Θεμελιώδη Θέματα Επιστήμης Υπολογιστών 2021-22

1η σειρά γραπτών ασκήσεων (ολοκληρωμένη)

(αυτόματα – τυπικές γλώσσες – γραμματικές)

Άσκηση 1.

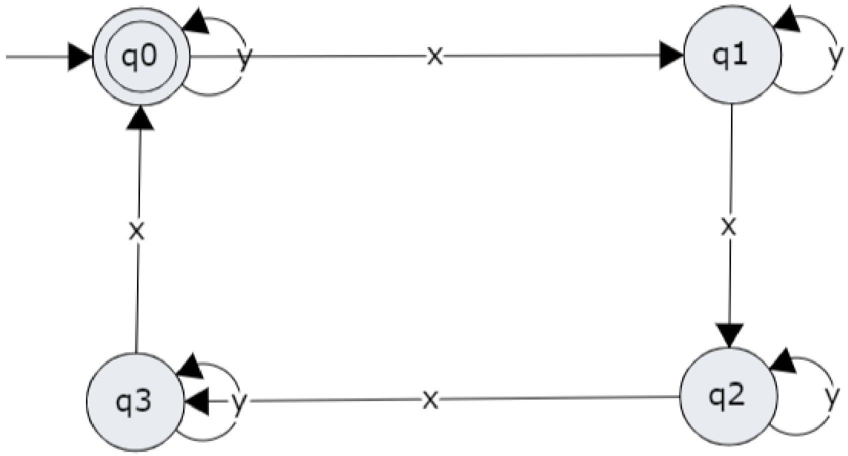
A) Υπάρχουν 4 ενδεχόμενα για το x, καθώς η διαίρεσή του με το 4 μπορεί να έχει υπόλοιπο:

- 0 : q_0 (αρχική κατάσταση)
- 1 : q_1
- 2: q_2
- 3: q_3

Άρα για το ζητούμενο DFA ισχύει: $Q_1 = \{ q_0, q_1, q_2, q_3\}$, $\Sigma_1 = \{x, y\}$, $q_{0(1)} = \{q_0\}$, $F_1 = \{q_0\}$, $\delta_1 = Q_1 \times \Sigma_1 \rightarrow Q_1$, δηλαδή :

$M_1 = \{ \{ q_0, q_1, q_2, q_3 \}, \{x, y\}, \; Q_1 \times \Sigma_1 \rightarrow Q_1, \{q_0\}, \{q_0\} \}$.

	x	y
$\rightarrow q_0$	q_1	q_0
q_1	q_2	q_1
q_2	q_3	q_2
q_3	q_0	q_3



Αποδεικνύουμε ότι το συγκεκριμένο DFA είναι ελάχιστο:

- 0 Equivalence: $\{q_1, q_2, q_3\}, \{q_0\}$
- 1 Equivalence: $\{q_1\}, \{q_2, q_3\}, \{q_0\}$
- 2 Equivalence: $\{q_1\}, \{q_2\}, \{q_3\}, \{q_0\}$

Για την κανονική παράσταση:

$q_0 = \varepsilon + y^*q_0 + x^*q_3$ Από θεώρημα $q_0 = (\varepsilon + x^*q_3)^*y^*$

$q_1 = y^*q_1 + x^*q_0$ => $q_1 = x^*q_0{}^*y^*$

$q_2 = y^*q_2 + x^*q_1$ Arden $q_2 = x^*q_1{}^*y^* = x^*(x^*q_0{}^*y^*)^*y^* = x^2(y^*)^2q_0$

$q_3 = y^*\, q_3 + x^*\, q_2$ $q_3 = x^*q_2{}^*y^* = x^*(x^2(y^*)^2\, q_0)^*y^* = x^3(y^*)^3q_0$

Συνεπώς: $q_0 = (\varepsilon + x^*q_3)^*y^* = [\varepsilon + x^*(x^3(y^*)^3q_0)]^*y^* = [\varepsilon + x^4(y^*)^3q_0]y^* =$
$= \varepsilon y^* + x^4(y^*)^4q_0 = \varepsilon y^*[x^4(y^*)^4]^* = y^*[x^4(y^*)^4]^*$

Άρα, $R_1 = [x^4(y^*)^4]^*y^*$.

Για την κανονική γραμματική:

- $V_1 = \{q_1, q_2, q_3\}$,
- $T_1 = \{q_0\}$,
- $S_1 = \{q_0\}$,
- $P_1 = \{q_0 \rightarrow y\ q_0 | x\ q_1 | \varepsilon,\ q_1 \rightarrow y\ q_1 | x\ q_2,\ q_2 \rightarrow y\ q_2 | x\ q_3,\ q_3 \rightarrow y\ q_3 | x\ q_0,\}$

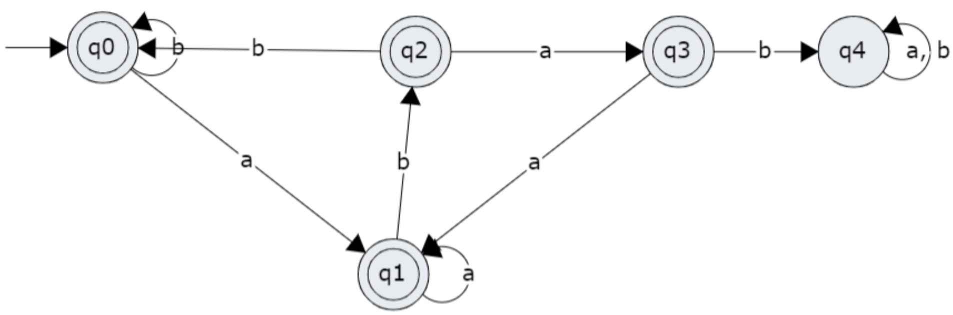
Δηλαδή: $G_1 = \{\ \{q_1, q_2, q_3\}, \{q_0\}, \{q_0\}, \{q_0 \rightarrow y\ q_0 | x\ q_1 | \varepsilon,\ q_1 \rightarrow y\ q_1 | x\ q_2,\ q_2 \rightarrow y\ q_2 | x\ q_3,\ q_3 \rightarrow y\ q_3 | x\ q_0,\}\ \}$

B) Χρησιμοποιώ τις καταστάσεις: $q_0 = \varepsilon$, $q_1 = a$, $q_2 = ab$, $q_3 = aba$, $q_4 = abab$, με σκοπό οι πρώτες τέσσερις να είναι **final states** (και πιο συγκεκριμένα, το q_0 να είναι και αρχικό **state**) ενώ το q_4 να είναι **dead state.** Το παραπάνω **DFA** περιγράφεται ως εξής:

$Q_2 = \{ q_0, q_1, q_2, q_3, q_4\}$, $\Sigma_2 = \{a, b\}$, $q_{0(2)} = \{q_0\}$, $F_2 = \{q_0, q_1, q_2, q_3\}$, $\delta_2 = Q_2 \times \Sigma_2 \rightarrow Q_2$, δηλαδή :

$M_2 = \{ \{ q_0, q_1, q_2, q_3, q_4\}, \{a, b\}, \; Q_2 \times \Sigma_2 \rightarrow Q_2, \{q_0\}, \{q_0, q_1, q_2, q_3\} \}$.

	a	b
$\rightarrow q_0$	q_1	q_0
q_1	q_1	q_2
q_2	q_3	q_0
q_3	q_1	q_4
q_4	q_4	q_4



Ελέγχουμε την ελαχιστότητα:

- 0 Equivalence: $\{q_0, q_1, q_2, q_3\}$, $\{q_4\}$
- 1 Equivalence: $\{q_0, q_1, q_2\}$, $\{q_3\}$, $\{q_4\}$
- 2 Equivalence: $\{q_0, q_1\}$, $\{q_2\}$, $\{q_3\}$, $\{q_4\}$
- 3 Equivalence: $\{q_0\}$, $\{q_1\}$, $\{q_2\}$, $\{q_3\}$, $\{q_4\}$

Άρα το **DFA** είναι ελάχιστο.

Για την κανονική παράσταση:

$q_0 = \varepsilon + b^*q_0 + b^*q_2$ Από θεώρημα $q_0 = (\varepsilon + b^*q_2)b^*$

$q_1 = a^*q_0 + a^*q_1 + a^*q_3$ Arden $q_1 = a^*(q_0 + q_3)^*a^* = a^+(b^* + (b^*)^2q_2 + q_3)$

$q_2 = b^*q_1$ => $q_2 = b^* a^+(b^* + (b^*)^2q_2 + q_3) = (b^* a^+)(q_3 + b^*)[(b^*)^3a^+]^*$

$q_3 = a^*q_2$ $q_3 = (ab^*)^2[(b^*)^3a^+]^* + a(b^* a^+)[(b^*)^3a^+]^*q_3$

$= \{(ab^*)^2[(b^*)^3a^+]^*\}\{a(b^* a^+)[(b^*)^3a^+]^*\}^*$

Συνεπώς:

$R_2 = q_0 + q_1 + q_2 + q_3 = q_3 + a(b^*)^2[(b^*)^3a^+]^* + (b^* a^+)[(b^*)^3a^+]^*q_3 +$

$+ a^+(b^* + a(b^*)^4[(b^*)^3a^+]^* + (b^*)^2(a^+)[(b^*)^3a^+]^*q_3 + q_3) +$

$+(\varepsilon + b^* (ab^*)^2[(b^*)^3a^+]^* + a(b^* a^+)[(b^*)^3a^+]^*q_3)b^* = \ldots$

Για την κανονική γραμματική:

- $V_2 = \{q_4\}$
- $T_2 = \{q_0, q_1, q_2, q_3\}$
- $S_2 = \{a, b\}$
- $P_2 = \{q_0 \rightarrow bq_0|aq_1|\varepsilon, q_1 \rightarrow aq_1|bq_2|\varepsilon, q_2 \rightarrow aq_3|bq_0|\varepsilon, q_3 \rightarrow aq_1|bq_4|\varepsilon, q_4 \rightarrow q_4\}$

Συνεπώς:

$G_2 = \{\{q_4\}, \{a, b\}, \{q_0, q_1, q_2, q_3\}, \{q_0 \rightarrow bq_0|aq_1|\varepsilon, q_1 \rightarrow aq_1|bq_2|\varepsilon, q_2 \rightarrow aq_3|bq_0|\varepsilon, q_3 \rightarrow aq_1|bq_4|\varepsilon, q_4 \rightarrow q_4\}\}$.

Άσκηση 2.

A) Ένας αριθμός μπορεί να είναι της μορφής:

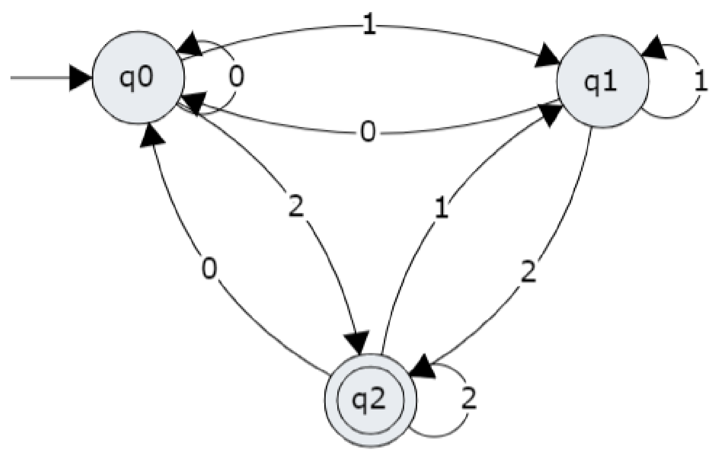
- $3m + 0$ (q_0)
- $3m + 1$ (q_1)
- $3m + 2$ (q_2)

Όταν ένας αριθμός 3m εμφανίζει ένα επιπλέον 0 στο τέλος του, ουσιαστικά τριπλασιάζεται. Όταν εμφανίζει επιπλέον 1, και τριπλασιάζεται και του προστίθεται 1, ενώ όταν εμφανίζει επιπλέον 2, και τριπλασιάζεται και του προστίθενται 2. Συνεπώς, για το DFA ισχύουν τα εξής:

$Q_1 = \{ q_0, q_1, q_2\}$, $\Sigma_1 = \{0, 1\}$, $q_{0(1)} = \{q_0\}$, $F_1 = \{q_2\}$, $\delta_1 = Q_1 \times \Sigma_1 \rightarrow Q_1$, δηλαδή :

$M_1 = \{ \{ q_0, q_1, q_2 \}, \{0, 1\}, \; Q_1 \times \Sigma_1 \rightarrow Q_1, \{q_0\}, \{q_2\} \}$.

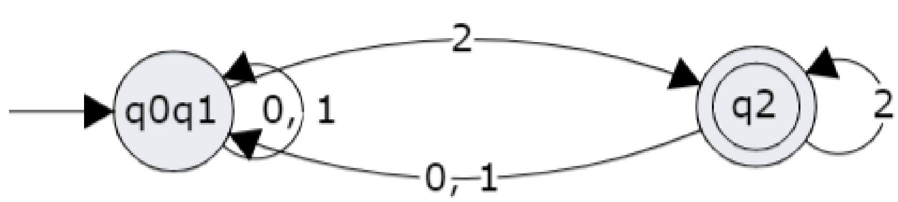
	0	1	2
$\rightarrow q_0$	q_0	q_1	q_2
q_1	q_0	q_1	q_2
q_2	q_0	q_1	q_2



Ελέγχουμε την ελαχιστότητα του DFA:

- 0 Equivalence: $\{ q_0, q_1\}, \{q_2\}$
- 1 Equivalence: $\{ q_0, q_1\}, \{q_2\}$

Συνεπώς, οι καταστάσεις $\{q_0, q_1\}$ μπορούν να συμπτυχθούν σε μια ενιαία κατάσταση q_0q_1
Άρα το DFA γίνεται:



με χαρακτηριστικά:

$M_1 = \{ \{ q_0q_1, q_2 \}, \{0, 1\}, \; Q_1 \times \Sigma_1 \rightarrow Q_1, \{q_0q_1\}, \{q_2\} \}$

και πίνακα:

	0	1	2

→ q_0q_1	q_0q_1	q_0q_1	q_2
q_2	q_0q_1	q_0q_1	q_2

Για κανονική παράσταση:

$q_0q_1 = ε + (0 + 1)q_0q_1 + (0 + 1)q_2 = (0 + 1)q_2[ε + (0 + 1)]^* = [(0 + 1) + (0 + 1)^+]q_2$

$q_2 = 2q_0q_1 + 2q_2 = 2[ε + (0 + 1) + (0 + 1)^+]q_2 = \{2[ε + (0 + 1)^*]\}^*$

Συνεπώς: $R_1 = \{2[ε + (0 + 1)^*]\}^*$

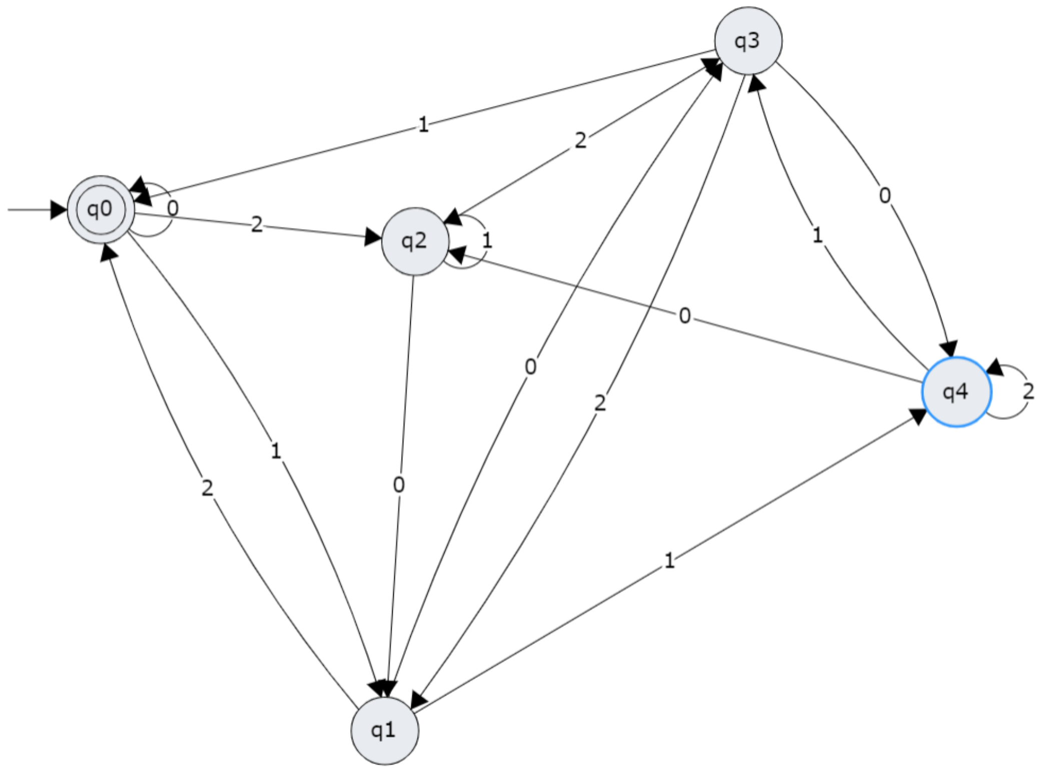
B) Όμοια με το ερώτημα (A), έχουμε 5 πιθανές καταστάσεις, στις οποίες το υπόλοιπο της διαίρεσης n/5 ισούται με:

1. 0 (q_0)
2. 1 (q_1)
3. 2 (q_2)
4. 3 (q_3)
5. 4 (q_4)

Για το αντίστοιχο DFA ισχύει: $Q_2 = \{ q_0, q_1, q_2, q_3, q_4\}$, $Σ_2 = \{0, 1, 2\}$, $q_{0(2)} = \{q_0\}$, $F_2 = \{q_0\}$, $δ_2 = Q_2 \times Σ_2 \rightarrow Q_2$, δηλαδή :

$M_2 = \{ \{ q_0, q_1, q_2, q_3, q_4\}, \{0, 1, 2\}, Q_2 \times Σ_2 \rightarrow Q_2, \{q_0\}, \{q_0\} \}$.

	0	1	2
→ q_0	q_0	q_1	q_2
q_1	q_3	q_4	q_0
q_2	q_1	q_2	q_3
q_3	q_4	q_0	q_1
q_4	q_2	q_3	q_4



Για την κανονική παράσταση:

- $q_0 = ε + 0q_0 + 2q_1 + 1\, q_3 = (ε + 2q_1 + 1\, q_3)2^*$
- $q_1 = 1q_0 + 0q_2 + 2q_3 = 1(ε + 2q_1 + 1q_3)2^* + 0q_2 + 2q_3 = 12^* + 12^+q_1 + 1^2 2^* q_3 + 0q_2 + 2q_3 = (12^* + 1^2 2^* q_3 + 0q_2 + 2q_3)(12^+)^*$
- $q_2 = 2q_0 + 1q_2 + 0\, q_4 = (ε + 2q_1 + 1\, q_3)2^+ + 1q_2 + 0q_4 = 2^+ + 2^2 2^* q_1 + 12^+ q_3 + 0q_4 =$

$= 2^+ + 1(12^+)^* 2^2(2^*)^* + 1^2(2^*)^2 2^2 (12^+)^* q_3 + 0(12^+)^* 2^2 2^* q_2 + (12^+)^* 2^3 2^* q_3 + 12^+ q_3 + 0q_4 = [2^+ + 1(12^+)^* 2^2(2^*)^* + 1^2(2^*)^2 2^2 (12^+)^* q_3 + (12^+)^* 2^3 2^* q_3 + 12^+ q_3 + 0q_4][0(12^+)^* 2^2 2^*]^*$

Χάρην ευκολίας, ονομάζω p1 την παράσταση: $p1 = [0(12^+)^* 2^2 2^*]^*$

- $q_3 = 0q_1 + 1q_4 + 2q_2 = 012^*(12^+)^* + 0(12^+)^*(1^2 2^* + 2)q_3 + 0(12+)*p1[2^+ + 1(12^+)^* 2^2(2^*)^*] + 0(12+)*p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]q_3 + 0(12+)*p10q_4 + 1q_4 + 2p1[2^+ + 1(12^+)^* 2^2(2^*)^*] + 21[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]q_3 + 20(12+)*p10q_4 =$

  $= \{012^*(12^+)^* + 0(12+)*p1[2^+ + 1(12^+)^* 2^2(2^*)^*] + 2p1[2^+ + 1(12^+)^* 2^2(2^*)^*] + [0(12+)*p10 + 1 + 20(12+)*p10]q_4\} +$

$+\{0(12^+)^*(1^2 2^* + 2) + 0(12+)^*p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+] + 21[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]\}q_3 =$

$= \{p2 + p3\, q_4\}[p4]$, όπου

$p2 = 012^*(12^+)^* + 0(12+)^*p1[2^+ + 1(12^+)^* 2^2(2^*)^*] + 2p1[2^+ + 1(12^+)^* 2^2(2^*)^*]$,

$p3 = [0(12+)^*p10 + 1 + 20(12+)^*p10]$ και

$p4 = \{0(12^+)^*(1^2 2^* + 2) + 0(12+)^*p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+] + 21[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]\}^*$

- $q_4 = 0q_3 + 1q_1 + 2q_4 = 0p2p4 + 0p3p4q_4 + 1q_1 + 2q_4$

$\rightarrow 1q_1 = 1\,(12^* + 1^2 2^* q_3 + 0q_2 + 2q_3)(12^+)^* = 1(12^+)^* 12^* + 1(12^+)^*(2 + 1^2 2^*)p2p4 + 1(12^+)^*(2 + 1^2 2^*)p3p4q_4 + 01(12^+)^* p12^+ + 01^2[(12^+)^*]^2 p12^2(2^*)^* + 01(12^+)^* p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]q_3 + 0^2 1(12^+)^* p1\, q_4 = \{1(12^+)^* 12^* + 1(12^+)^*(2 + 1^2 2^*)p2p4 + 01(12^+)^* p12^+ + 01^2[(12^+)^*]^2 p12^2(2^*)^* + 01(12^+)^* p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]p2p4\}\ \{01(12^+)^* p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]p3p4 + 1(12^+)^*(2 + 1^2 2^*)p3p4 + 0^2 1(12^+)^* p1\}q_4 =$

$p5p6q_4$, όπου

$p5 = 1(12^+)^* 12^* + 1(12^+)^*(2 + 1^2 2^*)p2p4 + 01(12^+)^* p12^+ + 01^2[(12^+)^*]^2 p12^2(2^*)^* + 01(12^+)^* p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]p2p4$ και

$p6 = 01(12^+)^* p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]p3p4 + 1(12^+)^*(2 + 1^2 2^*)p3p4 + 0^2 1(12^+)^* p1$

Άρα,

- $q_4 = 0p2p4 + 0p3p4q_4 + p5p6q_4 + 2q_4 = 0p2p4(0p3p4 + p5p6 + 2)^* = p7p8$, όπου $p7 = 0p2p4$ και $p8 = (0p3p4 + p5p6 + 2)^*$

Αντικαθιστώντας στο q_0:

$q_0 = (\varepsilon + 2q_1 + 1\, q_3)2^* = 2^* + 2^+\{(12^+)^* 12^* + (12^+)^*(2 + 1^2 2^*)p2p4 + 0(12^+)^* p12^+ + 01[(12^+)^*]^2 p12^2(2^*)^* + 0(12^+)^* p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]p2p4\}\ \{0(12^+)^* p1[1^2(2^*)^2 2^2 (12^+)^* + (12^+)^* 2^3 2^* + 12^+]p3p4 + (12^+)^*(2 + 1^2 2^*)p3p4 + 0^2(12^+)^* p1\}q_4 + 12^* p2p4 + 12^* p4p3q_4$

Ονομάζω: $p9 = 2^{+}\{(12^{+})^{*}12^{*} + (12^{+})^{*}(2 + 1^{2}2^{*})p2p4 + 0(12^{+})^{*}p12^{+} + 01[(12^{+})^{*}]^{2}p12^{2(}2^{*})^{*} + 0(12^{+})^{*}p1[1^{2}(2^{*})^{2}2^{2} (12^{+})^{*} + (12^{+})^{*}2^{3}2^{*} + 12^{+}]p2p4\}$ και

$p10 = 0(12^{+})^{*}p1[1^{2}(2^{*})^{2}2^{2} (12^{+})^{*} + (12^{+})^{*}2^{3}2^{*} + 12^{+}]p3p4 + (12^{+})^{*}(2 + 1^{2}2^{*})p3p4 + 0^{2}(12^{+})^{*}p1$, άρα το q_0 γράφεται ως εξής:

$q_0 = 2^{*} + p9p10p7p8 + 12^{*}p2p4 + 12^{*}p4p3p7p8.$

Συνεπώς:

$R_2 = 2^{*} + p9p10p7p8 + 12^{*}p2p4 + 12^{*}p4p3p7p8$, όπου τα pi (i = {1, 2, ..., 10}) έχουν καθοριστεί παραπάνω.

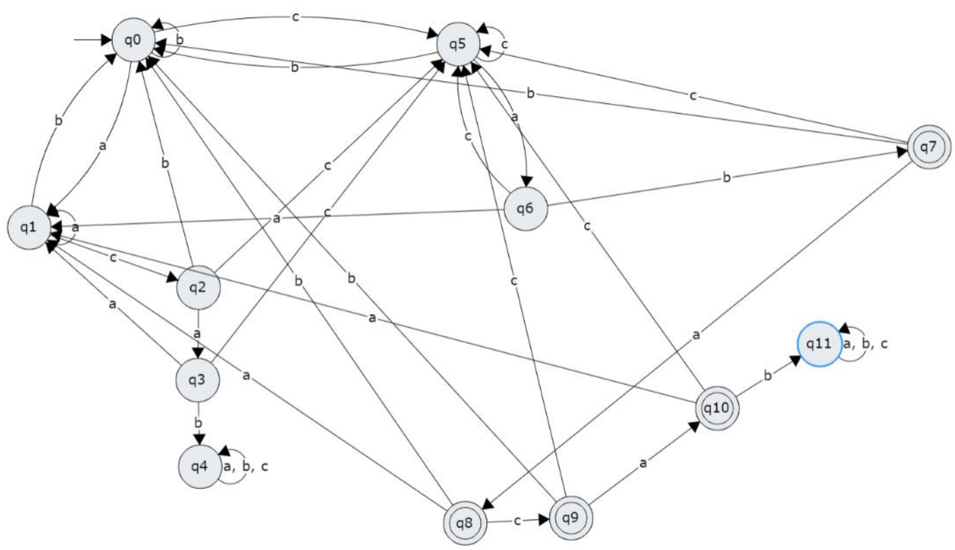
Άσκηση 3.

A) Σε αυτή την άσκηση, στην ουσία το DFA εκτελεί δύο διαδικασίες: τον εντοπισμό του cab και τον εντοπισμό του acab. Επομένως, το DFA θα πρέπει να εκτελεί την αναζήτηση του acab δύο φορές, μία πριν και μια μετά τον εντοπισμό του cab. Για τον σκοπό αυτό κατασκευάζουμε το εξής DFA:

$Q_1 = \{ q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}\}$, $\Sigma_1 = \{a, b, c\}$, $q_{0(1)} = \{q_0\}$, $F_1 = \{q_7, q_8, q_9, q_{10}\}$, $\delta_1 = Q_1 \times \Sigma_1 \rightarrow Q_1$, δηλαδή :

$M_1 = \{ \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11} \}, \{a, b, c\}, Q_1 \times \Sigma_1 \rightarrow Q_1, \{q_0\}, \{q_7, q_8, q_9, q_{10}\} \}$.

	a	b	c
$\rightarrow q_0$	q_1	q_0	q_5
q_1	q_1	q_0	q_2
q_2	q_3	q_0	q_5
q_3	q_1	q_4	q_5
q_4	q_4	q_4	q_4
q_5	q_6	q_0	q_6
q_6	q_1	q_7	q_5
q_7	q_8	q_0	q_5
q_8	q_1	q_0	q_9
q_9	q_{10}	q_0	q_5
q_{10}	q_1	q_{11}	q_5
q_{11}	q_{11}	q_{11}	q_{11}



Έλεγχος ελαχιστότητας:

- 0 Equivalence: $\{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_{11}\}, \{q_7, q_8, q_9, q_{10}\}$
- 1 Equivalence: $\{q_0, q_1, q_2, q_3, q_4, q_5, q_{11}\}, \{q_6\}, \{q_7, q_9\}, \{q_8\}, \{q_{10}\}$
- 2 Equivalence: $\{q_0, q_1, q_2, q_3, q_4, q_{11}\}, \{q_5\}, \{q_6\}, \{q_7\}, \{q_9\}, \{q_8\}, \{q_{10}\}$
- 3 Equivalence: $\{q_0, q_2, q_3\}, \{q_1, q_4, q_{11}\}, \{q_5\}, \{q_6\}, \{q_7\}, \{q_9\}, \{q_8\}, \{q_{10}\}$
- 4 Equivalence: $\{q_0\}, \{q_2\}, \{q_3\}, \{q_1\}, \{q_4\}, \{q_{11}\}, \{q_5\}, \{q_6\}, \{q_7\}, \{q_9\}, \{q_8\}, \{q_{10}\}$

Άρα το DFA είναι ελάχιστο.

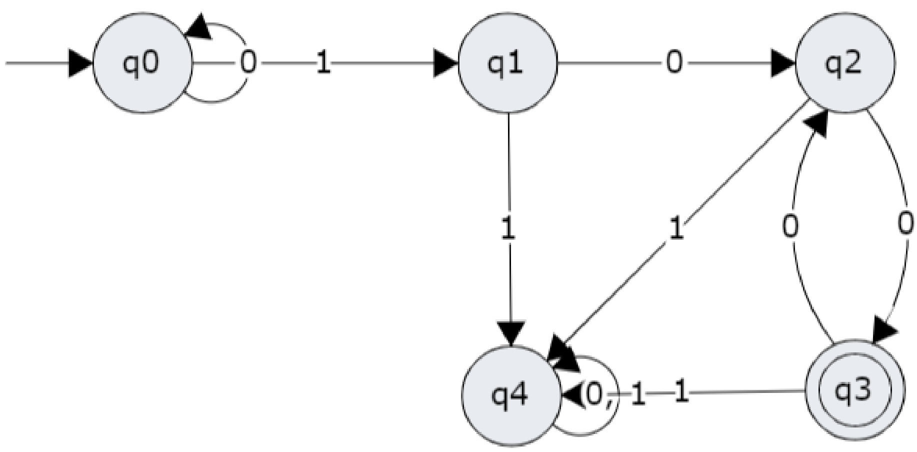
B) Στο δυαδικό σύστημα, οι αριθμοί της μορφής 4^k, αναπαρίστανται από έναν μόνο άσσο στην αρχή του αριθμού, ο οποίος ακολουθείται από άρτιο πλήθος μηδενικών.
Χρειαζόμαστε 4 καταστάσεις, τέτοιες ώστε:

- q_0 : ελέγχει τον αρχικό άσσο και, εφόσον τον εντοπίσει, πηγαίνει στο q_1
- q_1 - q_2 : ελέγχουν αν ακολουθούν μηδενικά ή άσσοι
- q_3 : ελέγχει αν το πλήθος των μηδενικών είναι άρτιο (final state)
- q_4 : dead state στο οποίο οδηγείται το DFA αν οποιαδήποτε από τις παραπάνω υποθέσεις δεν ισχύουν

Συνεπώς:

$M_2 = \{ \{ q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, Q_2 \times \Sigma_2 \rightarrow Q_2, \{q_0\}, \{q_3\} \}.$

	0	1
$\rightarrow q_0$	q_0	q_1
q_1	q_2	q_4
q_2	q_3	q_4
q_3	q_2	q_4
q_4	q_4	q_4



Έλεγχος ελαχιστότητας:

- 0 Equivalence: $\{q_0, q_1, q_2, q_4\}$, $\{q_3\}$
- 1 Equivalence: $\{q_0, q_1, q_4\}$, $\{q_2\}$, $\{q_3\}$
- 2 Equivalence: $\{q_0, q_4\}$, $\{q_1\}$, $\{q_2\}$, $\{q_3\}$
- 3 Equivalence: $\{q_0\}$, $\{q_4\}$, $\{q_1\}$, $\{q_2\}$, $\{q_3\}$

Άρα το DFA είναι ελάχιστο.

Άσκηση 4.

A) Έστω ότι η $A = \{w \in \{0, 1\}^*\}$ είναι regular language, τότε θα ισχύει το pumping lemma. Το y μπορεί να περιέχει είτε μόνο μηδενικά, είτε μόνο άσσους, είτε συνδυασμό μηδενικών και άσσων. (Παρακάτω χρησιμοποιούνται οι μεταβλητές $i, j, k \in \mathbf{N}$).

- <u>$y = 0^i$:</u>

- $x = \varepsilon$ και $z \neq 0$: η συμβολοσειρά έχει μορφή 0^i1^j, 1^j0^i, $0^i1^j0^k$, $1^j0^i1^k$ κ.ο.κ. Σε καθεμία από αυτές τις περιπτώσεις το πλήθος των μηδενικών μπορει να είναι διπλάσιο του πλήθους των άσσων. Εφόσον το pumping lemma δεν ισχύει για αυτή την περίπτωση, η γλώσσα A δεν είναι κανονική.

B) Έστω ότι η $B = \{0^n1^+0^m: n, m \geq 1, n \leq 2m\}$ είναι regular language. Οι πιθανές τιμές του y είναι:

- <u>$y = 0^i$:</u>
  - $x = \varepsilon$ και $z \neq 0$: η συμβολοσειρά παίρνει τη μορφή $(0^i)^j1^+0^m$ ή $0^n1^+(0^i)0^k \rightarrow$ το y^j μπορεί να γίνει pumped τόσες φορές, ώστε $ij \geq m$ άρα το pumping lemma δεν ισχύει.
- <u>$y = 0^i1$:</u>
  - $x = \varepsilon$ και $z \neq 0$: η συμβολοσειρά παίρνει τη μορφή $0^k(0^i1)^j1^*0^m \rightarrow$ το y^j μπορεί να γίνει pumped τόσες φορές, ώστε $ij \geq 1$, άρα καταστρατηγείται η μορφή της γλώσσας B και το pumping lemma δεν ισχύει.
- <u>$y = 0^i1^+$:</u>
  - $x = \varepsilon$ και $z \neq 0$: η συμβολοσειρά παίρνει τη μορφή $0^k(0^i1^+)^j0^m \rightarrow$ το y^j μπορεί να γίνει pumped τόσες φορές, ώστε $ij \geq 1$, άρα η συμβολοσειρά περιέχει περισσότερους από έναν "solo" άσσους, άρα καταστρατηγείται η μορφή της γλώσσας B και το pumping lemma δεν ισχύει.
- <u>$y = 1$:</u>
  - $x = \varepsilon$ και $z \neq 0$: η συμβολοσειρά παίρνει τη μορφή $0^n1^i1^*0^m \rightarrow$ το y^j μπορεί να γίνει pumped τόσες φορές, ώστε $ij \geq 1$, άρα καταστρατηγείται η μορφή της γλώσσας B και το pumping lemma δεν ισχύει.
- <u>$y = 1^+$:</u>
  - $x = \varepsilon$ και $z \neq 0$: η συμβολοσειρά παίρνει τη μορφή $0^n[(1^+)^i]^j0^m \rightarrow$ το y^j μπορεί να γίνει pumped τόσες φορές, ώστε $ij \geq 1$, άρα η συμβολοσειρά περιέχει περισσότερους από έναν "solo" άσσους, άρα καταστρατηγείται η μορφή της γλώσσας B και το pumping lemma δεν ισχύει.
- <u>$y = 1^+0^i$:</u>
  - $x = \varepsilon$ και $z \neq 0$: η συμβολοσειρά παίρνει τη μορφή $0^n(1^+0^i)^j0^k \rightarrow$ το y^j μπορεί να γίνει pumped τόσες φορές, ώστε $ij \geq 1$, άρα η συμβολοσειρά περιέχει περισσότερους από έναν "solo" άσσους, άρα καταστρατηγείται η μορφή της γλώσσας B και το pumping lemma δεν ισχύει.
- <u>$y = 1^*0^i$:</u>

- $x = \varepsilon$ και $z \neq 0$: η συμβολοσειρά παίρνει τη μορφή $0^n1(1^*0^i)^j0^k \rightarrow$ το y^j μπορεί να γίνει **pumped** τόσες φορές, ώστε $ij \geq m$ άρα το **pumping lemma** δεν ισχύει.

(όπου $i, j, k \in N$ και $ijk \geq 1$)

Εφόσον σε όλες τις δυνατές περιπτώσεις, το y μπορεί να γίνει **pumped** αρκετές φορές ώστε να μην ισχύει το **pumping lemma**, η παραγόμενη συμβολοσειρά $\notin$ **B**.

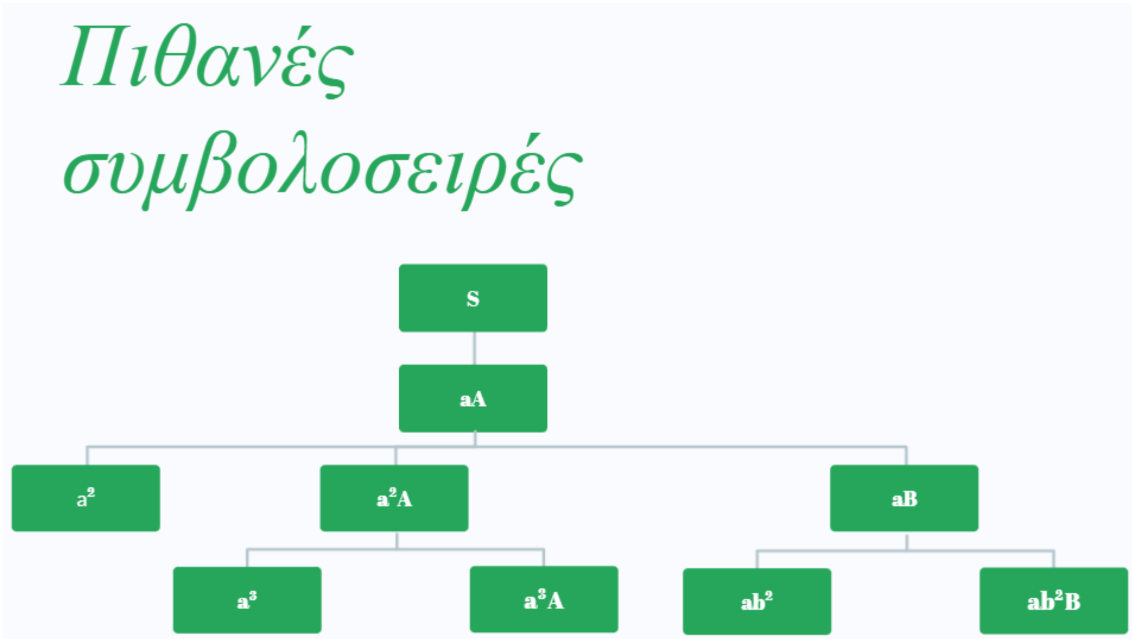
C) $C = \{0^n1^+0^m: n, m \in N\}$

- <u>$y = 0^i$ (είτε στην πριν είτε μετά το 1^+):</u> οποιαδήποτε παραγόμενη συμβολοσειρά $xy^jz \in C$.
- <u>$y = 0^i1$:</u> τότε $xy^jz = 0^{n-i}(0^i1)^j1^*0^m \rightarrow$ οποιαδήποτε παραγόμενη συμβολοσειρά $xy^jz \in C$.
- <u>$y = 1$:</u> τότε $xy^iz = 0^n1^i1^*0^m \rightarrow$ οποιαδήποτε παραγόμενη συμβολοσειρά $xy^iz \in C$.
- <u>$y = 1^+$:</u> τότε $xy^iz = 0^n(1^+)^i0^m \rightarrow$ οποιαδήποτε παραγόμενη συμβολοσειρά $xy^iz \in C$.
- <u>$y = 1^*$:</u> τότε $xy^iz = 0^n1(1^*)^i0^m \rightarrow$ οποιαδήποτε παραγόμενη συμβολοσειρά $xy^iz \in C$.
- <u>$y = 1^*0^i$:</u> τότε $xy^jz = 0^n1(1^*0^i)^j0^m \rightarrow$ οποιαδήποτε παραγόμενη συμβολοσειρά $xy^jz \in C$.

Εφόσον σε όλες τις δυνατές περιπτώσεις ισχύει το **pumping lemma**, η παραγόμενη συμβολοσειρά $\in$ **B**.

Άσκηση 5.

A) Από τις ιδιοτητες της **G** έχουμε ότι: $S \rightarrow aA$, $A \rightarrow a \mid aA \mid B$, $B \rightarrow bb \mid bbB$



Συνεπώς: $L(G) = \{a^n, a^nA, a^nb^m, a^nAb^m, a^nb^mB, a^nAb^mB, n>0, m \geq 0\}$, ή πιο απλά

$L(G) = \{a^nb^m, a^nAb^m, a^nb^mB, a^nAb^mB, n>0, m \geq 0\}$, όπου $m=2k$ ($k \in N$).

B) Αποδεικνύουμε εύκολα, μέσω του pumping lemma, ότι η $L = \{0^n1^n, n \geq 0\}$ δεν είναι regular language, άρα θα την υλοποιήσουμε με Turing Machine.

Αν το $n = 0$, τότε γίνεται λόγος για το κενό string, που είναι αποδεκτό.

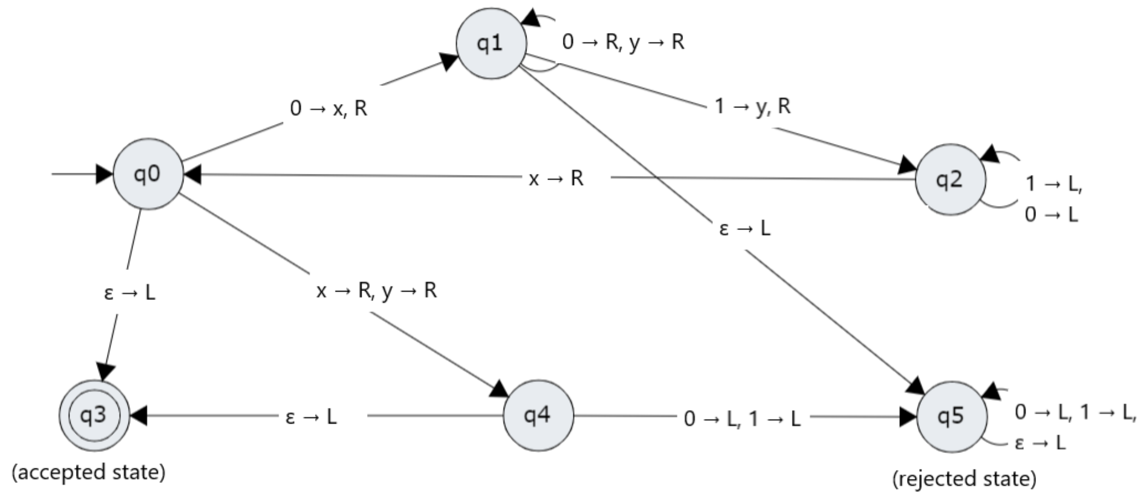
Για $n>0$: Έστω $n=3$. Θα θεωρήσουμε ε το blank space στο tape του turing machine και $\Gamma = \{x, y\}$ τη γλώσσα του tape. Το string, συνεπώς, έχει την εξής αρχική μορφή:

↓

ε	0	0	0	1	1	1	ε	...

(όπου, προφανώς, το βελάκι δείχνει στο header του tape)

Για να ελέγξω αν το πλήθος των 0 είναι ίσο με αυτό των 1, κάθε φορά που βρίσκω ένα 0 θα το αντικαθιστώ με x και το header θα μεταφέρεται στο αριστερότερο 1 που υπάρχει στο tape. Στη συνέχεια, αυτό το 1 θα αντικατασταθεί με y και το header θα μετακινηθεί στο αριστερότερο 0. Επαναλαμβάνουμε τη διαδικασία έως ότου ολα τα 0 και 1 έχουν αντικατασταθεί από x και y αντίστοιχα. Ελέγχουμε αν μετά το δεξιότερο y υπάρχουν μόνο blank spaces (ε) και, αν αυτό ισχύει, οδηγούμαστε σε accepted state. Αν αυτό δεν ισχύει, ή αν βρεθούμε σε 0 που δεν έχει αντικατασταθεί από x ενώ όλα τα 1 έχουν αντικατασταθεί από y, τότε οδηγούμαστε σε rejected state. Έτσι, προκύπτει το παρακάτω αυτόματο:



Άσκηση 6.

A) Έστω ότι οι γραμματικές $G_1 = (V_1, \Sigma_1, R_1, S_1)$ και $G_2 = (V_2, \Sigma_2, R_2, S_2)$ είναι γραμματικές χωρίς συμφραζόμενα, με ξένα $V_1 - \Sigma_1$ και $V_2 - \Sigma_2$.

- <u>Ένωση:</u> Έστω S => $G=(V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \rightarrow S_1, S \rightarrow S_2\}, S)$. Συνεπώς, $L(G)=L(G_1) \cup L(G_2)$. Εφόσον $G_1 \neq G_2$, $w \in L(G)$.
- <u>Παράθεση:</u> $L(G)=L(G_1) * L(G_2)$ για $G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \rightarrow S_1 * S_2\}, S)$.
- <u>Kleene Star:</u> $L(G_1)^*$ για $G = (V_1 \cup \{S\}, \Sigma_1, R_1 \cup \{S \rightarrow \varepsilon, S \rightarrow S * S_1\}, S)$.
-

B) Έστω $L = \{a^n b^n c^n: n \geq 0\}$ γλώσσα χωρίς συμφραζόμενα, $L = L(G)$ για γραμματική χωρίς συμφραζόμενα, $G = (V, \Sigma, R, S)$. Έστω $n > \varphi(G)/3$. Τότε $w = a^n b^n c^n \in L(G)$ και έχει μια αναπαράσταση $w = uvxyz \neq \varepsilon$ και η $uv^n x y^n z \in L(G)$ για κάθε $n = 1, 2, \ldots$

Έστω $a \in (vy)$ και $b \in (vy)$ και $c \in (vy)$: τουλάχιστον η v ή η y περιέχει 2 εμφανίσεις τουλάχιστον 2 συμβόλων ΟΜΩΣ έτσι, η $uv^n x y^n z$ περιέχει 2 εμφανίσεις σε λάθος σειρά, άρα καταστρατηγείται η μορφή της L.

Έστω ότι η (vy) περιέχει τουλάχιστον 1 εκ των 3 συμβόλων, τότε ισχύει το ίδιο για την uv^nxy^nz.

Συνεπώς, δεν ισχύει το pumping lemma, άρα η L δεν είναι κλειστή ως προς τη συμπλήρωση.

Άσκηση 7.

Γενικώς το halting problem είναι μη επιλύσιμο (undecidable) - original απόδειξη από τον Allan Turing. Έστω, όμως, ότι δεν είναι.

Θεωρούμε το πρόγραμμα Π_ε, το οποίο αποφαίνεται αν ένα πρόγραμμα Π' τερματίζει (halts, doesn't loop), αν η είσοδος είναι η κενή συμβολοσειρά. Θεωρούμε επίσης ένα πρόγραμμα Π και είσοδο x. Τοποθετούμε τα (Π, x) σε ένα Universal Turing Machine (UTM):

UTM(Π, x) → either halts or loops.

Για το πρόγραμμα Π' θεωρώ: c(x) : if (UTM(Π, x) == Halt) Loop;

Else Halt;

Αν όμως τρέξω το c <u>για</u> το c:

- If c == halt: c(c) → Loop.
- If c == loop: c(c) → Halt.

Άρα δεν μπορούμε να γνωρίζουμε αν το Π_ε τερματίζει.

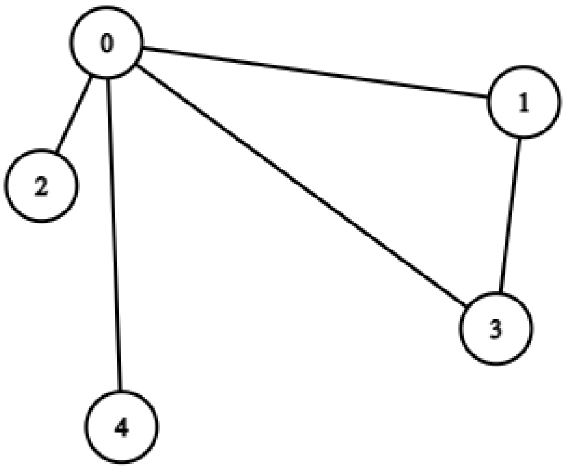
Άσκηση 8.

Ο αλγόριθμος θα λειτουργεί ως εξής:

- Για κάθε όρο $\neg X_n$ θα αντιστρέφεται η τιμή του X_n που λαμβάνεται κατά την είσοδο, δηλαδή αν X_n = True, τότε $\neg X_n$ = False και το αντίστροφο.
- Για κάθε ένωση των X_n με το σύμβολο V (= or), ελέγχουμε αν έστω ένα X_n = True, στην οποία περίπτωση όλη η ένωση ισούται με True. Διαφορετικά, η ένωση ισούται με False.
- Για κάθε ένωση των X_n με το σύμβολο Λ (= and), ελέγχουμε αν έστω ένα X_n = False, στην οποία περίπτωση όλη η ένωση ισούται με False. Διαφορετικά, ισούται με True.

Άσκηση 9.

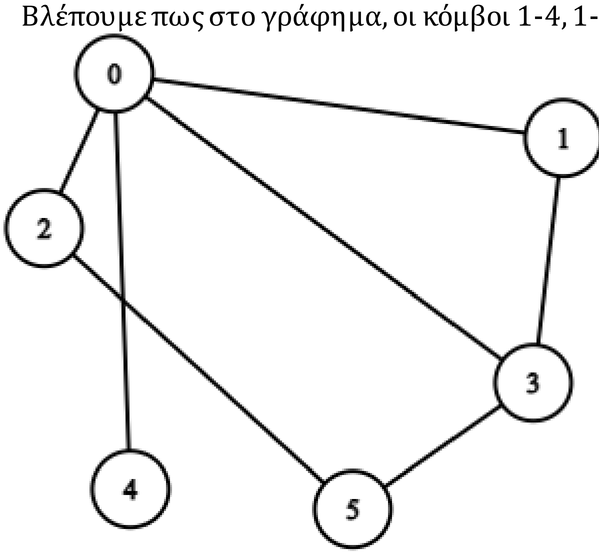
A) Θα δημιουργήσουμε ένα γράφημα G με διαφορετικές εισόδους k_1, k_2, k_3 και θα εξετάσουμε αν το halting problem για καθένα από αυτά δίνει θετική ή αρνητική απάντηση.



1. Για το παρακάτω γράφημα G, έστω ότι παίρνουμε είσοδο $k_1 = 3$. Παρατηρούμε ότι οι ακμές 2, 4 και 3 μπορούν να θεωρηθούν ανεξάρτητο σύνολο κορυφών, καθώς ανά δύο δεν συνδέονται με ακμή. Συνεπώς, το γράφημα έχει τουλάχιστον 3 κορυφές ανεξάρτητες μεταξύ τους, άρα το halting problem (G, k_1) δίνει θετική απάντηση.
2. Έστω ότι παίρνουμε είσοδο $k_2 = 10$. Δεν χρειάζεται να εξετάσουμε αν υπάρχει ανεξάρτητο σύνολο κορυφών με 10 κορυφές, καθώς το ίδιο το γράφημα G διαθέτει 5 μόνο κορυφές. Συνεπώς, το halting problem (G, k_2) έχει αρνητική απάντηση.
3. Έστω, τέλος, ότι παίρνουμε είσοδο $k_3 = 5$. Θα εξετάσουμε τον μέγιστο αριθμό κορυφών που συνιστούν ανεξάρτητο σύνολο κορυφών. Εύκολα παρατηρούμε πως η κορυφή 0 συνδέεται με όλες τις υπόλοιπες κορυφές του γράφου, ενώ οι κορυφές 1 και 3 δεν είναι μεταξύ τους ανεξάρτητες. Συνεπώς, το γράφημα G διαθέτει σύνολο 3 ανεξάρτητες κορυφές (2, 3 και 4), τρεις κατ' αριθμό. Άρα το halting problem (G, k_3) δίνει αρνητική απάντηση.

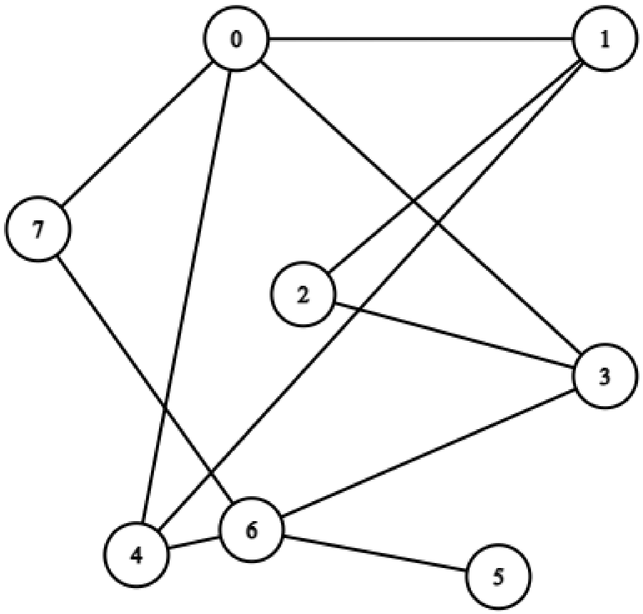
B) Θα δημιουργήσουμε 3 ανεξάρτητα γραφήματα G_1, G_2, G_3 με εισόδους k_1, k_2, k_3 και r_1, r_2, r_3 αντίστοιχα (όπου $r_i = |V_i| - k_i$, $i = \{1, 2, 3\}$), και θα εξετάσουμε αν το halting problem δίνει θετική ή αρνητική απάντηση, τόσο για το πρόβλημα IS, όσο και για το πρόβλημα VC.

1. Έστω ότι για το IS πρόβλημα έχουμε είσοδο $k_1 = 3$ και για το VC πρόβλημα $r_1 = 6 – 3 = 3$. Βλέπουμε πως στο γράφημα, οι κόμβοι 1-4, 1-2, 1-5, 0-5, 2-4, 2-3, 3-4 είναι ανεξάρτητοι.



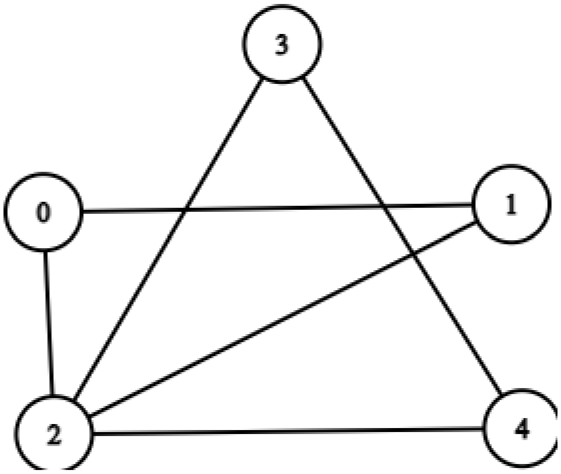
Άρα το G_1 έχει 7 ανεξάρτητα σύνολα 2 μόνο κόμβων, καθώς όλοι οι υπόλοιποι είναι εξαρτημένοι ανά δύο. Συνεπώς, για το IS πρόβλημα, το halting problem δίνει αρνητική απάντηση. Για το VC πρόβλημα, το halting problem δίνει επίσης θετική απάντηση. Αυτό συμβαίνει γιατί υπάρχει το σύνολο 0-2-3 με μέγεθος $r_1 = 3$ που είναι κάλυμμα κορυφών.

2. Έστω ότι για το IS πρόβλημα έχουμε είσοδο $k_1 = 5$, τότε για το VC πρόβλημα θα έχουμε είσοδο $r_2 = 3$. Ελέγχουμε πόσες από της κορυφές του διαγράμματος G_2 είναι ανεξάρτητες και παρατηρούμε ότι τα σύνολα 0-2, 0-5, 1-3, 1-5, 1-6, 1-7, 2-4, 2-5, 2-6, 2-7, 3-4, 3-5, 3-7, 4-



5, 4-7, 5-7 είναι ανεξάρτητα μεταξύ τους. Συμπτύσσοντάς τα σε μεγαλύτερα σύνολα, παρατηρούμε ότι τα σύνολα 0-2-5-7, 1-3-5-7, 1-6, 2-4-5-7, 2-6 είναι σύνολα ανεξάρτητων κόμβων. Εφόσον δεν υπάρχει σύνολο με 5 ανεξάρτητους κόμβους, το IS πρόβλημα δίνει αρνητική απάντηση. Επίσης, για το VC πρόβλημα παίρνουμε αρνητική απάντηση. Αυτό συμβαίνει επειδή τα καλύμματα κορυφών που υπάρχουν είναι μεγέθους τουλάχιστον 4, π.χ. 1-3-4-6.

3. Έστω ότι για το πρόβλημα IS έχουμε είσοδο $k_3 = 2$, άρα για το πρόβλημα VC έχουμε είσοδο $r_3 = 3$. Το πρόβλημα IS έχει θετική απάντηση, καθώς υπάρχουν σύνολα με τουλάχιστον 2 ανεξάρτητες κορυφές, π.χ. 1-3, 1-4… Το VC πρόβλημα έχει επίσης θετική απάντηση, καθώς υπάρχουν καλύμματα κορυφών με μέγεθος το πολύ τρία, π.χ. 0-2-4.



Συνεπώς, παρατηρούμε ότι το IS πρόβλημα και το VC πρόβλημα έχουν πάντα την ίδια απάντηση, όταν αναφέρονται σε κοινό γράφημα, με $r = |V| - k$.

C) Όπως αποδείξαμε στο ερώτημα **(B)**, ένα γράφημα G έχει IS μεγέθους k, αν και μόνο αν έχει VC μεγέθους $r = |V| - k$. Συνεπώς, το πρόβλημα VC μπορεί να ελαχιστοποιηθεί σε πρόβλημα IS, και το αντίστροφο. Άρα ισχύει πως $VC \leq_p IS$. Γνωρίζουμε πως το VC είναι NP-πλήρες και γνωρίζουμε επίσης πως, για ένα πρόβλημα απόφασης X, αν $x \in NP$ και $x \leq_p X$, τότε το X είναι NP-πλήρες. Συνεπώς, εφόσον αποδείξαμε ότι $VC \leq_p IS$ και εφόσον το VC είναι NP-πλήρες, τότε και το IS είναι NP-πλήρες.