z μετά τη δεύτερη εμφάνιση της q (βλ. Σχήμα 18).

Είναι φανερό ότι $|v| \geq 1$ και ότι $|uv| \leq n$, αφού σίγουρα όταν θα έχουμε διαβάσει το n-οστό γράμμα της z θα έχουμε ήδη δεί τουλάχιστον μια κορυφή δύο φορές, και ανάμεσα σε αυτές που έχουμε δεί δύο φορές η πρώτη είναι εξ ορισμού η q. Επίσης είναι φανερό ότι το βρόγχο που ξεκινά και τελειώνει στην q μπορούμε να μην το διανύσουμε καθόλου (οπότε δείχνουμε ότι η λέξη $uw \in L$) ή να τον διανύσουμε όσες φορές θέλουμε, ας πούμε i. Άρα η λέξη $uv^iw \in L$, όπως οφείλαμε να δείξουμε.
□

Πώς χρησιμοποιούμε το Λήμμα Άντλησης για να δείξουμε ότι η γλώσσα L_1 δεν είναι κανονική; Υποθέτουμε ότι είναι και καταλήγουμε σε άτοπο. Αν είναι λοιπόν, υπάρχει, από το Λήμμα Άντλησης, ένας φυσικός αριθμός



Σχήμα 18. Το μονοπάτι που αντιστοιχεί στη λέξη uvw

n, τέτοις ώστε αν $z \in L_1$ και $|z| \ge n$ τότε ισχύουν τα συμπεράσματα του Λήμματος. Έχουμε $z = 0^n 1^n \in L_1$ και $|z| \ge n$, άρα $z = uvw$, με $|uv| \le n$ και μη κενό v, τέτοια ώστε για κάθε $i = 0, 1, \ldots$ έχουμε $uv^i w \in L_1$. Αυτό όμως δε γίνεται μια και η λέξη $uv^2w = uvvw$ έχει σίγουρα περισσότερα 0 απ΄ ότι 1, αφού, λόγω του ότι $|uv| \le n$, και το u και το v έχουν μόνο μηδενικά μέσα τους.

Ας δούμε τώρα άλλη μια εφαρμογή του λήμματος άντλησης σε παρόμοιο πρόβλημα. Για κάθε λέξη $x = \alpha_1 \cdots \alpha_n$, $\alpha_i \in \Sigma$, ορίζουμε την *αντεστραμμένη* λέξη $x^R = \alpha_n \cdots \alpha_1$, να είναι η x με αντεστραμμένη τη σειρά των γραμμάτων του. Ορίζουμε τη γλώσσα

$$L_2 = \{x \in (0+1)^* : \quad x = x^R\},$$

να απαρτίζεται από όλες εκείνες τις λέξεις που διαβάζονται το ίδιο από αριστερά και από τα δεξιά.

Δείχνουμε τώρα ότι η L_2 δεν είναι κανονική. Έστω ότι είναι και έστω n ο φυσικός αριθμός του οποίου η ύπαρξη προκύπτει από το Λήμμα Άντλησης. Ορίζουμε τη λέξη $z = 1^n 0 1^n$ που είναι στην L_2. Γράφεται τότε η z ως $z = uvw$, με μη κενό v και $|uv| \le n$, οπότε και οι λέξεις u, v έχουν μέσα μόνο 1, ώστε $uv^i w \in L_2$, για $i = 0, 1, 2, \ldots$. Αλλά προφανώς η λέξη $uw \notin L_2$ αφού αφαιρώντας το v αφαιρέσαμε κάποιους άσσους από την αριστερή ομάδα άσσων αλλά όχι από τη δεξιά ομάδα, οπότε έχουμε καταλήξει σε άτοπο, άρα η L_2 δεν είναι κανονική.

Άσκηση 49. Δείξτε ότι η γλώσσα $\{xx : \quad x \in \{0,1\}^*\}$ δεν είναι κανονική.

Άσκηση 50. Ποιες από τις παρακάτω γλώσσες του $(0+1)^*$ είναι κανονικές;

1. $\{0^{2n} : \quad n \ge 1\}$
2. $\{0^m 1^n 0^{m+n} : \quad m, n \ge 1\}$
3. Οι λέξεις που δεν έχουν τρία διαδοχικά μηδενικά
4. Λέξεις με τόσα μηδενικά όσα και άσσους.
5. $\{xwx^R : \quad x, w \in (0+1)^+\}$

Κεφάλαιο 4

Αλγόριθμοι για DFA.

4.1 Πότε ένα DFA αναγνωρίζει κενή ή άπειρη γλώσσα

Δοθέντος ενός DFA M καλούμαστε να αποφασίσουμε *με αλγοριθμικό τρόπο* για το αν

1. Υπάρχει έστω και μία λέξη που αναγνωρίζεται από το M,
2. Υπάρχουν άπειρες λέξεις που αναγνωρίζονται από το M.

Και στις δύο περιπτώσεις καλούμαστε να απαντήσουμε απλώς με ένα ναι ή όχι, και δε ζητούμε να βρούμε μία ή περισσότερες (πόσο μάλλον άπειρες) λέξεις για να αποδείξουμε τα λεγόμενά μας.

Πρέπει δηλ. να βρούμε ένα τρόπο να απαντήσουμε αν η γλώσσα του M είναι κενή ή όχι, και αν είναι άπειρη ή όχι ($L(M) = \emptyset$; και $|L(M)| = \infty$;). Και πρέπει ο τρόπος απόφασης να είναι αλγοριθμικός, δηλ. να μπορούμε να γράψουμε ένα πρόγραμμα στον υπολογιστή το οποίο, όποια και να είναι η απάντηση, να μπορεί να τη βρίσκει. Αυτό σημαίνει ότι σε κάθε περίπτωση (όποια και να είναι η απάντηση) πρέπει το πρόγραμμα αυτό (α) να τελειώσει και (β) να δώσει τη σωστή απάντηση.

Η απαίτηση για το (β) είναι προφανής στον περισσότερο κόσμο αλλά δεν είναι ίσως φανερό τι εννοούμε με το (α). Είναι λοιπόν χρήσιμο να τονίσουμε ότι ο ακόλουθος αλγόριθμος για να αποφασίσουμε το ερώτημα $L(M) = \emptyset$ δεν είναι αποδεκτός:

> Με τον υπολογιστή μας απαριθμούμε όλες τις λέξεις του Σ^* ως εξής. Πρώτα απαριθμούμε όλες τις λέξεις μήκους 0 (υπάρχει μόνο μία, η κενή λέξη ϵ), μετά απαριθμούμε τις λέξεις μήκους 1 (υπάρχουν τόσες όσα και α γράμματα του Σ, δηλ. το πλήθος τους είναι $|\Sigma|$), μετά τις λέξεις μήκους 2 (υπάρχουν ακριβώς $|\Sigma|^2$ τέτοιες), κ.ο.κ. Με αυτό τον τρόπο απαρίθμησης διανύουμε όλες τις λέξεις του Σ^*, χωρίς να ξαχνάμε καμιά. Για κάθε λέξη θα έρθει κάποτε η σειρά να την εξετάσουμε. *Παράλληλα* με την απαρίθμηση προγραμματίζουμε τον υπολογιστή μας να εξετάζει κάθε μια από τις αριθμούμενες λέξεις για το αν περνάνε από το M ή όχι. Αν έστω και μια βρεθεί που να αναγνωρίζεται από το M τότε σταματάει ο αλγόριθμος και απαντά ΝΑΙ (στο ερώτημα αν $L(M) = \emptyset$), αλλιώς συνεχίζει.

Είναι φανερό ότι αυτή η μέθοδος κάνει, κατά κάποιο τρόπο, μισή δουλειά, μια και δεν είναι ποτέ δυνατό να απαντήσει ΟΧΙ, αφού δεν ξέρουμε, όσες λέξεις και να έχουμε δει μέχρι στιγμής, για το αν υπάρχει λέξη της $L(M)$ που να βρίσκεται παρακάτω στην αρίθμησή μας, π.χ. να έχει μεγαλύτερο μήκος απ΄ ό,τι έχουμε εξετάσει μέχρι στιγμής. Αν η απάντηση στο ερώτημα είναι ΝΑΙ τότε, αργά ή γρήγορα, ο αλγόριθμός μας θα απαντήσει ΝΑΙ, δε συμβαίνει όμως το ίδιο και με το ΟΧΙ.

Πώς θα μπορούσε κάπως να διορθωθεί ο αλγόριθμος που μόλις περιγράψαμε;

Αν είχαμε ένα τρόπο, με δεδομένη την περιγραφή του M, να ξέρουμε πόσο μεγάλη (το πολύ) είναι η μικρότερη σε μήκος λέξη της $L(M)$, αν μια τέτοια λέξη υπάρχει, τότε θα προγραμματίζαμε τον υπολογιστή μας να σταματάει την απαρίθμηση όταν έχει περάσει αυτό το φράγμα και δεν έχει βρει ακόμη λέξη της $L(M)$, αφού είναι πλέον σίγουρο ότι όσο και να συνεχίσει δε θα βρει άλλη. Αυτό το σκοπό, και για τα δύο ερωτήματα που μας απασχολούν, εξυπηρετεί το παρακάτω θεώρημα, που είναι ουσιαστικά πόρισμα του Λήμματος Άντλησης.

Θεώρημα 9. *Αν M είναι ένα DFA με n καταστάσεις τότε*

1. *$L(M) \neq \emptyset$ αν και μόνο αν υπάρχει $w \in L(M)$ με $|w| < n$.*
2. *$|L(M)| = \infty$ αν και μόνο αν υπάρχει $w \in L(M)$ με $n \leq |w| < 2n$.*

Απόδειξη. 1. Αν υπάρχει $w \in L(M)$ με $|w| < n$ τότε προφανώς $L(M) \neq \emptyset$. Αντίστροφα, έστω $L(M) \neq \emptyset$ και $w \in L(M)$ έχει ελάχιστο μήκος. Αν $|w| \geq n$ τότε, από το Λήμμα Άντλησης, υπάρχει σπάσιμο $w = uvz$ με $v \neq \epsilon$ τέτοιο ώστε για κάθε $i \geq 0$ έχουμε $uv^iz \in L(M)$. Για $i = 0$ παίρνουμε $uz \in L(M)$, η οποία όμως λέξη έχει μήκος μικρότερο της $w = uvz$ η οποία είχε εξ αρχής υποτεθεί ότι έχει ελάχιστο μήκος, πράγμα άτοπο, άρα $|w| < n$.

2. Αν υπάρχει $w \in L(M)$ με $n \leq |w| < 2n$ τότε (χρησιμοποιώντας μόνο την ανισότητα $n \leq |w|$) από το Λήμμα Άντλησης το w σπάει σε $w = uvz$, $v \neq \epsilon$, έτσι ώστε οι λέξεις uv^iz, $i \geq 0$, ανήκουν όλες στην $L(M)$. Αλλά αυτές είναι άπειρες το πλήθος, άρα $|L(M)| = \infty$. Αντίστροφα, έστω $|L(M)| = \infty$ και υποθέσουμε ότι δεν υπάρχει λέξη w με μήκος από n έως και $2n - 1$, ας πάρουμε w να είναι λέξη με ελάχιστο μήκος μεγαλύτερο ή ίσο του $2n$ (τέτοιες λέξεις υπάρχουν αναγκαστικά αφού η $L(M)$ έχει υποτεθεί άπειρη γλώσσα). Το Λήμμα Άντλησης εφαρμόζεται και πάλι αφού $|w| \geq 2n \geq n$ άρα η λέξη w γράφεται $w = uvz$, με $|uv| \leq n$ και $v \neq \epsilon$ και $uv^iz \in L(M)$ (για $i \geq 0$). Άρα $uz \in L(M)$ και αφού $|uvz| \geq 2n$ και $|v| \leq n$ συμπεραίνουμε ότι $|uz| \geq n$, άρα (έχουμε υποθέσει ότι δεν υπάρχουν λέξεις στην $L(M)$ με μήκος από n έως και $2n - 1$) $|uz| \geq 2n$, πράγμα που αντιφάσκει με το ότι η w έχει ελάχιστο μήκος ανάμεσα στις λέξεις της $L(M)$ με μήκος τουλάχιστον $2n$.
□

Είμαστε τώρα σε θέση να δείξουμε το ακόλουθο αποτέλεσμα.

Θεώρημα 10. *Τα ακόλουθα ερωτήματα είναι αλγοριθμικά αποφασίσιμα:*

1. *Αναγνωρίζει το DFA M μια μη κενή γλώσσα;*
2. *Αναγνωρίζει το DFA M μια άπειρη γλώσσα;*
3. *Αναγνωρίζουν τα DFA M_1 και M_2 την ίδια γλώσσα;*

Απόδειξη. 1. Απαριθμούμε όλες τις λέξεις μήκους μέχρι και n με γράμματα από το Σ (υπάρχουν ακριβώς $1 + |\Sigma| + |\Sigma|^2 + \cdots + |\Sigma|^n$ τέτοιες λέξεις) και τις ελέγχουμε όλες αν περνάνε από το αυτόματο M. Αν έστω και μια από αυτές αναγνωρίζεται τότε απαντάμε ΝΑΙ, αλλιώς απαντάμε ΟΧΙ. Η μέθοδος είναι σωστή με βάση το Θεώρημα 9.1.

2. Απαριθμούμε όλες τις λέξεις μήκους από n έως και $2n - 1$ με γράμματα από το Σ. Αν έστω και μια από αυτές αναγνωρίζεται τότε απαντάμε ΝΑΙ, αλλιώς απαντάμε ΟΧΙ. Η μέθοδος είναι σωστή με βάση το Θεώρημα 9.2.

3. Θέλουμε να απαντήσουμε στο ερώτημα

$$L(M_1) \neq L(M_2).$$

Η απάντηση σε αυτό είναι ΝΑΙ αν και μόνο αν η ακόλουθη γλώσσα είναι μη κενή

$$(L(M_1) \cap L(M_2)^c) \cup (L(M_1)^c \cap L(M_2)). \tag{4.1}$$

Μπορούμε όμως αλγοριθμικά να κατασκευάσουμε ένα DFA M που αναγνωρίζει ακριβώς αυτή τη γλώσσα, οπότε μετά χρησιμοποιούμε τον αλγόριθμο του σκέλους 1 αυτού του Θεωρήματος για να αποφασίσουμε αν η γλώσσα της (4.1) είναι κανή ή όχι. Το αυτόματο M κατασκευάζεται ως εξής. Παρατηρούμε ότι η (4.1) γράφεται

$$(L(M_1)^c \cup L(M_2))^c \cup (L(M_1) \cup L(M_2)^c)^c$$

και ότι αυτή η έκφραση χρησιμοποιεί μόνο ενώσεις και συμπληρώματα. Αν έχουμε ένα DFA N και θέλουμε να φτιάξουμε ένα DFA N' για τη συμπληρωματική γλώσσα τότε απλά αλλάζουμε τις τελικές καταστάσεις του N σε μη τελικές και τις μη τελικές σε τελικές. Για να φτιάξουμε ένα DFA για την ένωση δύο γλωσσών (των οποίων ξέρουμε κάποια DFA) φτιάχνουμε πρώτα ένα ϵ-NFA για αυτή την ένωση και το μετατρέπουμε στη συνέχεια σε DFA. Όλα αυτά γίνονται αλγοριθμικά.
□

4.2 Σχέσεις ισοδυναμίας για γλώσσες και αυτόματα. Θεώρημα Myhill–Nerode

Πάνω στο σύνολο Σ^* όλων των λέξεων ορίζονται φυσιολογικά οι εξής σχέσεις ισοδυναμίας

1. Αν $L \subseteq \Sigma^*$ είναι μια γλώσσα η σχέση R_L ορίζεται ως: $x\ R_L\ y$ αν και μόνο αν για κάθε $z \in \Sigma^*$ έχουμε
$$xz \in L \Leftrightarrow yz \in L,$$
δηλ. τα xz, yz είτε ανήκουν και τα δυο στην L είτε κανένα.

2. Αν M είναι ένα DFA τότε ορίζουμε $x\ R_M\ y$ αν και μόνο αν
$$\delta(q_0, x) = \delta(q_0, y),$$
όπου q_0 είναι η αρχική κατάσταση του M και $\delta(\cdot,\cdot)$ είναη η συνάρτηση μετάβασης του M. Ισχύει δηλ. $x\ R_M\ y$ αν και μόνο αν το αυτόματο M καταλήγει στην ίδια κορυφή αφού διαβάσει το x και αφού διαβάσει το y, ξεκινώντας πάντα από την αρχική του κορυφή.

Άσκηση 51. Δείξτε ότι οι σχέσεις R_L και R_M είναι σχέσεις ισοδυναμίας.

Άσκηση 52. Δείξτε ότι η γλώσσα L είναι ένωση κάποιων κλάσεων ισοδυναμίας της R_L (ανεξάρτητα από το αν είναι η L κανονική ή όχι). Κάθε R_L-κλάση δηλ. είτε περιέχεται πλήρως στην L είτε δεν περιέχει λέξεις της L.

Ορισμός 31. *Δείκτης* μιας σχέσης ισοδυναμίας R λέγεται ο πληθάριθμος του συνόλου των κλάσεων ιδοσυναμίας της R.

Άσκηση 53. Δείξτε ότι ο δείκτης της R_M είναι πεπερασμένος και ίσος με το πλήθος των κορυφών του M που είναι προσπελάσιμες από την αρχική κορυφή q_0. Μια κορυφή v λέγεται προσπελάσιμη αν υπάρχει λέξη w τ.ώ. $\delta(q_0, w) = v$.

Ορισμός 32. Μια σχέση R ορισμένη πάνω στο Σ^* λέγεται *δεξιά αναλλοίωτη* (right invariant) αν για κάθε $x, y, z \in \Sigma^*$ ισχύει
$$x\ R\ y \Rightarrow xz\ R\ yz. \tag{4.2}$$

Άσκηση 54. Δείξτε ότι οι σχέσεις R_L και R_M που ορίσαμε παραπάνω είναι δεξιά αναλλοίωτες.

Άσκηση 55. Αν M είναι ένα DFA δείξτε ότι η γλώσσα $L(M)$ που αναγνωρίζει το M είναι μια ένωση κλάσεων ισοδυναμίας μιας σχέσης ισοδυναμίας με πεπερασμένο δείκτη. (*Υπόδειξη:* Εξετάστε την R_M.)

Θεώρημα 11. (Θεώρημα Myhill–Nerode) *Αν $L \subseteq \Sigma^*$ τα ακόλουθα είναι ισοδύναμα:*

1. *Η γλώσσα L είναι κανονική.*
2. *Η γλώσσα L είναι ένωση κάποιων κλάσεων ισοδυναμίας μιας δεξιά αναλλοίωτης σχέσης ισοδυναμίας R με πεπερασμένο δείκτη.*
3. *Η σχέση R_L έχει πεπερασμένο δείκτη.*

Απόδειξη.

$1 \Rightarrow 2$
Αν η L είναι κανονική τότε αναγνωρίζεται από ένα DFA M και το ζητούμενο είναι η Άσκηση 55.

$2 \Rightarrow 3$
Έστω ότι η γλώσσα L είναι ένωση κλάσεων μιας σχέσης ισοδυναμίας R που έχει πεπερασμένο δείκτη $t < \infty$. Θα δείξουμε ότι ο δείκτης της R_L είναι το πολύ t, άρα πεπερασμένος.

Δείχνουμε κατ' αρχήν τη συνεπαγωγή $x\ R\ y \Rightarrow x\ R_L\ y$: Έστω $z \in \Sigma^*$, $xz \in L$ και $x\ R\ y$. Επειδή R δεξιά αναλλοίωτη έπεται $xz\ R\ yz$. Άρα το yz ανήκει στην ίδια R-κλάση με το xz. Αφού όμως $xz \in L$, και η L είναι ένωση κλάσεων της R, ολόκληρη η R-κλάση του xz περιέχεται στην L, άρα $yz \in L$. Δείξαμε δηλ. ότι $xz \in L \Rightarrow yz \in L$, και ακριβώς το ίδιο προκύπτει και η αντίστροφη συνεπαγωγή, άρα $x\ R_L\ y$, όπως θέλαμε να δείξουμε.

Η συνεπαγωγή που δείξαμε ($x\ R\ y \Rightarrow x\ R_L\ y$) σημαίνει ότι κάθε κλάση της R περιέχεται εξ ολοκλήρου σε *μια* κλάση της R_L. Πράγματι, έστω $x \in K$, όπου K μια R-κλάση, και έστω $x \in S$, όπου S η R_L-κλάση του x. Δείχνουμε τότε ότι $K \subseteq S$. Έστω λοιπόν $y \in K$. Αυτό σημαίνει $x\ R\ y$ άρα και $x\ R_L\ y$, οπότε $y \in S$.

Αφού κάθε R-κλάση περιέχεται σε μια R_L-κλάση λοιπόν δε μπορούν οι R_L-κλάσεις να είναι περισσότερες από τις R-κλάσεις, γιατι τότε θα υπήρχε κάποια R_L-κλάση που δε θα περιείχε καμιά R-κλάση και θα ήταν κενή, πράγμα που εξ ορισμού δε γίνεται.

$3 \Rightarrow 1$
Υποθέτουμε τώρα ότι η R_L έχει πεπερασμένο δείκτη και δείχνουμε ότι η L αναγνωρίζεται από κάποιο DFA M, άρα είναι κανονική. Αν $x \in \Sigma^*$ συμβολίζουμε με $[x]$ την R_L-κλάση ισοδυναμίας της λέξης x.

- Ως σύνολο καταστάσεων του αυτομάτου M παίρνουμε το σύνολο Q όλων των R_L-κλάσεων, που είναι πεπερασμένο από την υπόθεση.
- Ως αρχική κατάσταση παίρνουμε την κλάση $[\epsilon]$ της κενής λέξης.
- Τη συνάρτηση μετάβασης ορίζουμε ως εξής:

$$\delta([x], \alpha) = [x\alpha], \quad (\alpha \in \Sigma). \tag{4.3}$$

- Ως τελικές κορυφές παίρνουμε εκείνες τις κλάσεις της R_L που περιέχονται στην L (δες Άσκηση 52).

Άσκηση 56. Δείξτε ότι ο ορισμός (4.3) είναι 'καλός'. Αυτό σημαίνει ότι αν $K = [x] = [y]$ και εφαρμόσουμε τον (4.3) για να ορίσουμε την κορυφή $\delta(K, \alpha)$ τότε παίρνουμε το ίδιο αποτέλεσμα είτε χρησιμοποιήσουμε το στοιχείο x στον (4.3) είτε το y.

Επαναλαμβάνοντας τον ορισμό (4.3) (ή εφαρμόζοντας επαγωγή στο μήκος της λέξης $w \in \Sigma^*$) βλέπουμε ότι

$$\delta([x], w) = [xw],$$

άρα $\delta([\epsilon], w) = [\epsilon w] = [w]$ που είναι τελική κατάσταση του M αν και μόνο αν $w \in L$, άρα το M αναγνωρίζει ακριβώς τη γλώσσα L.
$\square$

4.3 Ελαχιστοποίηση DFA

Το ερώτημα που μας απασχολεί εδώ είναι πώς, δοθέντος ενός DFA M, να βρούμε ένα άλλο DFA, έστω N, που να είναι *ισοδύναμο* με το M (δηλ. $L(M) = L(N)$ – αναγνρίζουν την ίδια γλώσσα) και να έχει τον ελάχιστο αριθμό καταστάσεων.

Εξετάζοντας την απόδειξη του θεωρήματος Myhill-Nerode βλέπει κανείς εύκολα ότι υπάρχει ουσιαστικά ένα μοναδικό τέτοιο ελάχιστο DFA, και είναι αυτό που ως σύνολο καταστάσεών του έχει το σύνολο των κλάσεων ισοδυναμίας της δεξιά αναλλοίωτης σχέσης ισοδυναμίας R_L, με $L = L(M)$, και ως συνάρτηση μετάβασης τη συνάρτηση $\delta([x], \alpha) = [x\alpha]$ (δείτε την απόδειξη του Θεωρήματος Myhill-Nerode) όπου $[x]$ συμβολίζει την κλάση της λέξης $x \in \Sigma^*$ και $\alpha \in \Sigma$. Είδαμε επίσης στην απόδειξη του θεωρήματος Myhill-Nerode ότι οι κλάσεις της σχέσης R_M είναι υποσύνολα των κλάσεων της R_L, και αυτό σημαίνει ότι, όποιο και να είναι το DFA M, το ελάχιστο αυτόματο μπορεί να προκύψει από το M αν όλες οι κορυφές του M που είναι R_L ισοδύναμες *συμπτυχθούν* σε μία κορυφή, ώστε πλέον να μην υπάρχουν στο νέο αυτόματο δύο ή περισσότερες κορυφές, που η κλάσεις τους να είναι R_L ισοδύναμες.

Η μέθοδος ελαχιστοποίησης λοιπόν αποφασίζει για το ποιες θα είναι οι καταστάσεις του ελάχιστου DFA αφού βρεί, για κάθε ζεύγος κορυφών του M, αν οι καταστάσεις αυτές είναι μεταξύ τους R_L-ισοδύναμες. Οι καταστάσεις του ελάχιστου αυτομάτου θα είναι τα σύνολα R_L-ισοδυνάμων κορυφών του M. Για να υπολογίσουμε για κάθε ζεύγος κορυφών u και v του M αν είναι R_L-ισοδύναμες θεωρούμε κατ΄ αρχήν όλα τα ζεύγη κορυφών ισοδύναμα και αν προκύψει για ένα ζεύγος κορυφών ότι δεν είναι τότε μόνο τις χωρίζουμε.

Έτσι, θέλουμε να υπολογίσουμε μια συνάρτηση $f(u, v)$ όπου u και v είναι δύο οποιεσδήποτε διαφορετικές κορυφές του M, και θέλουμε η συνάρτηση αυτή να κάνει 1 αν οι u και v ΔΕΝ είναι ισοδύναμες και 0 αν είναι. Κατ΄ αρχήν λοιπόν δίνουμε τις αρχικές τιμές,

$$\forall u, v \in Q, u \neq v, \quad f(u, v) = 0, \tag{4.4}$$

όπου Q το σύνολο των κορυφών του M.

Σύμφωνα με τον ορισμό της σχέσης ισοδυναμίας R_L δύο λέξεις x και y είναι μεταξύ τους R_L-ισοδύναμες αν όποια και να είναι η λέξη z είτε και οι δύο λέξεις xz, yz ανήκουν στην L είτε κι οι δύο δεν ανήκουν. Αμέσως αυτό μας δίνει (με επιλογή $z = \epsilon$) ότι αν $u \in F$ και $v \notin F$ τότε οι u και v δεν είναι ισοδύναμες. Αυτό μας δίνει το επόμενο βήμα του αλγορίθμου μας

$$\forall u \in F, v \notin F \quad f(u, v) = 1. \tag{4.5}$$

Είναι επίσης φανερό από τον ορισμό της R_L ότι αν για τις κορυφές u και v γνωρίζουμε ήδη ότι δεν είναι ισοδύναμες, και για τις δύο κορυφές a και b υπάρχει μια λέξη σ τέτοια ώστε $u = \delta(a, \sigma)$ και $v = \delta(b, \sigma)$ τότε και οι a, b δεν είναι μεταξύ τους R_L-ισοδύναμες. Αυτή η παρατήρηση μας δίνει τον υπόλοιπο αλγόριθμο: Ορίζουμε κατ΄ αρχήν το σύνολο S από ζεύγη κορυφών να περιέχει όλα τα ζεύγη κορυφών (u, v) με $u \in F$, $v \notin F$, δηλ. όλα τα ζεύγη κορυφών για τα οποία γνωρίζουμε ότι τα μέλη τους δεν είναι ισοδύναμα. Έστω τώρα ένα νέο σύνολο κορυφών T που αποτελείται από όλα τα ζεύγη κορυφών (a, b) για τα οποία ισχύει ακόμη $f(a, b) = 0$ (θεωρούνται δηλ. ακόμη ισοδύναμα) και για τα οποία υπάρχει ένα γράμμα $\sigma \in \Sigma$ τέτοιο ώστε το ζεύγος

$$(\delta(a, \sigma), \delta(b, \sigma)) \in S.$$

Αφού υπολογίσουμε το σύνολο αυτό T θέτουμε μετά $f(a, b) = 1$ για όλα τα $(a, b) \in T$, και τέλος αντιγράφουμε το T πάνω στο παλιό S, το οποίο και πετάμε, και συνεχίζουμε την ίδια διαδικασία (φτιάχνουμε ένα νέο T που προκύπτει από το νέο S, κ.ο.κ.), έως ότου το σύνολο T που φτιάξουμε να προκύψει κενό. Αυτό σηματοδοτεί ότι δεν έχουμε πλέον άλλη πληροφορία να αντλήσουμε και σταματάμε εδώ.

Αποδεικνύεται ότι ο αλγόριθμος αυτός δουλεύει, δηλ. όταν σταματήσει έχουμε $f(u, v) = 1$ ακριβώς για εκείνα τα ζεύγη κορυφών (u, v) με μη R_L-ισοδύναμα μέλη.

Η κατασκευή του ελάχιστου αυτομάτου γίνεται τώρα όπως στην απόδειξη του θεωρήματος Myhill-Nerode: βάζουμε από μιά κορυφή για κάθε ομάδα ισοδυνάμων κορυφών του M. Για να βρούμε που θα πάμε με το γράμμα α από μια τέτοια 'υπερκορυφή' επιλέγουμε μια οποιαδήποτε M-κορυφή της υπερκορυφής αυτής και βλέπουμε που μας πάει το α αν ξεκινήσουμε από αυτή την κορυφή στο παλιό μας αυτόματο M. Η M-κορυφή όου καταλήγουμε ανήκει σε μια υπερκορυφή του νέου μας υπό κατασκευή αυτομάτου και σε αυτή πρέπει να μεταβούμε. Μια υπερκορυφή θεωρείται τελική αν περιέχει κάποια τελική κορυφή του M (αναγκαστικά τότε θα περιέχει μόνο τελικές κορυφές του F) ενώ η αρχική υπερκορυφή είναι αυτή που περιέχει την αρχική κορυφή του M.