אוניברסיטת חיפה
החוג למדעי המחשב

מודלים חישוביים

חוברת תרגולים

נכתב ע"י סרגיי ארטמנקו*

תוכן עניינים

* חוברת זו נכתבה בסיוע המלגה לזכרו של סרן אורי מיכאל עקביה ז"ל

הגדרות:

- **א"ב** Σ (סיגמה) – קבוצה סופית. לאיבר של Σ קוראים "אות".
  דוגמאות: $\{0,1\}$ - א"ב בינארי, $\{a,\dots,z\}$ - א"ב אנגלי.
- **מילה מעל א"ב** – סדרה סופית של אותיות.
  דוגמאות: 0011 - מילה מעל א"ב בינארי.
- **מילה ריקה** ε (אפסילון) – מילה שמכילה אפס אותיות.
- **גודל המילה** – מספר אותיות בה. סימון $|\ |$.

  דוגמאות: $|001| = 3, |\varepsilon| = 0$
- **שרשור של המילים** – נתונות שתי מילים w, v

$$w = w_1 w_2 \cdots w_k$$
$$v = v_1 v_2 \cdots v_m$$
$$w_i, v_j \in \Sigma$$

אזי שרשור שלהן מוגדר באופן הבא:

$$w \circ v = w_1 w_2 \cdots w_k v_1 v_2 \cdots v_m$$
$$v \circ w = v_1 v_2 \cdots v_m w_1 w_2 \cdots w_k$$

דוגמאות:

$$w = 01110, v = 111$$
$$w \circ v = 01110111$$
$$w \circ \varepsilon = \varepsilon \circ w = w = 01110$$

- **שפה** – קבוצת מילים מעל א"ב נתון.
  דוגמאות: $\{$w מחרוזת בינארית המתחילה ב- 0 $: w\}$, $\{\varepsilon, 0, 111\}$
  חשוב: $\{\varepsilon\} \neq \{\} \triangleq \varnothing$

- **שרשור של השפות** – נתונות שתי שפות L_1, L_2
  אזי שרשור שלהן מוגדר באופן הבא:

$$L_1 \circ L_2 = \{w \circ v : w \in L_1, v \in L_2\}$$

דוגמאות:

$$L_1 = \{a\}, L_2 = \{a, b\}$$
$$L_1 \circ L_2 = \{aa, ab\}$$
$$L_2 \circ L_1 = \{aa, ba\}$$
$$L_2 \circ L_2 = \{aa, ab, ba, bb\}$$
$$L_1 \circ \{\varepsilon\} = L_1$$
$$L_1 \circ \varnothing = \varnothing$$

- **חזקה של השפה** – נתונה שפה L, אזי שפה L^n כאשר n מספר שלם אי-שלילי מוגדרת באופן הבא: $L^n = \underbrace{L \circ L \circ \cdots \circ L}_{n}$ $n > 0$, $L^0 = \{\varepsilon\}$
- **Kleene star** – נתונה שפה L, נגדיר שפה חדשה L^* באופן הבא: $L^* = L^0 \cup L^1 \cup L^2 \cup \cdots$

דוגמאות:

$$L_1 = \{ab\}$$
$$L_1^* = \{\varepsilon, ab, abab, ababab, abababab, \ldots\}$$
$$\{\varepsilon\}^* = \{\varepsilon\}$$
$$\varnothing^* = \{\varepsilon\}$$

Σ^* – שפה שמכילה כל המילים מעל א"ב Σ (כולל ε).

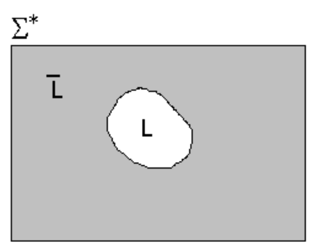
- **Kleene plus** – נתונה שפה L, נגדיר שפה חדשה L^+ באופן הבא: $L^+ = L^1 \cup L^2 \cup \cdots$

דוגמאות:

$$L_1 = \{ab\}$$
$$L_1^+ = \{ab, abab, ababab, abababab, \ldots\}$$
$$\{\varepsilon\}^+ = \{\varepsilon\}$$
$$\varnothing^+ = \varnothing$$

- **משלים של L** – נתונה שפה L, נגדיר שפה חדשה $\overline{L}$ (משלים של L) כקבוצת כל המילים שלא שייכות ל- L

$$\overline{L} = \Sigma^* \setminus L = \{w \in \Sigma^* : w \notin L\}$$



טענה 1: אם Σ לא ריקה, אזי Σ^* קבוצה אינסופית בת-מניה.

הוכחה: נגדיר A_i להיות קבוצת כל המילים מעל א"ב Σ מגודל i כאשר $i = 0,1,2,\ldots$.

$|A_i| = |\Sigma|^i$ כלומר, לכל i קבוצה A_i סופית ולכן בת-מניה. בנוסף מתקיים $\Sigma^* = \bigcup_{i=0}^{\infty} A_i$. לכן, Σ^* קבוצה בת מניה כאחוד בן מניה של קבוצות בנות מניה.

טענה 2: קבוצת כל השפות מעל א"ב לא ריק Σ היא לא בת-מניה.

הוכחה: באמצעות שיטת הלכסון.

תרגיל 1: L שפה מעל א"ב לא ריק Σ. צריך להוכיח או להפריך:

א. $(L^*)^* = L^*$

ב. $L^* = L^* \circ L^*$

ג. $L^+ = L^* \setminus \{\varepsilon\}$

פתרון:

א. הוכחה: לכל $x \in (L^*)^*$ מתקיים:

$$x \in (L^*)^* \Rightarrow \exists k \geq 0 : x \in (L^*)^k \Rightarrow \exists m_1, \ldots, m_k \geq 0 : x = w_1 \circ \cdots \circ w_k,\ w_i \in L^{m_i} \Rightarrow x \in L^{m_1+\ldots+m_k} \Rightarrow x \in L^*$$

לכן, $(L^*)^* \subseteq L^*$ ובנוסף מתקיים $L^* = (L^*)^1 \subseteq (L^*)^0 \cup (L^*)^1 \cup (L^*)^2 \cup \ldots = (L^*)^*$

מסקנה: $(L^*)^* = L^*$.

ב. הוכחה: $\varepsilon \in L^*$ לכן, לכל $x \in L^*$ מתקיים $x \in L^* \Rightarrow x = x \circ \varepsilon \in L^* \circ L^*$

לכן, $L^* \subseteq L^* \circ L^*$ ובנוסף מתקיים $L^* \circ L^* = (L^*)^2 \subseteq (L^*)^* = L^*$.

מסקנה: $L^* = L^* \circ L^*$.

ג. דוגמה נגדית: $\{\varepsilon\} = \{\varepsilon\}^+ \neq \{\varepsilon\}^* \setminus \{\varepsilon\} = \{\varepsilon\} \setminus \{\varepsilon\} = \varnothing$.

תרגיל 2: הוכיחו: אם $L \neq \varnothing$, אזי $L \subseteq L^2 \Leftrightarrow \varepsilon \in L$.

פתרון:

$\varepsilon \in L \Rightarrow L \subseteq L^2$: אם $\varepsilon \in L$, אזי $L = L \circ \{\varepsilon\} \subseteq L \circ L = L^2$

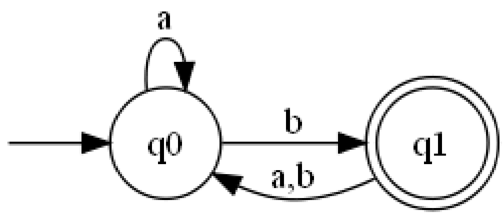
$\varepsilon \notin L \Rightarrow L \not\subseteq L^2$: מתוך כל המילים בעלות אורך מינימאלי בשפה L נבחר מילה אחת (לדוגמה, ראשונה לפי סדר לקסיקוגרפי) - w.

מתקיים $|w| > 0$. אורך של מילה מינימאלית ב- L^2 הוא $2|w|$ (כי מילה בעלת אורך מינימאלי בשפה L^2 מתקבלת כשרשור של שתי מילים בעלות אורך מינימאלי ב- L). לכן, $w \notin L^2$.

מסקנה: $L \not\subseteq L^2$.

אוטומט סופי דטרמיניסטי (אס"ד)

M:



אס"ד מורכב ממספר <u>סופי</u> של מצבים, המסומנים באופן הבא:

מצב (לא מקבל) –

מצב מקבל –

בנוסף, קיים מצב אחד שמוגדר כ**מצב התחלתי** ומסומן ב- .

<u>לכל מצב</u> באס"ד <u>ולכל אות</u> בא"ב Σ מוגדר <u>מעבר יחיד</u> (שמסומן בחץ) למצב אחר או לעצמו.

כאשר מקבלים כקלט מילה מעל Σ – מתחילים במצב ההתחלתי, קוראים מילה משמאל לימין ועוברים למצב הבא בהתאם לאות שקראנו ומעברים שמוגדרים במצב הנוכחי.

לדוגמה,

חישוב על המילה aaabbbab באס"ד M:

$$q_0 \xrightarrow{a} q_0 \xrightarrow{a} q_0 \xrightarrow{a} q_0 \xrightarrow{b} q_1 \xrightarrow{b} q_0 \xrightarrow{b} q_1 \xrightarrow{a} q_0 \xrightarrow{b} q_1$$

חישוב על המילה ריקה ε באס"ד M:

$$q_0$$

הגדרות:

- **מסלול חישוב** – סדרת המצבים המתקבלת במהלך החישוב של אס"ד על המילה.
  דוגמאות:
  מסלול חישוב של מילה aaabbbab באס"ד M הוא $(q_0, q_0, q_0, q_0, q_1, q_0, q_1, q_0, q_1)$.
  מסלול חישוב של ε באס"ד M הוא (q_0).
- **מילה מתקבלת ע"י אס"ד** – מצב האחרון במסלול חישוב של המילה הוא מצב מקבל.
  דוגמאות:
  מילה aaabbbab מתקבלת ע"י אס"ד M כי המצב האחרון במסלול חישוב שלה הוא מצב מקבל q_1.

מילה ε לא מתקבלת ע"י אס"ד M כי המצב האחרון במסלול חישוב שלה הוא מצב לא מקבל q_0.

- **שפה של אס"ד** – אוסף כל המילים מעל א"ב Σ שאס"ד מקבל.

דוגמה: מילה מעל א"ב $\Sigma = \{a,b\}$ היא בשפה של אס"ד M (סימון $L(M)$) אמ"ם היא סדרה של b-ים בגודל אי-זוגי או שהיא מסתיימת בסדרה של b-ים בגודל אי-זוגי שיש אות a לפניה.

באופן פורמאלי אס"ד מוגדר ע"י חמישייה סדורה (Q,Σ,δ,q_0,F)

Q – קבוצה סופית של מצבים.

Σ – א"ב.

$\delta: Q \times \Sigma \to Q$ – פונקצית מעבר.

q_0 – מצב התחלתי.

$F \subseteq Q$ – קבוצת מצבים מקבלים.

לדוגמה, הגדרה פורמאלית של אס"ד M שהגדרנו בצורה גרפית: $M = (\{q_0,q_1\},\{a,b\},\delta,q_0,\{q_1\})$

$$\delta(q_0,b) = q_1$$
$$\delta(q_0,a) = \delta(q_1,a) = \delta(q_1,b) = q_0$$

הגדרה:

- $\hat{\delta}(q,w)$ מצב שנמצאים בו אם התחלנו ממצב q וקראנו את המילה w.

אם $w = w_1 \cdots w_k$ אזי $\hat{\delta}(q,w) = \delta(\delta(\cdots\delta(\delta(q,w_1),w_2),\cdots),w_{k-1}),w_k)$

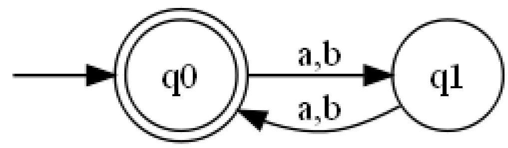
מתקיים $w \in L(M) \Leftrightarrow \hat{\delta}(q_0,w) \in F$

תרגיל 1: נתונה שפה $L_1 = \{w \in \{a,b\}^* : |w|$ זוגי $\}$

הערה: $|\varepsilon|$ זוגי

תכננו אס"ד שמקבל את השפה (כלומר, השפה שלו שווה ל-L_1).

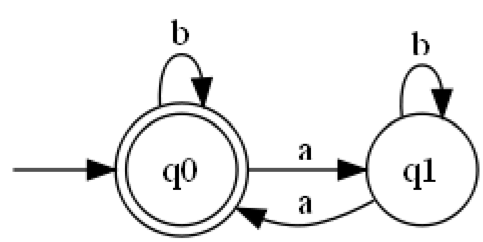
פתרון:



תרגיל 2: נתונה שפה $L_2 = \{w \in \{a,b\}^* :$ מספר a-ים ב- w זוגי $\}$

תכננו אס"ד שמקבל את השפה.

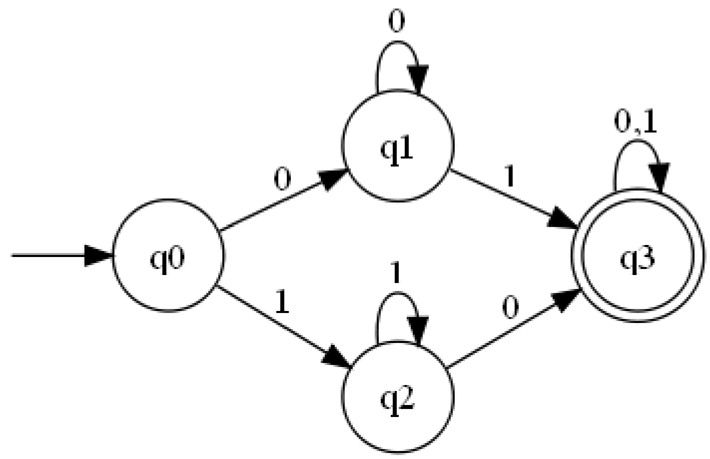
פתרון:



תרגיל 3: נתונה שפה $L_3 = \{w \in \{0,1\}^* :$ w מכילה גם 0 וגם 1 $\}$

תכננו אס"ד שמקבל את השפה.

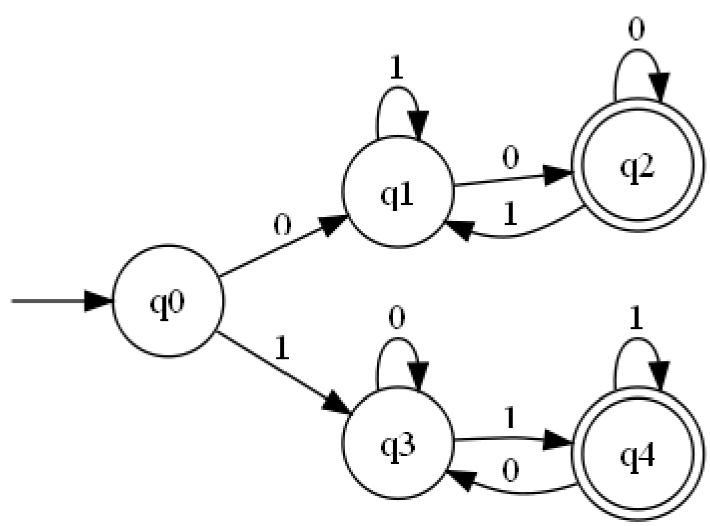
פתרון:



תרגיל 4: נתונה שפה $L_4 = \{w \in \{0,1\}^* :$ w מתחילה ומסתיימת באותו תו, $|w| > 1\}$

תכננו אס"ד שמקבל את השפה.

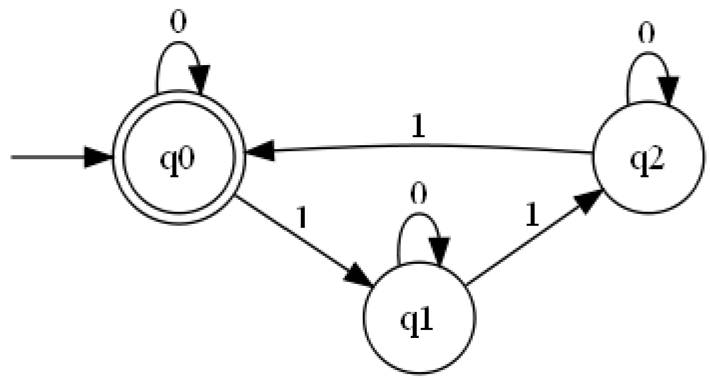
פתרון:



תרגיל 5: נתונה שפה $L_5 = \{w \in \{0,1\}^* : \sum w_i \equiv 0 \ (\text{mod}3)\}$. כלומר, מספר 1-ים מתחלק ב-3 ללא שארית.

תכננו אס"ד שמקבל את השפה <u>והוכיחו נכונות הבניה</u>.

פתרון:

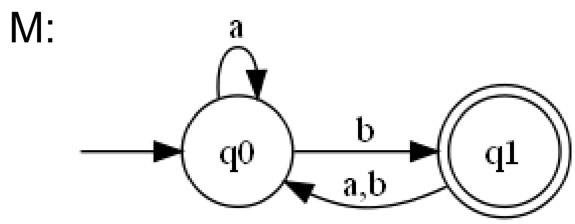


הוכחת נכונות:
נוכיח באינדוקציה על אורך המילה כי מתקיים:
$\hat{\delta}(q_0, w) = q_i$ אם מספר 1-ים ב-w מתחלק ב-3 עם שארית i.
בדיקה עבור $|w| = 0, 1$: $\hat{\delta}(q_0, \varepsilon) = q_0$; $\hat{\delta}(q_0, 0) = q_0$; $\hat{\delta}(q_0, 1) = q_1$
נניח כי הטענה נכונה עבור מילים מאורך k ונראה שאז הטענה נכונה גם למילים מאורך $k+1$.
$|w| = k+1$
מקרה א': $w = v \circ 0$
מתקיים: אם מספר 1-ים ב-w מתחלק ב-3 עם שארית i, אזי גם מספר 1-ים ב-v מתחלק ב-3 עם שארית i.
לפי הנחת האינדוקציה $|v| = k \Rightarrow \hat{\delta}(q_0, v) = q_i$
לפי בניה של אס"ד: $\hat{\delta}(q_0, w) = \hat{\delta}(q_0, v \circ 1) = \delta(\hat{\delta}(q_0, v), 1) = \delta(q_i, 0) = q_i$

מקרה ב': $w = v \circ 1$
מתקיים: אם מספר 1-ים ב-w מתחלק ב-3 עם שארית i, אזי מספר 1-ים ב-v מתחלק ב-3 עם שארית $i-1$ (מודולו 3).
לפי הנחת האינדוקציה $|v| = k \Rightarrow \hat{\delta}(q_0, v) = q_{i-1}$
לפי בניה של אס"ד: $\hat{\delta}(q_0, w) = \hat{\delta}(q_0, v \circ 1) = \delta(\hat{\delta}(q_0, v), 1) = \delta(q_{i-1}, 1) = q_i$ (חישוב מודולו 3)

מסקנה: הטענה נכונה גם למילים מאורך $k+1$ ולכן הטענה נכונה למילים מכל אורך.

תרגיל 6: נתון אס"ד M



צריך להוכיח:

$L(M)=\{w\in\{\mathrm{a,b}\}^*:$ לפניה a סדרה של b-ים בגודל אי-זוגי או מסתיימת בסדרה של b-ים בגודל אי-זוגי שיש $w\}$

פתרון:

<u>מקרה א'</u>: w לא מכילה אות a. כלומר, $w=\mathrm{b}^n$.

נוכיח באינדוקציה כי $\hat{\delta}(\mathrm{q}_0,\mathrm{b}^n)=\mathrm{q}_1$ כאשר n אי-זוגי ו- q_0 אחרת.

$$\hat{\delta}(\mathrm{q}_0,\mathrm{b}^0)=\hat{\delta}(\mathrm{q}_0,\varepsilon)=\delta(\mathrm{q}_0,\varepsilon)=\mathrm{q}_0$$

$$\hat{\delta}(\mathrm{q}_0,\mathrm{b}^1)=\hat{\delta}(\mathrm{q}_0,\mathrm{b})=\delta(\mathrm{q}_0,\mathrm{b})=\mathrm{q}_1$$

נניח כי הטענה נכונה עבור מילים מאורך k (בה"כ k אי-זוגי) ונראה שאז הטענה נכונה ל-$k+1$ ו- $k+2$.

$$\hat{\delta}(\mathrm{q}_0,\mathrm{b}^{k+1})=\delta(\hat{\delta}(\mathrm{q}_0,\mathrm{b}^k),\mathrm{b})=\delta(\mathrm{q}_1,\mathrm{b})=\mathrm{q}_0\notin \mathrm{F}$$
$$\hat{\delta}(\mathrm{q}_0,\mathrm{b}^{k+2})=\delta(\hat{\delta}(\mathrm{q}_0,\mathrm{b}^{k+1}),\mathrm{b})=\delta(\mathrm{q}_0,\mathrm{b})=\mathrm{q}_1\in \mathrm{F}$$

מסקנה: אם $w=\mathrm{b}^n=\varepsilon\circ\mathrm{b}^n$ אזי w מתקבלת אמ"ם n אי-זוגי.

<u>מקרה ב'</u>: w מכילה אות a.

טענה: לאחר שקראנו אות a נהיה במצב q_0.

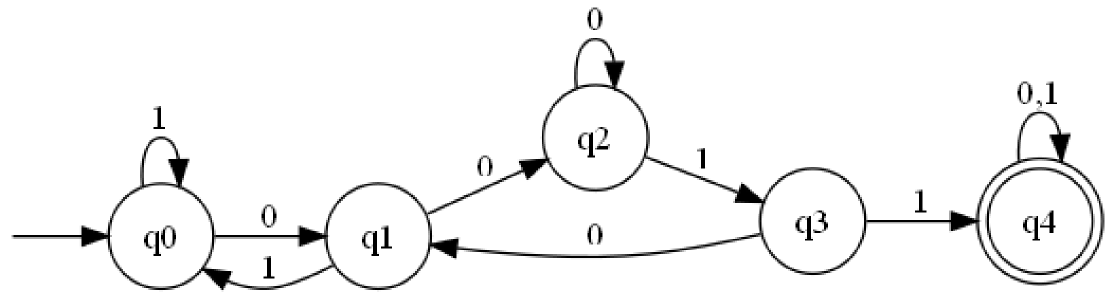
הוכחה: $\delta(q_0,a)=\delta(q_1,a)=q_0$

לכן, לאחר שקראנו אות a אחרונה ב- w נהיה במצב q_0. כעת נותר לקרוא מהקלט רק סדרה של אפס או יותר b-ים. אבל ראינו במקרה א', אם מתחילים במצב q_0 וקוראים מהקלט סדרה של b-ים נגיע למצב מקבל q_1 אמ"ם מספרם אי-זוגי.

תרגיל 7: נתונה שפה $L_7 = \{w \in \{0,1\}^* : 0011$ מכילה w $\}$

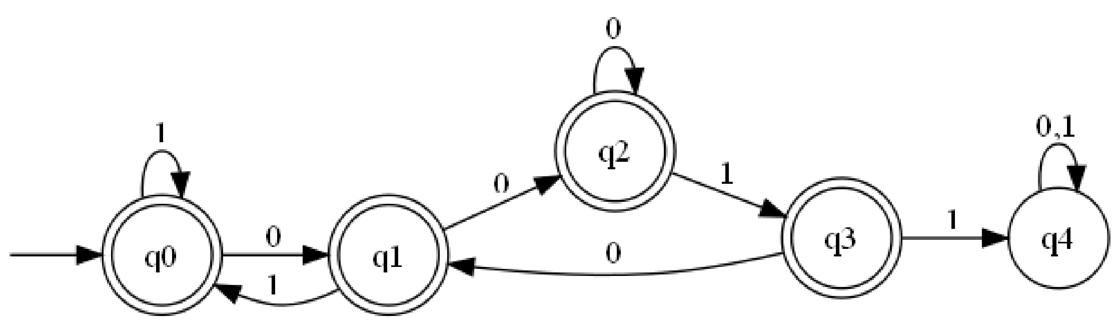
תכננו אס"ד שמקבל את השפה.

פתרון:



תרגיל 8: תכננו אס"ד שמקבל את השפה $\overline{L_7}$.

פתרון:

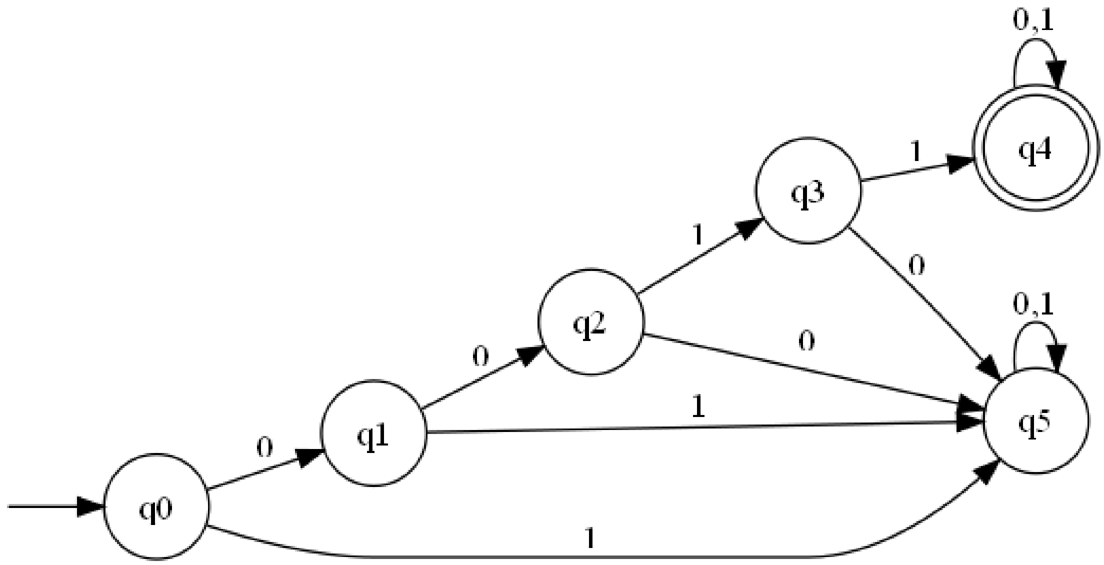


באופן כללי, אם שפה של אס"ד היא L אזי אם נגדיר כל מצב מקבל כלא מקבל ולהפך נקבל אס"ד חדש שהשפה שלו היא $\overline{L}$.

תרגיל 9: נתונה שפה $L_9 = \{w \in \{0,1\}^* :$ 0011 מתחילה ב- w $\}$

תכננו אס"ד שמקבל את השפה.

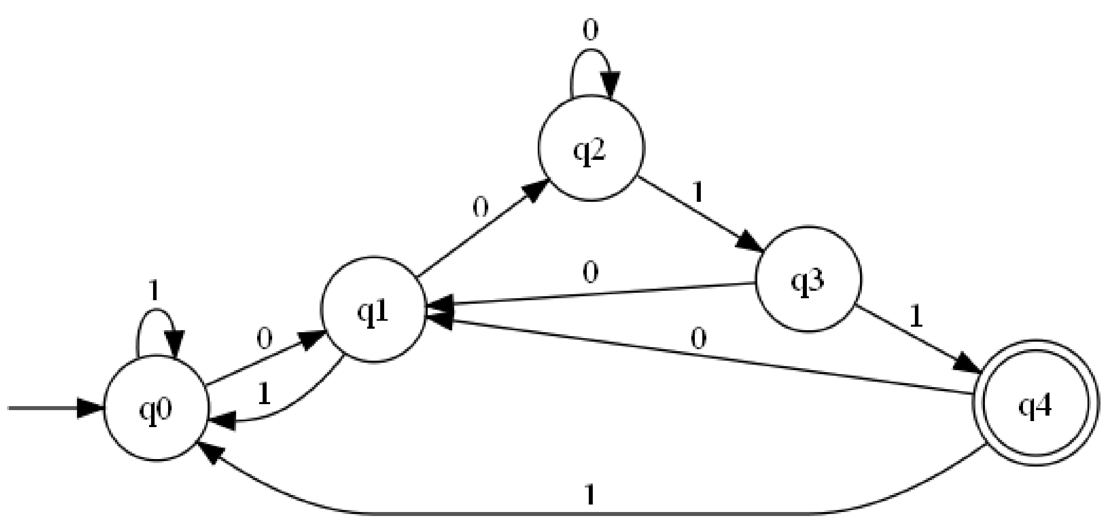
פתרון:



תרגיל 10: נתונה שפה $L_{10} = \{w \in \{0,1\}^* :$ 0011 מסתיימת ב- w $\}$

תכננו אס"ד שמקבל את השפה.

פתרון:



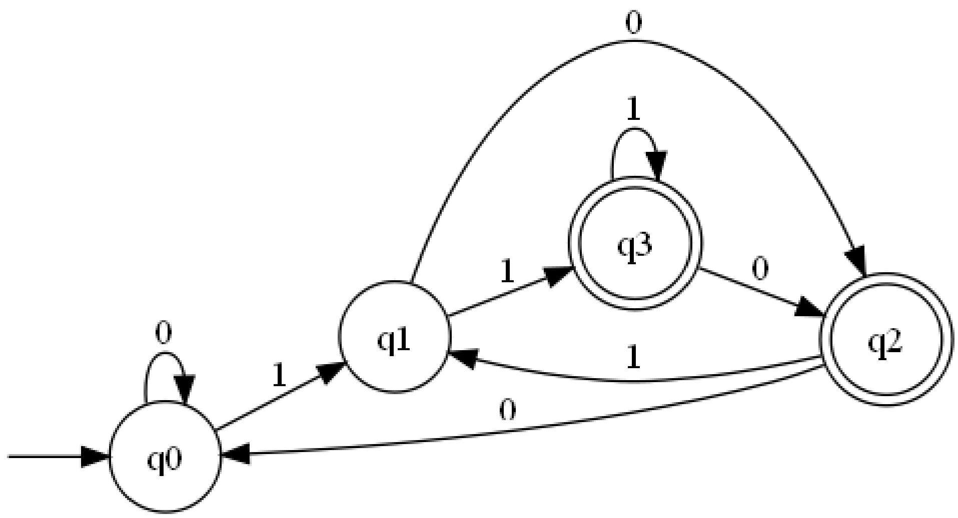
תרגיל 11:

נתונה שפה $L_{11} = \{w \in \{0,1\}^* :$ התו לפני אחרון ב-w הוא 1, $|w| > 1\}$

תכננו אס"ד שמקבל את השפה.

פתרון:

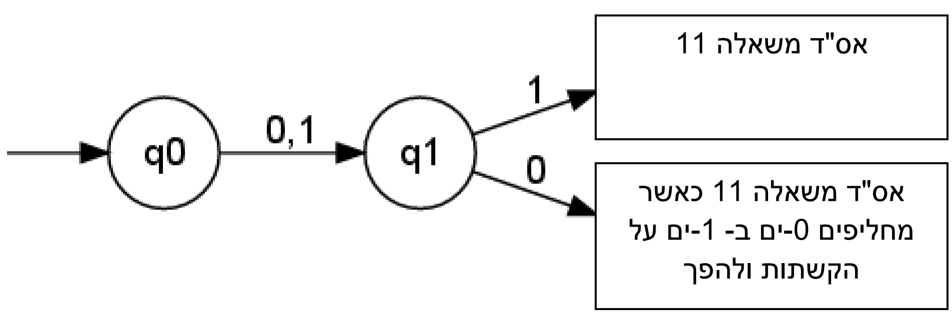
L_{11} = כל המילים שמסתיימות ב- 11, 10.



תרגיל 12: נתונה שפה $L_{12} = \{w \in \{0,1\}^* :$ התו השני מהתחלה שווה לתו השני מהסוף, $|w| > 3\}$

תכננו אס"ד שמקבל את השפה.

פתרון:

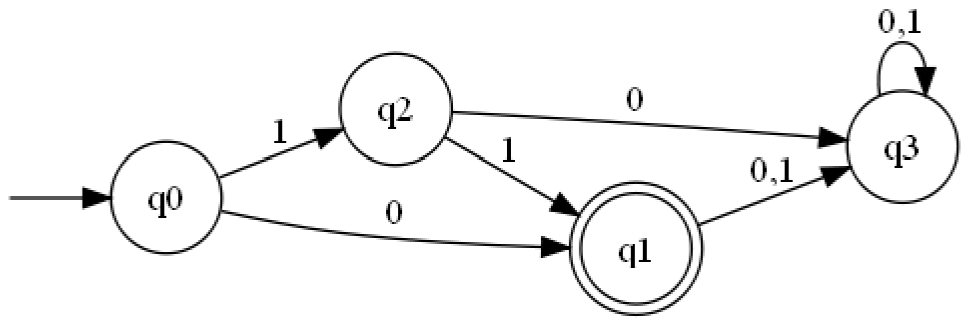


תרגיל 13: נתונה שפה $L_{13} = \{w \in \{0,1\}^* : w \neq 0,11\}$

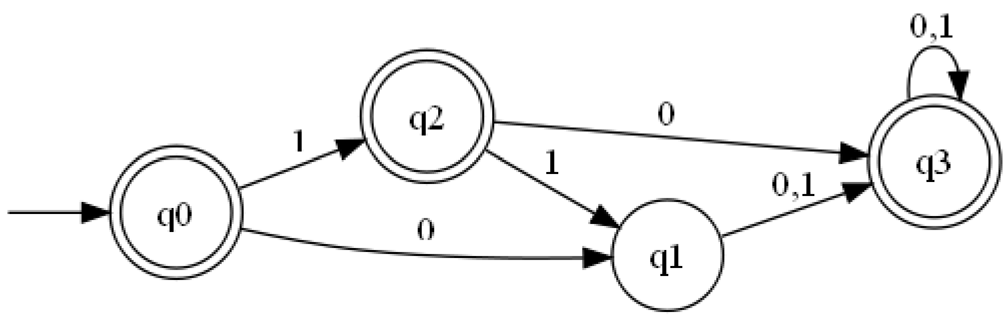
תכננו אס"ד שמקבל את השפה.

פתרון:

אס"ד לשפה $\overline{L_{13}} = \{w \in \{0,1\}^* : w = 0,11\}$:



לכן אס"ד לשפה L_{13} :

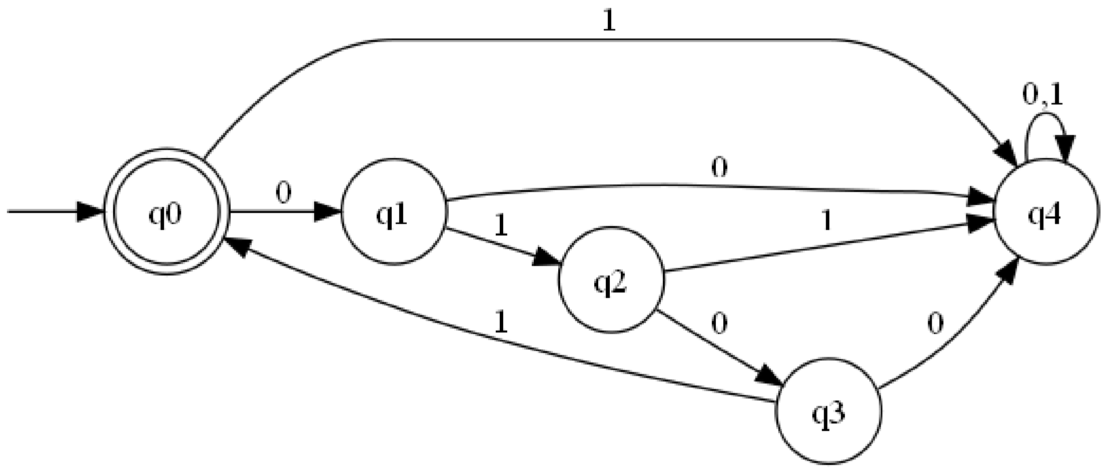


תרגיל 14: נתונה שפה $L_{14}=\{(01)^{2n}:n=0,1,2,\ldots\}$

תכננו אס"ד שמקבל את השפה.

פתרון:

דוגמה למילים בשפה: $\varepsilon,0101,01010101,010101010101,\ldots$



הגדרה:

- **שפה רגולרית** – שפה L נקראת רגולרית אם קיים אס"ד M כך שמתקיים $L(M)=L$.

משפט: שפות רגולריות סגורות תחת איחוד/חיתוך סופי.

רעיון: בהינתן מילת קלט נריץ במקביל אס"דים של השפות נתונות על המילה זו. אם בכל אחד מהם מילה התקבלה – מילה שייכת לחיתוך של השפות. אם היא התקבלה לפחות באחד מהם היא שייכת לאיחוד של השפות.

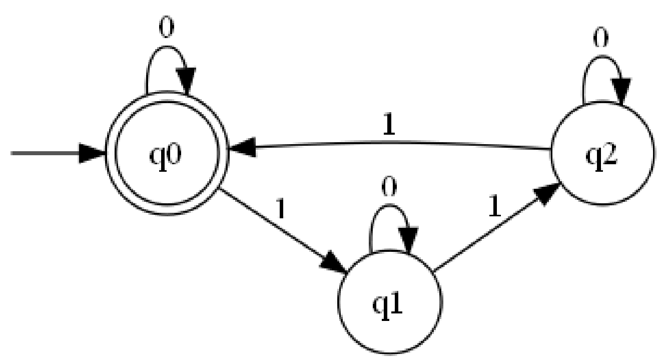
מימוש: נבנה אס"ד שמבצע הרצה במקביל של מספר סופי של אס"דים. קבוצת המצבים שלו היא מכפלה קרטזית של קבוצות המצבים של האס"דים הנתונים.

דוגמה:

נתונות שפות רגולריות:

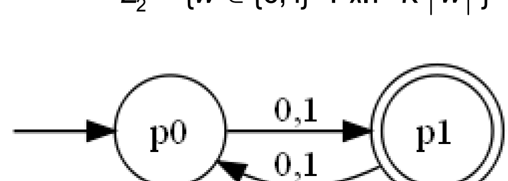
$$L_1=\{w\in\{0,1\}^*:\sum w_i\equiv 0\ (\text{mod}3)\}$$

$$M_1=(Q,\{0,1\},\delta_1,q_0,F_1)$$
$$Q=\{q_0,q_1,q_2\}$$
$$F_1=\{q_0\}$$
$$\delta_1(q_0,0)=q_0;\ \delta_1(q_1,0)=q_1;\ \delta_1(q_2,0)=q_2$$
$$\delta_1(q_0,1)=q_1;\ \delta_1(q_1,1)=q_2;\ \delta_1(q_2,1)=q_0$$



$L_2=\{w\in\{0,1\}^*:$ אי-זוגי $|w|\}$

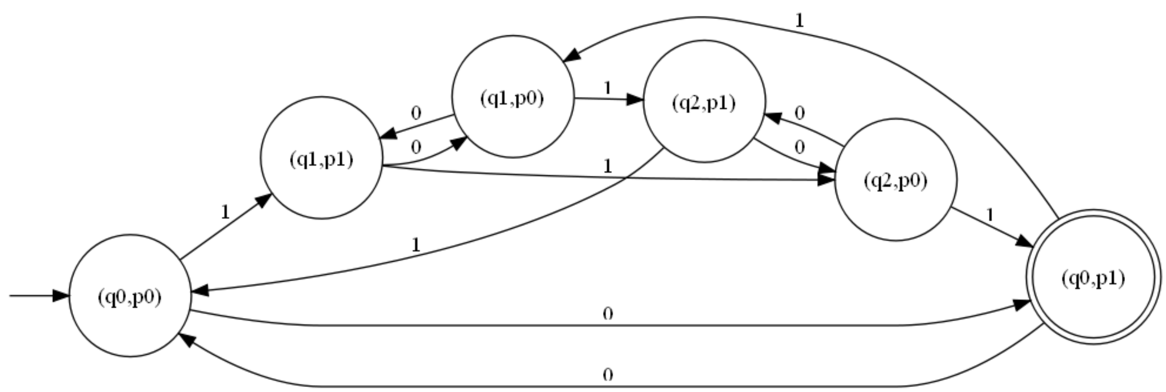
$$M_2=(P,\{0,1\},\delta_2,p_0,F_2)$$
$$P=\{p_0,p_1\}$$
$$F_2=\{p_1\}$$
$$\delta_2(p_0,0)=\delta_2(p_0,1)=p_1;\ \delta_2(p_1,0)=\delta_2(p_1,1)=p_0$$



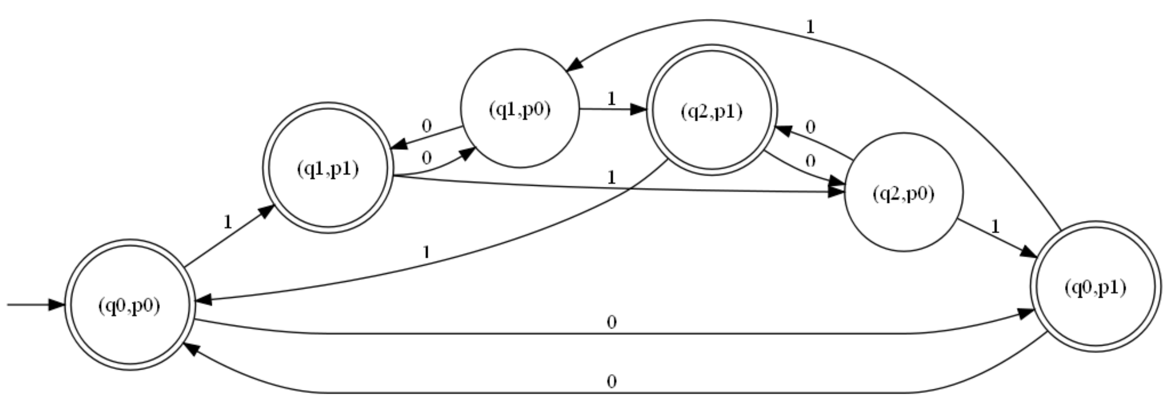
נבנה אס"ד M לשפה $L_1 \cap L_2$:

$$M_\cap = (Q \times P, \{0,1\}, \delta, (q_0, p_0), F_1 \times F_2)$$

$$Q \times P = \{(q_0,p_0),(q_0,p_1),(q_1,p_0),(q_1,p_1),(q_2,p_0),(q_2,p_1)\}$$
$$F_1 \times F_2 = \{(q_0,p_1)\}$$
$$\forall\ q \in Q, p \in P, \alpha \in \Sigma \quad \delta((q,p),\alpha) = (\delta_1(q,\alpha), \delta_2(p,\alpha))$$

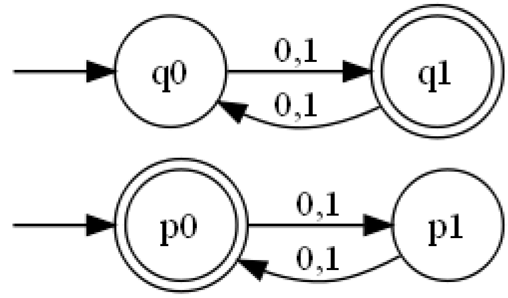


כדי לקבל אס"ד ל- $L_1 \cup L_2$ נגדיר קבוצת מצבים מקבלים כ- $F_1 \times P \cup Q \times F_2$

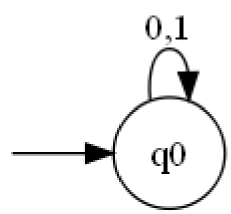


הערה: האוטומט המתקבל לא תמיד מינימאלי (מבחינת מספר המצבים)

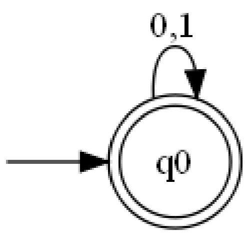
לדוגמה, $L_1 = \{w \in \{0,1\}^* : \text{אי-זוגי } |w|\}$, $L_2 = \{w \in \{0,1\}^* : \text{זוגי } |w|\}$



אם נפעל לפי השיטה שראינו נקבל אוטומט בעל 4 מצבים. אבל



$L_1 \cap L_2 = \varnothing$



$L_1 \cup L_2 = \Sigma^*$

האם שפות רגולריות סגורות תחת איחוד/חיתוך <u>אינסופי</u>? תשובה היא לא.

דוגמה נגדית:

$L_i = \{a^i b^i\}$ שפה רגולרית

נראה בהמשך כי $\bigcup_{n=0}^{\infty} L_n = \{a^n b^n : n \in \mathbb{N}\}$ שפה לא רגולרית.

$\overline{L_i}$ שפה רגולרית (כמשלים של שפה רגולרית)

$\bigcap_{n=0}^{\infty} \overline{L_n} = \overline{\{a^n b^n : n \in \mathbb{N}\}}$ שפה לא רגולרית (כי $\{a^n b^n : n \in \mathbb{N}\}$ לא רגולרית)

תרגיל 15:

L_1, L_2 שפות רגולריות. כלומר, קיימים אס"דים M_1, M_2 כך שמתקיים $L(M_1)=L_1, L(M_2)=L_2$

נגדיר שפה חדשה $L=\{a_1\circ b_1\circ a_2\circ b_2\circ\cdots\circ a_n\circ b_n : a_1\circ\cdots\circ a_n\in L_1,\ b_1\circ\cdots\circ b_n\in L_2,\ a_i, b_i\in\Sigma\}$

צריך להוכיח: L שפה רגולרית. כלומר, צריך לבנות אס"ד M כך שמתקיים $L(M)=L$.

רעיון: נבנה אס"ד בדומה לחיתוך/איחוד של השפות רק נריץ שני אס"דים לא "במקביל" אלא "לסירוגין".

כלומר, כאשר נקרא אותיות במקומות אי-זוגיים במילת הקלט נקדם רק את M_1 ובמקומות זוגיים נקדם רק את M_2.

בניה:

$$
\begin{aligned}
&\mathrm{M}_1=(Q,\Sigma,\delta_1,\mathrm{q}_0,F_1)\\
&\mathrm{M}_2=(P,\Sigma,\delta_2,\mathrm{p}_0,\mathrm{F}_2)\\
&\mathrm{M}=(Q\times P\times\{0,1\},\Sigma,\delta,(\mathrm{q}_0,\mathrm{p}_0,0),F_1\times\mathrm{F}_2\times\{0\})\\
&\forall q\in Q, p\in P,\alpha\in\Sigma:\ \begin{array}{l}\delta((q,p,0),\alpha)=(\delta_1(q,\alpha),p,1)\\ \delta((q,p,1),\alpha)=(q,\delta_2(p,\alpha),0)\end{array}
\end{aligned}
$$

הוכחה:

צריך להוכיח $L(M)=L$.

$L(M)\subseteq L$:

מילה $w=w_1w_2\cdots w_k,\ w_i\in\Sigma$ מתקבלת ע"י אס"ד M צריך להראות: k זוגי ו-
$w_1w_3\cdots w_{k-1}\in L_1,\ w_2w_4\cdots w_k\in L_2$

התחלנו את הריצה במצב התחלתי שמסומן ב-"0", המילה התקבלה לכן סיימנו את הריצה במצב מקבל שגם מסומן ב-"0". כל אות ב-w גורמת לשינוי של "0" ל-"1" ולהפך. לכן מספר אותיות ב-w הוא זוגי.

$$0\xrightarrow{w_1}1\xrightarrow{w_2}0\xrightarrow{w_3}1\xrightarrow{w_4}0$$

$$\begin{pmatrix} q_0 \\ p_0 \\ 0 \end{pmatrix} \xrightarrow{w_1} \begin{pmatrix} \delta_1(q_0,w_1) \\ p_0 \\ 1 \end{pmatrix} \xrightarrow{w_2} \begin{pmatrix} \delta_1(q_0,w_1) \\ \delta_2(p_0,w_2) \\ 0 \end{pmatrix} \to \cdots \to \begin{pmatrix} \delta_1(\delta_1(\cdots\delta_1(q_0,w_1),w_3)\cdots w_{k-1}) \\ \delta_2(\delta_2(\cdots\delta_2(p_0,w_2),w_4)\cdots w_k) \\ 0 \end{pmatrix} \in \begin{matrix} F_1 \\ \times \\ F_2 \\ \times \\ \{0\} \end{matrix}$$

$$\delta_1(\delta_1(\cdots\delta_1(q_0,w_1),w_3)\cdots w_{k-1}) \in F_1 \Rightarrow w_1w_3\cdots w_{k-1} \in L_1$$
$$\delta_2(\delta_2(\cdots\delta_2(p_0,w_2),w_4)\cdots w_k) \in F_2 \Rightarrow w_2w_4\cdots w_k \in L_2$$

$L(M) \supseteq L$:

$a_1a_2\cdots a_n \in L_1,\ b_1b_2\cdots b_n \in L_2$ צריך להראות: $a_1b_1a_2b_2\cdots a_nb_n \in L(M)$

$$\begin{pmatrix} q_0 \\ p_0 \\ 0 \end{pmatrix} \xrightarrow{a_1} \begin{pmatrix} \delta_1(q_0,a_1) \\ p_0 \\ 1 \end{pmatrix} \xrightarrow{b_1} \begin{pmatrix} \delta_1(q_0,a_1) \\ \delta_2(p_0,b_1) \\ 0 \end{pmatrix} \to \cdots \to \begin{pmatrix} \delta_1(\delta_1(\cdots\delta_1(q_0,a_1),a_2)\cdots a_n) \\ \delta_2(\delta_2(\cdots\delta_2(p_0,b_1),b_2)\cdots b_n) \\ 0 \end{pmatrix}$$

$\delta_1(\delta_1(\cdots\delta_1(q_0,a_1),a_2)\cdots a_n) \in F_1,\ \delta_2(\delta_2(\cdots\delta_2(p_0,b_1),b_2)\cdots b_n) \in F_2$ לפי הנתון. לכן

$$\begin{pmatrix} \delta_1(\delta_1(\cdots\delta_1(q_0,a_1),a_2)\cdots a_n) \\ \delta_2(\delta_2(\cdots\delta_2(p_0,b_1),b_2)\cdots b_n) \\ 0 \end{pmatrix} \in \begin{matrix} F_1 \\ \times \\ F_2 \\ \times \\ \{0\} \end{matrix}$$

כלומר, מילה $a_1b_1a_2b_2\cdots a_nb_n$ מתקבלת באס"ד M.

תרגיל 16 (מתוך מבחן סמסטר א' 2010):

$\Sigma = \{a, b, c, \ldots, z\}$

$L_{16} = \{w \in \Sigma^* :$ בדיוק פעמיים w מופיעה ב- σ כך ש- $\exists \sigma \in \Sigma\}$

תכננו אס"ד שמקבל את השפה. האם שפה זו היא סופית?

פתרון:

נייצג כל מצב באס"ד ע"י ווקטור באורך $|\Sigma| = 26$. כאשר ערך של רכיב i בווקטור שווה למספר מופעים של אות מספר i . ערכים שכל רכיב בווקטור יכול לקבל הם : "◊","2","1","0" (סה"כ 4^{26} מצבים).

מצב התחלתי מיוצג ע"י ווקטור $(0, 0, \ldots, 0)$.

פונקצית מעבר מוגדרת באופן הבא:

$$\delta(v, \sigma_i) = v'$$
$$v'[j] = \begin{cases} v[j] & j \neq i \\ v[j] + 1 & j = i \end{cases}$$
$$0 + 1 = 1, 1 + 1 = 2, 2 + 1 = \Diamond, \Diamond + 1 = \Diamond$$

מצבים מקבלים: כל מצבים שמיוצגים ע"י ווקטור שלפחות אחד הרכיבים שלו הוא "2".

שפה אינסופית: לדוגמה $i = 0,1,2,\ldots$ $aab^i \in L_{16}$

תרגיל 17:

אילו שינויים צרכים לבצע באס"ד לשפה L_{16} כדי לקבל אס"ד לשפה

$\Sigma = \{a,b,c,\ldots,z\}$

$L_{17} = \{w \in \Sigma^* :$ כל אות מופיעה ב- w בדיוק פעמיים $\}$

האם שפה זו היא סופית?

פתרון:

נייצג כל מצב באס"ד ע"י ווקטור באורך $|\Sigma| = 26$. כאשר ערך של רכיב i בווקטור שווה למספר מופעים של אות מספר i. ערכים שכל רכיב בווקטור יכול לקבל הם : "2","1","0" ומצב נוסף "בור" (סה"כ $3^{26}+1$ מצבים).

מצב התחלתי מיוצג ע"י ווקטור $(0,0,\ldots,0)$.

פונקצית מעבר מוגדרת באופן הבא:

$$\begin{aligned}
&\forall i\ \ \delta(\text{"}pit\text{"},\sigma_i) = \text{"}pit\text{"} \\
&v \neq \text{"}pit\text{"},\ v[i] \neq 2:\ \ \delta(v,\sigma_i) = v' \\
&v'[j] = \begin{cases} v[j] & j \neq i \\ v[j]+1 & j = i \end{cases} \\
&v \neq \text{"}pit\text{"},\ v[i] = 2:\ \ \delta(v,\sigma_i) = \text{"}pit\text{"}
\end{aligned}$$

מצב מקבל מיוצג ע"י ווקטור $(2,2,\ldots,2)$.

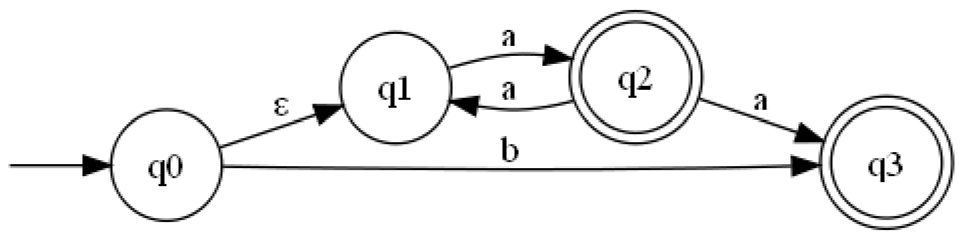
שפה סופית: $|L_{17}| = \dfrac{52!}{(2!)^{26}}$

אוטומט סופי לא דטרמיניסטי (אסל"ד)

אוטומט סופי לא דטרמיניסטי הוא הכללה של אוטומט סופי דטרמיניסטי. בניגוד לאוטומט דטרמיניסטי, בו דרשנו לכל מצב ולכל אות בא"ב בדיוק מעבר אחד, באוטומט לא דטרמיניסטי נאפשר לכל אות <u>מספר סופי כלשהו</u> (כולל אפס) של מעברים מכל מצב.

בנוסף נאפשר מעברים מיוחדים מסומנים ב- ε בין המצבים. המאפשרים לעבור ממצב למצב ללא קריאת אות מהקלט.

דוגמה:



בניגוד לאס"ד, שבו לכל מילה יש מסלול חישוב אחד, באסל"ד למילה יכולים להיות אפס או יותר (מספר סופי) מסלולי חישוב.

לדוגמה, למילה ab אין מסלול חישוב באסל"ד לעיל, למילה b יש מסלול חישוב מקבל אחד ולמילה aa יש שני מסלולי חישוב – אחד מקבל והשני לא.

מילה מתקבלת באסל"ד כאשר קיים לה לפחות מסלול מקבל (מסתיים במצב מקבל) אחד.

לכן, המילים aa, b מתקבלות, והמילה ab לא מתקבלת

באופן פורמאלי אסל"ד מוגדר ע"י חמישייה סדורה (Q,Σ,δ,q_0,F)

Q – קבוצה סופית של מצבים.

Σ – א"ב.

$\delta: Q\times(\Sigma\cup\{\varepsilon\})\rightarrow P(Q)$ – פונקצית מעבר. כאשר $P(Q)$ קבוצה של כל תתי-קבוצות של Q.

q_0 – מצב התחלתי.

$F\subseteq Q$ – קבוצת מצבים מקבלים.

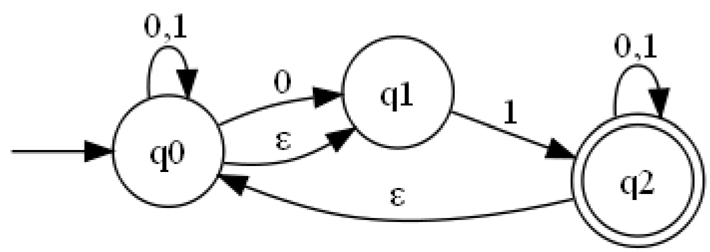
משפט: קיים אסל"ד לשפה אמ"ם קיים אס"ד לשפה.

בהינתן אסל"ד (Q,Σ,δ,q_0,F) ניתן לבנות אס"ד המקבל אותה שפה באופן הבא:

- כל תת-קבוצה של Q נייצג ע"י מצב באס"ד. כלומר, מספר מצבים באס"ד יהיה כמספר תתי-קבוצות של Q שווה ל- $2^{|Q|}$.
- מצב התחלתי באס"ד – מצב שמייצג קבוצת המצבים של אסל"ד שניתן להגיע אליהם מ- q_0 באמצעות מעברי ε (כולל q_0).
- מצבים מקבלים – כל מצב שמייצג תת-קבוצה של Q שמכילה לפחות מצב אחד מ- F.
- כאשר נמצאים במצב שמייצג קבוצה A ומקבלים כקלט אות $\alpha \in \Sigma$ עוברים למצב שמייצג קבוצת כל מצבים שניתן להגיע אליהם כאשר מקבלים קלט α ונמצאים באחד המצבים של A.

דוגמה:

נתון אסל"ד



שפה שלו $\{w \in \{0,1\}^* : 1$ מכילה $w\}$

נבנה אס"ד כפי שמתואר:

$Q=\{q_0,q_1,q_2\}$, כל תתי-קבוצות של Q: $\{q_0,q_1,q_2\},\{q_1,q_2\},\{q_0,q_2\},\{q_0,q_1\},\{q_2\},\{q_1\},\{q_0\},\varnothing$. לכן באס"ד יהיו 8 מצבים (אחד לכל תת-קבוצה).

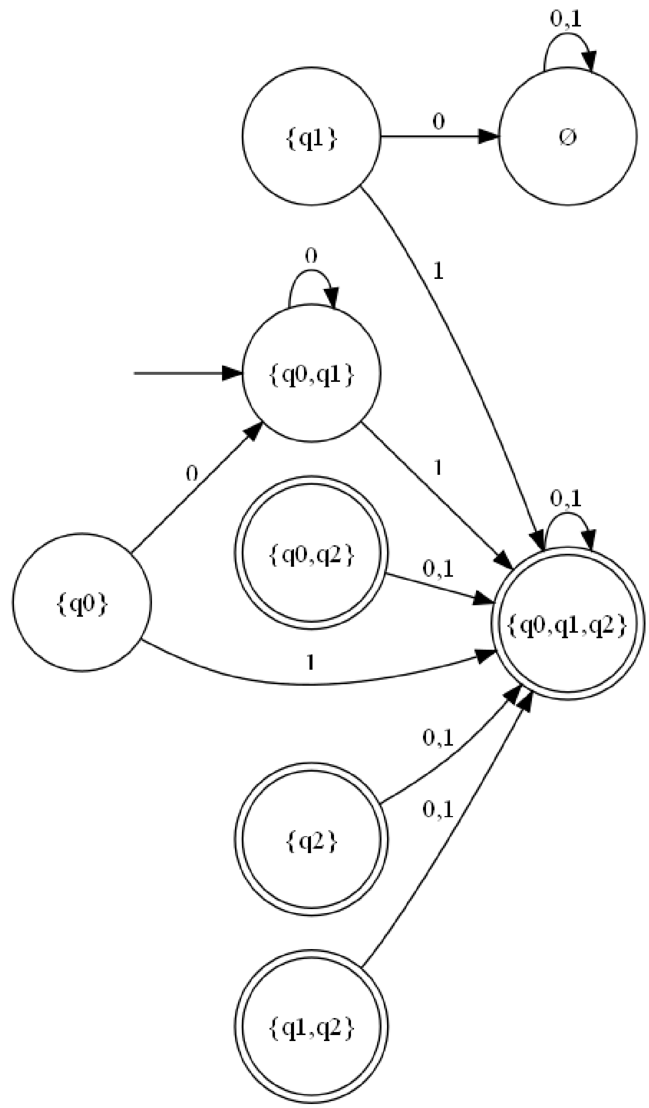
ממצב התחלתי q_0 באמצעות מעברי ε ניתן להגיע ל- q_0 (נשארים במקום) או למצב q_1. לכן, מצב התחלתי באס"ד הוא מצב המייצג $\{q_0,q_1\}$.

מצבים מקבלים הם כל המצבים שמייצגים קבוצות המכילות q_2: $\{q_0,q_1,q_2\},\{q_1,q_2\},\{q_0,q_2\},\{q_2\}$.

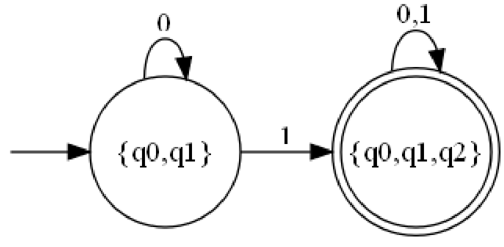
כאשר נמצאים במצב המייצג $\{q_1\}$ ומקבלים קלט 1 – ניתן להגיע למצבים q_0,q_1,q_2 (q_0,q_1 באמצעות מעברי ε מ- q_2). נקבל $\delta("\{q_1\}",1\}="\{q_0,q_1,q_2\}"$. כאשר נמצאים במצב המייצג $\{q_1\}$ ומקבלים קלט 0 – לא ניתן להגיע לאף מצב. נקבל $\delta("\{q_1\}",0\}="\varnothing"$.

כאשר נמצאים במצב המייצג $\{q_0,q_2\}$ ומקבלים קלט 0 – מ- q_0 ניתן להגיע ל- q_0,q_1 בנוסף מ- q_2 ניתן להגיע ל- q_2 (וגם ל- q_0,q_1). נקבל $\delta("\{q_0,q_2\}",0\}="\{q_0,q_1,q_2\}"$.

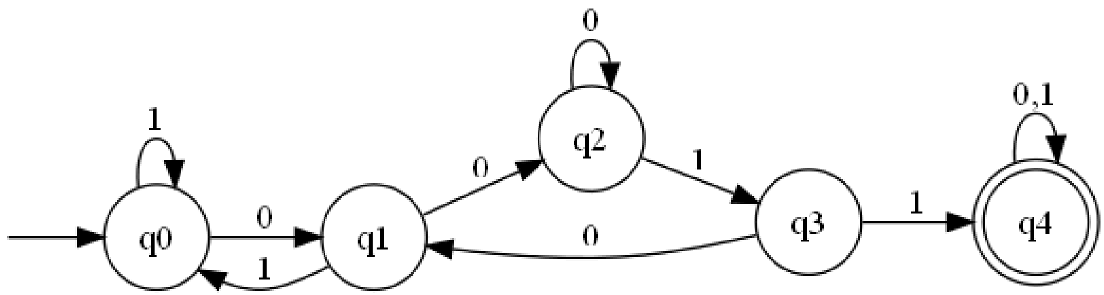
נפעל באותו אופן לכל מצב ולכל אות... נקבל:



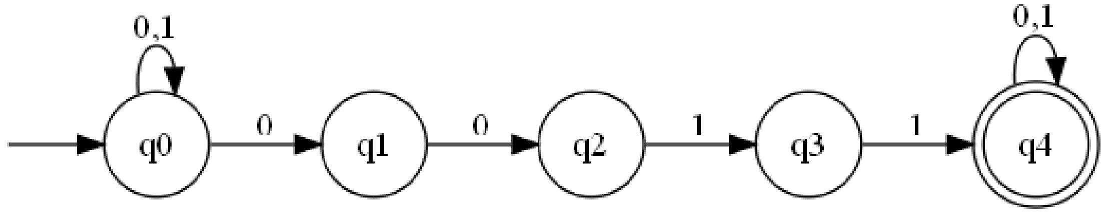
האוטומט המתקבל לא תמיד מינימאלי. לדוגמה באס"ד שקיבלנו לא ניתן להגיע למצבים $\varnothing,\{q_0\},\{q_1\},\{q_2\},\{q_0,q_2\},\{q_1,q_2\}$ ממצב ההתחלתי. לכן, ניתן להסיר אותם:



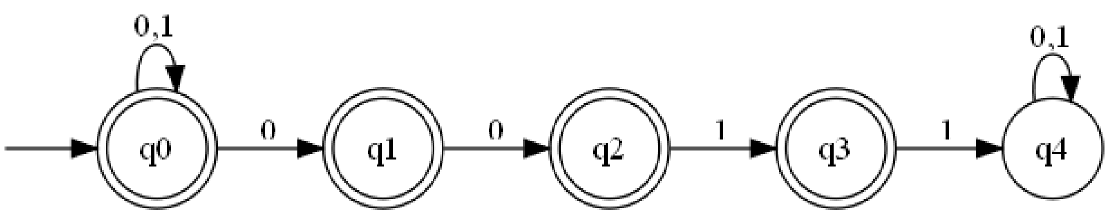
ראינו אס"ד לשפה $\{w \in \{0,1\}^* : \text{0011 מכילה } w\}$



אסל"ד המקבל את אותה שפה:



באס"ד ראינו שניתן להפוך מצבים מקבלים ללא מקבלים ולהפך ולקבל אס"ד למשלים של השפה. האם זה נכון גם לאסל"ד? תשובה היא לא.

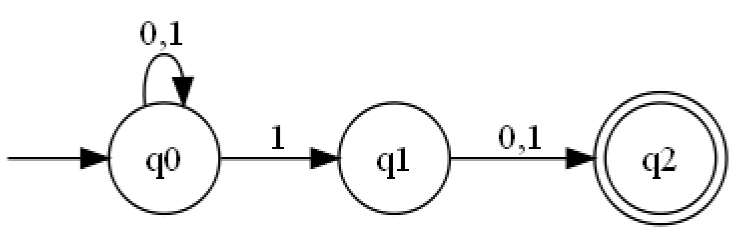


שפה שאסל"ד זה מקבל היא $\{0,1\}^*$.

תרגיל 1: נתונה שפה $L_1 = \{w \in \{0,1\}^* : \text{1 הוא w-התו לפני אחרון ב}, |w| > 1\}$

תכננו אסל"ד שמקבל את השפה.

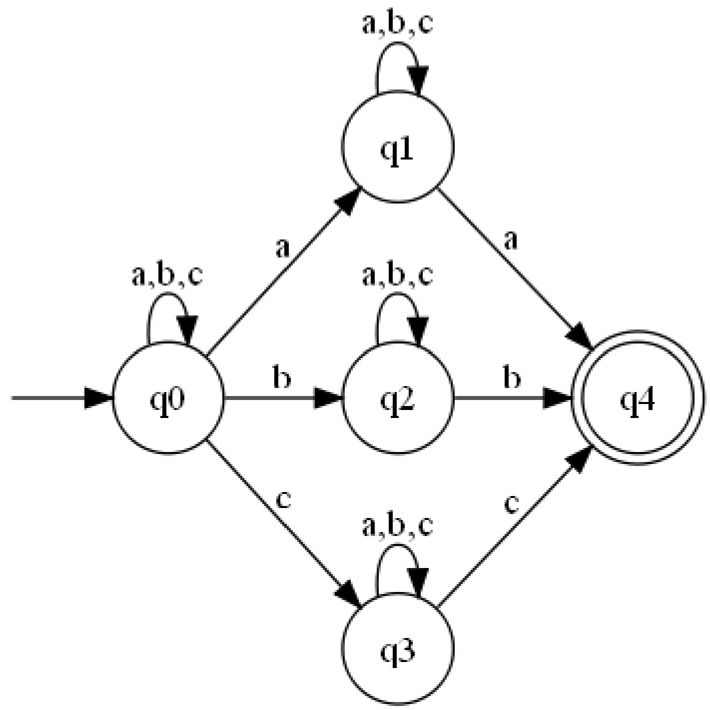
פתרון:



תרגיל 2: נתונה שפה $L_2 = \{w \in \{a,b,c\}^* :$ התו האחרון ב-w מופיע לפחות פעמיים במילה, $|w|>1\}$

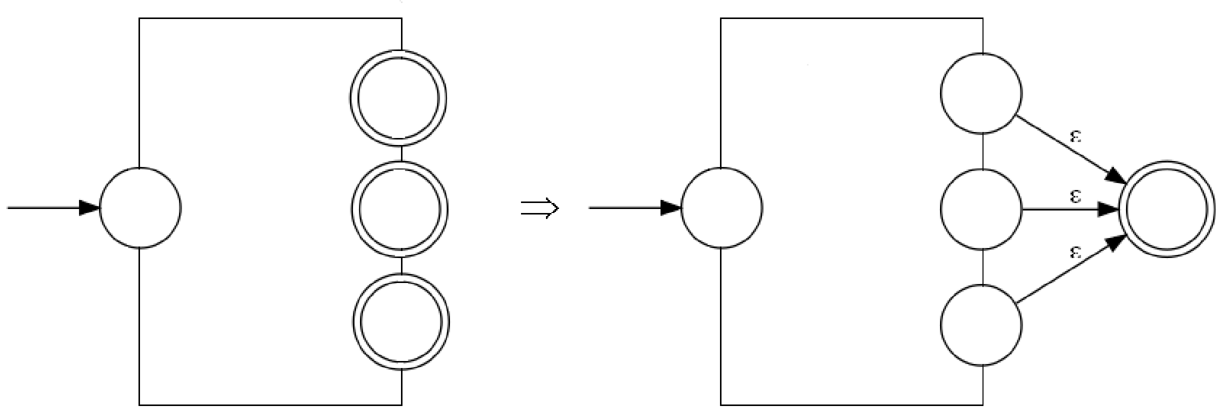
תכננו אסל"ד שמקבל את השפה.

פתרון:



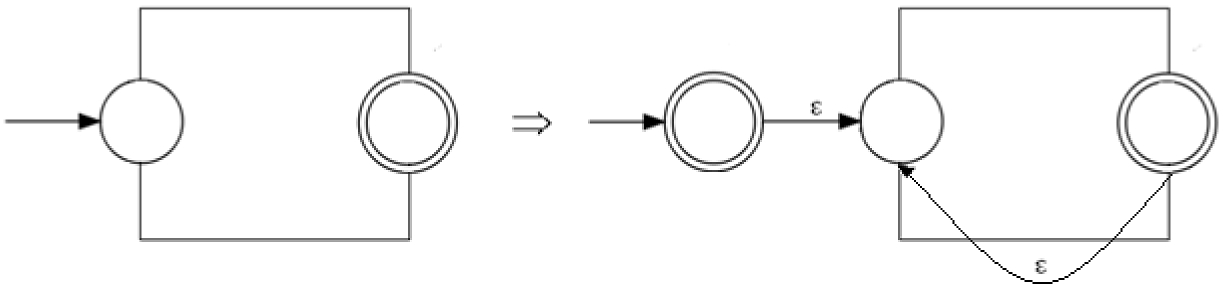
למה 1: אם קיים אסל"ד לשפה, אזי קיים אסל"ד בעל מצב מקבל אחד המקבל את השפה.

רעיון: בהינתן אסל"ד לשפה נבנה אסל"ד בעל מצב מקבל אחד באופן הבא: נוסיף מצב מקבל חדש. כל מצב שהיה מקבל נהפוך ללא מקבל ונחבר אותו עם מעבר ε למצב מקבל חדש.



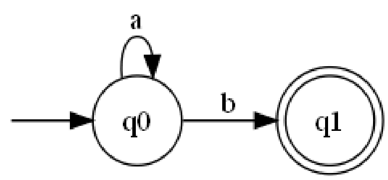
טענה 1: אם שפה L מתקבלת ע"י אסל"ד, אזי גם L^* מתקבלת ע"י אסל"ד.

רעיון: בהינתן אסל"ד לשפה L בעל מצב מקבל אחד (קיים לפי למה 1) נבנה אסל"ד ל- L^* באופן הבא: נחבר את המצב המקבל עם מעבר ε למצב ההתחלתי. נוסיף מצב התחלתי חדש ונגדיר אותו כמקבל (מצב התחלתי צריך להיות מקבל כי $\varepsilon \in L^*$). נחבר אותו עם מעבר ε למצב ההתחלתי הקודם.

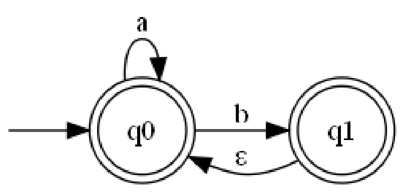


למה צרכים להוסיף מצב התחלתי חדש?

נתון אסל"ד



לשפה $L = \{a^i b : i \geq 0\}$ אם לא היינו מוסיפים מצב התחלתי חדש היינו מקבלים אסל"ד הבא



שמקבל כל מילה ובפרט מילה a שלא נמצאת ב- L^*.

תרגיל 3 (מבחן מועד א' סמסטר א' 2007): בהינתן שפה L נגדיר שפה $e(L)=\{w\cdot parity(w): w\in L\}$

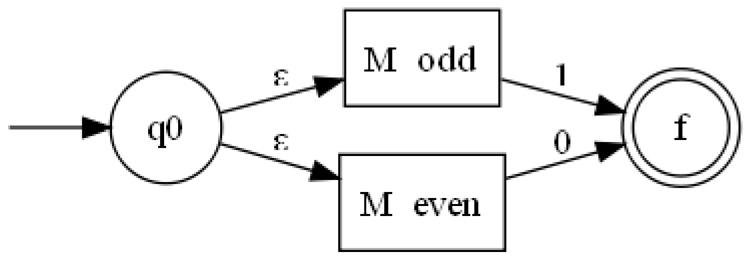
כאשר $parity(w)$ שווה 0 אם $|w|$ זוגי ו-1 אחרת.

הוכיחו: אם L שפה רגולרית, אזי $e(L)$ רגולרית.

פתרון:

מתקיים: $L\cap\{w\in\Sigma^*: |w| \text{ זוגי}\}$ ו- $L\cap\{w\in\Sigma^*: |w| \text{ אי-זוגי}\}$ שפות רגולריות (כחיתוך של שתי שפות רגולריות). לכן, קיימים אסל"ד M_{odd}, M_{even} בעלי מצב מקבל אחד המקבלים שפות אלו.

אסל"ד לשפה $e(L)$:



נכונות הבניה:

אם מילה v בשפה $e(L)$ ומסתיימת ב-0 אזי קיימת מילה $w'\in L$ שאורכה זוגי ומתקיים $v=w'\cdot 0$.

למילה w' קיים באסל"ד M_{even} מסלול חישוב $q_0^{even}\to\cdots\to f^{even}$ כאשר f^{even} מצב מקבל.

לכן, $q_0 \xrightarrow{\varepsilon} q_0^{even}\to\cdots\to f^{even}\xrightarrow{0} f$ מסלול חישוב מקבל של v באסל"ד שבנינו.

אם מילה מסתיימת ב-1 $q_0 \xrightarrow{\varepsilon} q_0^{odd}\to\cdots\to f^{odd}\xrightarrow{1} f$ מסלול חישוב מקבל של v.

אם מילה v מתקבלת באסל"ד יש לה מסלול חישוב מקבל מהצורה $q_0 \xrightarrow{\varepsilon} q_0^{even}\to\cdots\to f^{even}\xrightarrow{0} f$ או $q_0 \xrightarrow{\varepsilon} q_0^{odd}\to\cdots\to f^{odd}\xrightarrow{1} f$.

במקרה ראשון קיימת מילה w' שמתקבלת ב- M_{even} (ב- L ואורכה זוגי) כך ש- $v=w'\cdot 0$.

במקרה שני קיימת מילה w' שמתקבלת ב- M_{odd} (ב- L ואורכה אי-זוגי) כך ש- $v=w'\cdot 1$.

בשני המקרים מקבלים $v\in e(L)$.

תרגיל 4 (מבחן מועד ב' סמסטר א' 2004): בהינתן שפה $L \subseteq \Sigma^*$ נגדיר שפה L' באופן הבא:

$L' = \{u \in \Sigma^*$: קיימת $w \in L$ כך ש- u מתקבלת מ-w ע"י מחיקה של לכל היותר 100 אותיות$\}$

נניח כי L שפה רגולרית. האם L' בהכרח רגולרית?

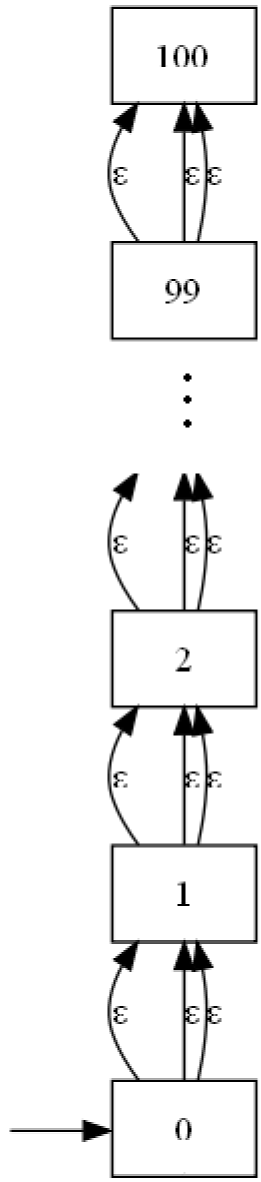
פתרון:

רעיון: אם בהינתן אס"ד לשפה L נבנה אסל"ד חדש ע"י הוספת מעבר ε בין כל שתי מצבים שיש ביניהם מעבר עם אות. כלומר, אם מוגדר באס"ד $\delta(q,\alpha) = p$ באסל"ד יהיה מוגדר $\delta'(q,\alpha) = \{p\}$ וגם $\delta'(q,\varepsilon) = \{p\}$. נקבל אסל"ד שמקבל כל מילה שמתקבלת ע"י מחיקה (שימוש במעבר ε במקום אות) של <u>מספר אותיות כלשהו</u> מהמילה ב- L. כלומר, אין שליטה על מספר האותיות שנמחקו. כדי להגביל מספר אותיות שנמחקו ל- 100 לכל היותר, "נספור" כמה פעמים השתמשנו במעברי ε:

נבנה אסל"ד שמורכב מ-101 העתקים של אס"ד לשפה L. ממוספרים מ-0 עד 100. אם בשלב מסוים בריצה על מילת הקלט הגענו להעתק i המשמעות של זה – השתמשנו במעברי ε i פעמים.

מעברי ε יהיו מוגדרים באופן הבא: אם מוגדר באס"ד לשפה L $\delta(q,\alpha) = p$ נגדיר $j = 0,1,\ldots 99$ $\delta'(q^j,\varepsilon) = \{p^{j+1}\}$ (כלומר, מעברי ε רק בין שני העתקים שכנים).

מצב התחלתי – מצב התחלתי של העתק מספר 0. מצבים מקבלים – כל מצבים מקבלים בכל העתקים. (לדוגמה, אם הינו דורשים מחיקה של <u>בדיוק</u> 100 אותיות, צריך היה להגדיר מצבים מקבלים רק בהעתק מספר 100 ובכל שאר העתקים להפוך מצבים מקבלים ללא מקבלים.)



באופן פורמאלי: בהינתן אס"ד לשפה L (Q,Σ,δ,q_0,F) נגדיר אסל"ד:

$(Q\times\{0,\ldots,100\},\Sigma,\delta',(q_0,0),F\times\{0,\ldots,100\})$

אם מוגדר באס"ד $\delta(q,\alpha)=p$ באסל"ד יהיה מוגדר $j=0,\ldots,100$ $\delta'((q,j),\alpha)=\{(p,j)\}$ ובנוסף,

$\delta'((q,j),\varepsilon)=\{(p,j+1)\}$ $j=0,1,\ldots 99$.

ואם הינו דורשים מחיקה של <u>לפחות</u> 100 אותיות?

נגדיר מצבים מקבלים רק בהעתק מספר 100 ובנוסף בין כל שני מצבים בהעתק זה שיש ביניהם מעבר עם אות נגדיר בנוסף מעבר ε (כפי שהיה ברעיון הראשוני).

ביטויים רגולריים

נתון א"ב Σ. נגדיר קבוצת ביטויים רגולריים באופן הבא:

- $\varnothing$
- ε
- a כאשר $a \in \Sigma$

בנוסף, אם R_1, R_2 ביטויים רגולריים אזי גם:

- $(R_1 \cup R_2)$ (לפעמים מחליפים סימן $\cup$ בסימן $+$)
- $(R_1 \circ R_2)$
- (R_1^*)

לדוגמה, אם $\Sigma = \{0,1\}$ אזי ביטויים רגולריים שנקבל:

$$\varnothing, \varepsilon, 1, 0, (\varepsilon \cup \varepsilon), (\varepsilon \cup 1), (\varnothing^*), (\varepsilon \circ 1), (0 \cup 1), ((\varepsilon \cup 1)^*), (((0^*)^*)^*), (((((0^*)^*)^*) \cup ((\varepsilon \cup 1)^*)) \circ \varnothing), \ldots$$

לכל ביטוי רגולרי R אפשר להתאים שפה. נסמן אותה ב- $L(R)$.

נגדיר:

- $L(\varnothing) = \{\} = \varnothing$
- $L(\varepsilon) = \{\varepsilon\}$
- $L(a) = \{a\}$ כאשר $a \in \Sigma$

אם R_1, R_2 ביטויים רגולריים אזי:

- $L((R_1 \cup R_2)) = L(R_1) \cup L(R_2)$
- $L((R_1 \circ R_2)) = L(R_1) \circ L(R_2)$
- $L((R_1^*)) = L(R_1)^*$

דוגמאות:

$$L(((0\cup 1)^*)) = L((0\cup 1))^* = (L(0)\cup L(1))^* = (\{0\}\cup\{1\})^* = \{0,1\}^*$$

$$L(((0\circ 1)^*)) = L((0\circ 1))^* = (L(0)\circ L(1))^* = (\{0\}\circ\{1\})^* = \{01\}^*$$

$$L(((1^*)\circ\varnothing)) = L((1^*))\circ L(\varnothing) = L(1)^*\circ\varnothing = \{1\}^*\circ\varnothing = \varnothing$$

$$L((\varnothing^*)) = L(\varnothing)^* = \varnothing^* = \varnothing^0\cup\varnothing^1\cup\varnothing^2\cup\ldots = \{\varepsilon\}\cup\varnothing\cup\ldots = \{\varepsilon\}$$

$$L(((0\cup\varepsilon)\circ(1\cup\varepsilon))) = L((0\cup\varepsilon))\circ L((1\cup\varepsilon)) = \{0,\varepsilon\}\circ\{1,\varepsilon\} = \{01,0,1,\varepsilon\}$$

בדרך כלל כדי להקל על הקריאה של ביטויים רגולריים מגדירים סדר קדימות

$*$ קדימות גבוהה

$\circ$

$\cup$

ולא כותבים סוגריים כשהסרתם לא פוגעת ביכולת לשחזר ביטוי מקורי (בדומה לביטויים אריתמטיים).
בנוסף, לפעמים לא כותבים סימן $\circ$.

לדוגמה, ביטוי רגולרי $((0\circ(1^*))\cup 1)$ ניתן לכתוב בצורה מקוצרת כ- $01^*\cup 1$.

$(((0\circ 1)^*)\cup 1)$ אפשר לכתוב כ- $(01)^*\cup 1$.

משפט: L שפה רגולרית (קיים אסל"ד/אס"ד לשפה) $\Leftrightarrow$ קיים ביטוי רגולרי R כך שמתקיים $L = L(R)$

תרגיל 1: כתבו ביטוי רגולרי לשפה $L_1 = \{w\in\{0,1\}^* :$ מכילה בדיוק שלושה 1-ים $w\}$.

פתרון: ביטוי רגולרי לשפה: $0^*10^*10^*10^*$

הוכחת נכונות: מילה ב- $L(0^*10^*10^*10^*)$ $\Leftrightarrow$ מילה מהצורה $0^i10^j10^k10^m$ כאשר $i,j,k,m\geq 0$ $\Leftrightarrow$
מילה מכילה בדיוק שלושה 1-ים $\Leftrightarrow$ מילה היא ב- L_1.

מסקנה: $L(0^*10^*10^*10^*) = L_1$.

תרגיל 2: כתבו ביטוי רגולרי לשפה $\overline{L_1}$.

פתרון: מילה ב- $\overline{L_1}$ אם מספר 1-ים בה שונה מ-3. כלומר, מספר 1-ים הוא $4,2,1,0$ או יותר.

0^* - מספר 1-ים במילה הוא 0.

0^*10^* - מספר 1-ים במילה הוא 1.

$0^*10^*10^*$ - מספר 1-ים במילה הוא 2.

$(0\cup 1)^*1(0\cup 1)^*1(0\cup 1)^*1(0\cup 1)^*1(0\cup 1)^*$ - מספר 1-ים במילה הוא 4 או יותר.

לכן, ביטוי רגולרי לשפה $\overline{L_1}$ הוא $0^*\cup 0^*10^*\cup 0^*10^*10^*\cup(0\cup 1)^*1(0\cup 1)^*1(0\cup 1)^*1(0\cup 1)^*1(0\cup 1)^*$.

תרגיל 3: נתון ביטוי רגולרי $R=(0\circ(1\cup 0)^*\circ 0)\cup(1\circ(1\cup 0)^*\circ 1)$. מהי $L(R)$?

פתרון: $L(R)=\{w\in\{0,1\}^*:\ |w|>1$ מתחילה ומסתיימת באותו תו ומתקיים $w\}$

הוכחה: $L((0\circ(1\cup 0)^*\circ 0)\cup(1\circ(1\cup 0)^*\circ 1))=\{0\circ v\circ 0: v\in\{0,1\}^*\}\cup\{1\circ v\circ 1: v\in\{0,1\}^*\}$ לכן, ברור שכל מילה ב- $L(R)$ היא מתחילה ב-1 ומסתיימת ב-1 או מתחילה ב-0 ומסתיימת ב-0 וגודלה לפחות 2.

מצד שני אם מילה מתחילה ומסתיימת באותו תו ומתקיים $|w|>1$ היא מהצורה $0\circ v\circ 0$ או $1\circ v\circ 1$כאשר $v\in\{0,1\}^*$ ולכן היא ב- $L(R)$.

האם $L((0\circ(1\cup 0)^*\circ 0)\cup(1\circ(1\cup 0)^*\circ 1))=L((0\cup 1)\circ(1\cup 0)^*\circ(0\cup 1))$?

תשובה היא לא. לדוגמה, $01\in L((0\cup 1)\circ(1\cup 0)^*\circ(0\cup 1))$.

תרגיל 4: כתבו ביטוי רגולרי לשפה $\{w\in\{0,1\}^*$: 101 לא מכילה $w\}$.

כל מילה בשפה היא מהצורה:

$$(\underbrace{0\cdots 0}_{\ge 0})\underbrace{(\underbrace{1\cdots 1}_{\ge 1}\underbrace{0\cdots 0}_{\ge 2})(\underbrace{1\cdots 1}_{\ge 1}\underbrace{0\cdots 0}_{\ge 2})\cdots(\underbrace{1\cdots 1}_{\ge 1}\underbrace{0\cdots 0}_{\ge 2})}_{\ge 0}(\underbrace{1\cdots 1}_{\ge 0})(\underbrace{0\cdots 0}_{\ge 0})$$

כלומר, בין כל שתי סדרות של 1-ים יש לפחות 2 0-ים.

לכן, הביטוי רגולרי לשפה הוא $0^*(11^*000^*)^*1^*0^*$.

סיכום תכונות סגור של שפות רגולריות:

אם L_1, L_2 שפות רגולריות, אזי גם

- $L_1 \cup L_2$
- $L_1 \cap L_2$
- $L_1 \circ L_2$
- $\overline{L_1}$
- L_1^*
- L_1^R

שפות רגולריות.

יחס שקילות

יחס בינארי R על קבוצה S - תת קבוצה של $S \times S$.

אם $(a,b) \in R$ מסמנים aRb.

דוגמה: $R = \{(1,2),(1,3),\ldots,(2,3),(2,4),\ldots,(3,4),(3,5),\ldots\}$ יחס בינארי על $\mathbb{N}$ שמסומן בד"כ ב-"$<$".

תכונות של יחס $R \subseteq S \times S$:

1. רפלקסיביות - $\forall a \in S,\ aRa$
2. סימטריות - $\forall a,b \in S,\ aRb \Leftrightarrow bRa$
3. טרנזיטיביות - $\forall a,b,c \in S,\ aRb \wedge bRc \Rightarrow aRc$

יחס בינארי שיש לו תכונות 1,2,3 נקרא **יחס שקילות**.

יחס שקילות מחלק את הקבוצה עליה הוא מוגדר לתת-קבוצות זרות – **מחלקות שקילות**:

$a,b \in S$ באותה מחלקת שקילות אמ"ם aRb.

מחלקת שקילות של איבר $a \in S$ מסומנת ב- $[a]_R$ ומוגדרת באופן הבא $[a]_R = \{b \in S : aRb\}$.

מתקיים $[a]_R = [b]_R$ אמ"ם aRb.

אוסף של כל מחלקות שקילות שמגדיר יחס שקילות $R \subseteq S \times S$ נקרא **קבוצת המנה** S/R.

$$S/R = \{[a]_R : a \in S\}$$

דוגמה: יחס בינארי על $\mathbb{Z}$ $\equiv_{\text{mod}5}$ (שוויון מודולו 5) הוא יחס שקילות.

תזכורת: $a \equiv b \pmod{n}$ אם מתקיים $a - b = k \cdot n$ כאשר k מספר שלם.

- ✓ רפלקסיביות: $\forall a \in \mathbb{Z},\ a - a = 0 \cdot 5 \Rightarrow a \equiv_{\text{mod}5} a$
- ✓ סימטריות: $\forall a,b \in \mathbb{Z},\ a \equiv_{\text{mod}5} b \Leftrightarrow a - b = k \cdot 5 \Leftrightarrow b - a = (-k) \cdot 5 \Leftrightarrow b \equiv_{\text{mod}5} a$
- ✓ טרנזיטיביות:

$$\forall a,b,c \in \mathbb{Z},\ a \equiv_{\text{mod}5} b \wedge b \equiv_{\text{mod}5} c \Rightarrow a - b = k \cdot 5\ \wedge\ b - c = m \cdot 5 \Rightarrow$$
$$a - c = (a-b) + (b-c) = k \cdot 5 + m \cdot 5 = (k+m) \cdot 5 \Rightarrow a \equiv_{\text{mod}5} c$$

מחלקות שקילות:

$$[0]=\{0,5,-5,10,-10,\ldots\}$$
$$[1]=\{1,-1,6,-6,\ldots\}$$
$$[2]=\{2,-2,7,-7,\ldots\}$$
$$[3]=\{3,-3,8,-8,\ldots\}$$
$$[4]=\{4,-4,9,-9,\ldots\}$$

קבוצת המנה:

$$\mathbb{Z}/\equiv_{\text{mod}\,5}=\{[0],[1],[2],[3],[4]\}$$

בהינתן שפה L נגדיר יחס $\equiv_L$ על Σ^* באופן הבא:

$x,y\in\Sigma^*$ $x\equiv_L y$ אם מתקיים $\forall w\in\Sigma^*:\ x\circ w\in L\Leftrightarrow y\circ w\in L$

טענה: $\equiv_L$ יחס שקילות.

הוכחה:

- ✓ רפלקסיביות: $\forall x\in\Sigma^*,\ \forall w\in\Sigma^*\ x\circ w\in L\Leftrightarrow x\circ w\in L$
- ✓ סימטריות: $\forall x,y\in\Sigma^*,\ x\equiv_L y\Rightarrow\forall w\in\Sigma^*:\ x\circ w\in L\Leftrightarrow y\circ w\in L\Rightarrow y\equiv_L x$
- ✓ טרנזיטיביות:

$$\forall x,y,z\in\Sigma^*,\ x\equiv_L y\wedge y\equiv_L z\Rightarrow$$
$$(*)\ \ \forall w\in\Sigma^*:\ (x\circ w\in L\Leftrightarrow y\circ w\in L)\wedge$$
$$(**)\ \forall w'\in\Sigma^*:(y\circ w'\in L\Leftrightarrow z\circ w'\in L)$$

לכל v מתקיים:

אם $z\circ v\in L\underset{(**)}{\Leftarrow}y\circ v\in L\underset{(*)}{\Leftarrow}x\circ v\in L$

אם $z\circ v\notin L\underset{(**)}{\Leftarrow}y\circ v\notin L\underset{(*)}{\Leftarrow}x\circ v\notin L$

לכן, $\forall v\in\Sigma^*:\ x\circ v\in L\Leftrightarrow z\circ v\in L$

כלומר, $x\equiv_L z$.

בהינתן אס"ד A נגדיר יחס $\equiv_A$ על Σ^* באופן הבא:

ריצה של x ו- y באס"ד A מסתיימת באותו מצב $\Leftrightarrow$ $x\equiv_A y$ $x,y\in\Sigma^*$

יחס $\equiv_A$ הוא רפלקסיבי, סימטרי וטרנזיטיבי ולכן הוא יחס שקילות.

כאשר A הוא אס"ד לשפה L לכל $a,b\in\Sigma^*$ מתקיים: $b\equiv_A a \Leftarrow b\equiv_L a$.

$b\equiv_A a \Leftarrow \hat{\delta}(q_0,a)=\hat{\delta}(q_0,b)$ לכל w מתקיים:

$$a\circ w\in L \Leftrightarrow \hat{\delta}(q_0,a\circ w)\in F_A \Leftrightarrow \hat{\delta}(\hat{\delta}(q_0,a),w)\in F_A \Leftrightarrow \hat{\delta}(\hat{\delta}(q_0,b),w)\in F_A \Leftrightarrow \hat{\delta}(q_0,b\circ w)\in F_A \Leftrightarrow b\circ w\in L$$

כלומר, $b\equiv_L a$.

הגדרה:

$R_1,R_2\subseteq S\times S$ יחסי שקילות. אם מתקיים $aR_2b \Leftarrow aR_1b$ לכל $a,b\in S$

אזי יחס R_2 נקרא **עידון** של R_1.

מתקיים: מספר האיברים ב- S/R_2 גדול שווה למספר האיברים ב- S/R_1.

יחס $\equiv_A$ הוא עידון של $\equiv_L$ כאשר A הוא אס"ד לשפה L:

$b\equiv_A a \Leftarrow \hat{\delta}(q_0,a)=\hat{\delta}(q_0,b)$ לכל w מתקיים:

$$a\circ w\in L \Leftrightarrow \hat{\delta}(q_0,a\circ w)\in F_A \Leftrightarrow \hat{\delta}(\hat{\delta}(q_0,a),w)\in F_A \Leftrightarrow \hat{\delta}(\hat{\delta}(q_0,b),w)\in F_A \Leftrightarrow \hat{\delta}(q_0,b\circ w)\in F_A \Leftrightarrow b\circ w\in L$$

כלומר, $b\equiv_L a$.

שפות לא רגולריות

נשתמש בשני כלים עיקריים כדי להוכיח שהשפה הנתונה לא רגולרית: למת הניפוח לשפות רגולריות ומשפט Myhill–Nerode.

בנוסף, ניתן להיעזר בתכונות סגור של שפות רגולריות כדי להוכיח ששפה לא רגולרית.

לדוגמה: ידוע כי שפה $\{a^n b^n : n = 0,1,2,\ldots\}$ לא רגולרית. נוכיח בעזרת שימוש בתכונות סגור של שפות רגולריות כי גם $L = \{w \in \{0,1\}^* : \#0 \neq \#1\}$ שפה לא רגולרית.

נניח בשלילה L רגולרית. אזי שפה $\overline{L} \cap L(a^* b^*)$ רגולרית כחיתוך של שתי שפות רגולריות (שפה $\overline{L}$ רגולרית כמשלים של שפה רגולרית). בסתירה לכך ש- $\{a^n b^n : n = 0,1,2,\ldots\} = \overline{L} \cap L(a^* b^*)$ שפה לא רגולרית.

מסקנה: $L = \{w \in \{0,1\}^* : \#0 \neq \#1\}$ שפה לא רגולרית.

למת הניפוח לשפות רגולריות: לכל שפה רגולרית L קיים מספר טבעי $n_0 \geq 1$ ומתקיים:

לכל מילה $w \in L$ כך ש- $|w| \geq n_0$ קיימת חלוקה $w = xyz$ שמקיימת:

1. $|y| \geq 1$
2. $xy^k z \in L$ לכל $k \geq 0$
3. $|xy| \leq n_0$

שימוש עיקרי של למת הניפוח לשפות רגולריות הוא להוכיח ששפה לא רגולרית.

הערה: קיימות שפות לא רגולריות שמקיימות תנאי למת הניפוח לשפות רגולריות.

חשוב: לא ניתן להוכיח כי שפה רגולרית ע"י כך שנראה שהיא מקיימת תנאי הלמה.

כדי להראות ששפה לא רגולרית בעזרת למת הניפוח בד"כ מבצעים שלבים הבאים:

- נניח בשלילה כי השפה היא רגולרית. לכן למת הניפוח מתקיימת...
- כפונקציה של n_0 (שקיים לפי הלמה) בוחרים מילה ארוכה מספיק **בשפה**.
- נראה שבכל חלוקה אפשרית של המילה שבחרנו תנאים $1,2,3$ לא יכולים להתקיים בוזמנית.
- קיום של מילה כזאת בשפה סותר נכונות של למת הניפוח (שהוכחה כנכונה). מסקנה: הנחה לא נכונה. כלומר, שפה לא רגולרית.

שימוש במשפט Myhill–Nerode: כדי להראות ששפה לא רגולרית מספיק למצוא סדרה <u>אינסופית</u> $\{w_i\}_{i=1}^{\infty}$ של מילים ב- Σ^* (לא בהכרח בשפה) כך שמתקיים לכל $i \neq j$ $w_i \not\equiv_L w_j$ (כלומר, קיימת $z \in \Sigma^*$ כך שמתקיים $w_i z \in L \wedge w_j z \notin L$ **או** $w_i z \notin L \wedge w_j z \in L$).

תרגיל 1: נתונה שפה $L_1 = \{a^n b^n : n \geq m\}$

הוכיחו כי שפה לא רגולרית.

פתרון:

בעזרת למת הניפוח לשפות רגולריות:

נניח בשלילה כי שפה זו רגולרית $\Leftarrow$ לפי למת הניפוח, קיים מספר טבעי $n_0 \geq 1$ ומתקיים: לכל מילה $w \in L_1$ כך ש- $|w| \geq n_0$ קיימת חלוקה $w = xyz$ ומתקיימים תנאים 1,2,3 של הלמה.

נבחר מילה $w = a^{n_0} b^{n_0} \in L_1$ מתקיים $|w| = 2n_0 \geq n_0$ לכן קיימת חלוקה $w = xyz$ שמקיימת תנאים 1,2,3 של הלמה.

לפי תנאי 3 $|xy| \leq n_0$, לכן בכל חלוקה אפשרית y סדרה של a-ים. לפי תנאי 1 $|y| \geq 1$, לכן y סדרה לא ריקה של a-ים.

לפי תנאי 2 $xy^k z \in L$ לכל $k \geq 0$. אם נבחר $k = 0$ נקבל $xy^0 z = xz = a^{n_0 - |y|} b^{n_0} \in L_1$. אבל $a^{n_0 - |y|} b^{n_0} \notin L_1$ כי $n_0 - |y| < n_0$ (מתקיים $|y| \geq 1$). כלומר, כל חלוקה של מילה $w = a^{n_0} b^{n_0}$ לא יכולה לקיים בוזמנית תנאים 1,2,3 של הלמה.

מסקנה: שפה לא רגולרית.

בעזרת משפט Myhill–Nerode:

קיימת סדרת מילים אינסופית $w_i = a^i \quad i = 1,2,3,\ldots$

נוכיח כי מתקיים: $w_i \not\equiv_L w_j$ לכל $i \neq j$.

נגדיר $z = b^{\max(i,j)}$ מתקיים:

אם $i > j$, $a^i z = a^i b^{\max(i,j)} = a^i b^i \in L,\ a^j z = a^j b^{\max(i,j)} = a^j b^i \notin L$

אם $i < j$, $a^i z = a^i b^{\max(i,j)} = a^i b^j \notin L,\ a^j z = a^j b^{\max(i,j)} = a^j b^j \in L$

תרגיל 2 (מבחן מועד א' סמסטר א' 2010): נתונה שפה

$L_2 = \{w \in \{a,b,\ldots,z\}^* :$ קיימת אות $\sigma \in \Sigma$ כך ש- σ מופיעה ב- w לפחות $\frac{|w|}{2}$ פעמים$\}$

הוכיחו כי שפה לא רגולרית.

פתרון:

בעזרת למת הניפוח לשפות רגולריות:

נניח בשלילה כי שפה זו רגולרית $\Leftarrow$ לפי למת הניפוח, קיים מספר טבעי $n_0 \geq 1$ ומתקיים: לכל מילה $w \in L_2$ כך ש- $|w| \geq n_0$ קיימת חלוקה $w = xyz$ ומתקיימים תנאים 1,2,3 של הלמה.

נבחר מילה $w = a^{2n_0} b^{2n_0} c^{4n_0} \in L_2$ מתקיים $|w| = 8n_0 \geq n_0$ לכן קיימת חלוקה $w = xyz$ שמקיימת תנאים 1,2,3 של הלמה.

מתנאי 3 ו- 1 נובע כי בכל חלוקה y סדרה לא ריקה של a-ים.

אם נבחר $k = 3$ נקבל $a^{2n_0+2|y|} b^{2n_0} c^{4n_0}$. מתקיים $\frac{\left|a^{2n_0+2|y|} b^{2n_0} c^{4n_0}\right|}{2} = \frac{8n_0 + 2|y|}{2} = 4n_0 + |y|$

$$\# a = 2n_0 + 2|y| = (2n_0 + |y|) + |y| \leq 3n_0 + |y| < 4n_0 + |y|$$

$$\# b = 2n_0 < 4n_0 + |y|$$

$$\# c = 4n_0 < 4n_0 + |y|$$

לכן $a^{2n_0+2|y|} b^{2n_0} c^{4n_0} \notin L_2$.

כל חלוקה של מילה $a^{2n_0}b^{2n_0}c^{4n_0} \in L_2$ לא יכולה לקיים בוזמנית תנאים 1,2,3 של הלמה

$\Leftarrow$ שפה L_2 לא רגולרית.

בעזרת משפט Myhill–Nerode:

קיימת סדרת מילים אינסופית $i = 1,2,3,\ldots$ $w_i = a^i b^i$

נוכיח כי מתקיים: $w_i \not\equiv_L w_j$ לכל $i \neq j$.

נגדיר $z = b^{2\cdot\min(i,j)}$ מתקיים:

אם $i > j$ אז $z = c^{2\cdot j}$. מתקיים $w_j z = a^j b^j c^{2j}$ בשפה. אבל $w_i z = a^i b^i c^{2j}$ לא בשפה כי

$\frac{|w_i z|}{2} = \frac{2i+2j}{2} = i+j$ אבל $\#a = \#b = i < i+j = \frac{|w_i z|}{2}$ ובנוסף $\#c = 2j < i+j = \frac{|w_i z|}{2}$.

אם $i < j$ אז $z = c^{2i}$. מתקיים $w_i z = a^i b^i c^{2i}$ בשפה. אבל $w_j z = a^j b^j c^{2i}$ לא בשפה.

תרגיל 3 (מבחן מועד א' סמסטר ב' 2010): נתונה שפה

$L_3 = \{w \in \{0,1\}^* :$ שרשור של שני פלינדרומים $w\}$

הוכיחו כי שפה לא רגולרית.

פתרון:

נוכיח טענת עזר: למילה מהצורה $0^m 1 0^n 11$ כאשר $1 \le m < n$ לא קיימת חלוקה לשני פלינדרומים.

כל החלוקות של המילה:

- $0^{m-l}10^n11$ ו- 0^l כאשר $0 \le l < m$. מילה $0^{m-l}10^n11$ לא פלינדרום (כי מתחילה ב-0 ומסתיימת ב-1).
- 10^n11 ו- 0^m. מילה 10^n11 לא פלינדרום כי $n \neq 0$.
- 0^n11 ו- 0^m1. מילה 0^m1 לא פלינדרום (כי מתחילה ב-0 ומסתיימת ב-1).
- $0^{n-l}11$ ו- 0^m10^l. 0^m10^j פלינדרום רק אם $l = m$ לכן $0^{n-l}11 = 0^{\overbrace{n-m}^{\ge 0}}11$ והיא לא פלינדרום (כי מתחילה ב-0 ומסתיימת ב-1).
- 1 ו- 0^m10^n1 מילה 0^m10^n1 לא פלינדרום (כי מתחילה ב-0 ומסתיימת ב-1).

בעזרת למת הניפוח לשפות רגולריות:

נניח בשלילה כי שפה זו רגולרית $\Leftarrow$ לפי למת הניפוח, קיים מספר טבעי $n_0 \geq 1$ ומתקיים: לכל מילה $w \in L_3$ כך ש- $|w| \geq n_0$ קיימת חלוקה $w = xyz$ ומתקיימים תנאים 1,2,3 של הלמה.

נבחר מילה $w = 0^{n_0+1}10^{n_0+1}11$. מילה זו בשפה L_3 כי $0^{n_0+1}10^{n_0+1}$ פלינדרום וגם 11 פלינדרום. בנוסף מתקיים $|w| = 2 \cdot n_0 + 5 > n_0$.

מתנאי 3 ו- 1 נובע כי בכל חלוקה y סדרה לא ריקה של 0-ים.

אם נבחר $k = 0$ נקבל $0^{n_0+1-|y|}10^{n_0+1}11$.

מתקיים $1 \leq n_0 + 1 - |y| < n_0 + 1$ לכן ניתן להשתמש בטענת עזר. נקבל כי למילה $0^{n_0+1-|y|}10^{n_0+1}11$ לא קיימת חלוקה לשני פלינדרומים.

מסקנה: מילה $0^{n_0+1-|y|}10^{n_0+1}11$ לא בשפה.

כל חלוקה של מילה $0^{n_0+1}10^{n_0+1}11 \in L_3$ לא יכולה לקיים בוזמנית תנאים 1,2,3 של הלמה

$\Leftarrow$ שפה L_3 לא רגולרית.

בעזרת משפט Myhill–Nerode:

קיימת סדרת מילים אינסופית $i = 1,2,3,\ldots$ $w_i = 0^i$

נוכיח כי מתקיים: $w_i \not\equiv_L w_j$ לכל $i \neq j$.

נגדיר $z = 10^{\max(i,j)}11$ מתקיים:

אם $i > j$ אז $z = 10^i 11$. מתקיים $w_i z = 0^i 10^i \cdot 11$ בשפה. אבל $w_j z = 0^j 10^i 11$ לא בשפה לפי טענת עזר.

אם $i < j$ אז $z = 10^j 11$. מתקיים $w_i z = 0^i 10^j 11$ לא בשפה לפי טענת עזר. אבל $w_j z = 0^j 10^j \cdot 11$ בשפה.

שפות חסרות הקשר

אוטומט מחסנית (לא דטרמיניסטי אם לא מצוין אחרת) – הרחבת מודל של אוטומט סופי לא דטרמיניסטי ע"י הוספת מחסנית (אמצעי אחסון הפועל בשיטה LIFO) מגודל לא חסום. מעבר למצב הבא מתבצע כפונקציה של המצב הנוכחי, אות הקלט ואות שנמצאת בראש המחסנית.

הערה: א"ב הקלט לא חייב להיות זהה לא"ב המחסנית.

נגדיר: **מילה מתקבלת ע"י א"מ** כאשר קיים לה מסלול חישוב שמסתיים במצב מקבל ומחסנית ריקה. (קיימות גם הגדרות אחרות של מוסג "מילה מתקבלת ע"י א"מ" בהן לא נשתמש בקורס זה)

קיימות שפות שמתקבלות ע"י אוטומט מחסנית (לא דטרמיניסטי) אבל לא קיים להן אוטומט מחסנית דטרמיניסטי (בשונה מאוטומטים סופיים שהשימוש באי-דטרמיניזם לא מוסיף להם כוח חישובי נוסף).

תיאור גראפי של א"מ:

משמעות של סימון $x,\alpha \to \beta$ על הקשת הוא שמעבר בין מצבים מתבצע רק אם אות הקלט היא x ובראש המחסנית נמצאת אות α. בנוסף, בזמן המעבר אות α נמחקת מראש המחסנית ומכניסים במקומה אות β ($POP(\alpha), PUSH(\beta)$).

שימוש ב- ε:

$x,\varepsilon \to \beta$ מעבר מתבצע כאשר אות הקלט היא x. בנוסף, מכניסים β לראש המחסנית ללא מחיקה של אות שהייתה בראש המחסנית ($PUSH(\beta)$).

$x,\alpha \to \varepsilon$ מעבר מתבצע כאשר אות הקלט היא x ובראש המחסנית נמצאת אות α. בנוסף, מוחקים α מראש המחסנית ($POP(\alpha)$).

$x,\varepsilon \to \varepsilon$ מעבר מתבצע כאשר אות הקלט היא x. המחסנית לא משתנה.

$\varepsilon,\alpha \to \beta$ מעבר ε שניתן להשתמש בו אם בראש המחסנית נמצאת אות α. בנוסף, בזמן המעבר אות α נמחקת מראש המחסנית ומכניסים במקומה אות β ($POP(\alpha), PUSH(\beta)$).

דקדוק חסר הקשר – סדרה של כללים ליצירת מילים מעל א"ב נתון.

דח"ה מורכב מ-

- קבוצה סופית V של נונטרמינלים (משתנים) שמסמנים אותם בד"כ באותיות לטיניות גדולות. בנוסף, נונטרמינל אחד מוגדר כ-"נונטרמינל התחלתי" ובד"כ מסומן ב- S.
- קבוצה סופית Σ של טרמינלים (אותיות).
- קבוצה סופית R של כללי גזירה. כל כלל גזירה הוא מהצורה:
  נונטרמינל $\rightarrow$ ε או סדרה סופית של טרמינלים ונונטרמינלים

לדוגמה,

$$V = \{S, A\}$$
$$\Sigma = \{a, b\}$$
$$R:$$
$$S \rightarrow abS$$
$$S \rightarrow A$$
$$A \rightarrow baS$$
$$A \rightarrow \varepsilon$$

ניתן לכתוב כללי גזירה בצורה מקוצרת :

$$S \rightarrow abS \mid A$$
$$A \rightarrow baS \mid \varepsilon$$

הפעלת כלל גזירה (צעד גזירה) – החלפת מחרוזת ששווה לצד שמאל של כלל גזירה (במקרה של דקדוק חסר הקשר בצד שמאל יש רק נונטרמינל אחד) במחרוזת שנמצאת בצד ימין של הכלל.

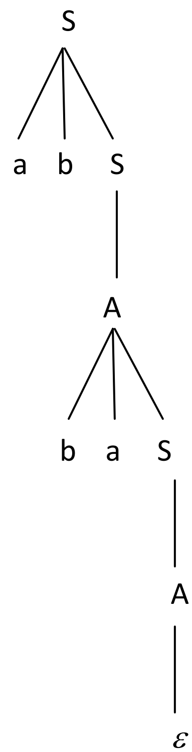
לדוגמה, $baA \Rightarrow babaS$ הפעלת כלל גזירה $A \rightarrow baS$.

דח"ה גוזר מילה (סדרה של טרמינלים) – אם היא מתקבלת מנונטרמינל התחלתי S ע"י הפעלת כללי גזירה מספר סופי של פעמים.

לדוגמה, $S \overset{S\rightarrow abS}{\Rightarrow} abS \overset{S\rightarrow A}{\Rightarrow} abA \overset{A\rightarrow baS}{\Rightarrow} abbaS \overset{S\rightarrow A}{\Rightarrow} abbaA \overset{A\rightarrow \varepsilon}{\Rightarrow} abba\varepsilon = abba$ מסמנים $S \Rightarrow^* abba$.

שפה של דח"ה G - $L(G) = \{w \in \Sigma^* : S \Rightarrow^* w\}$. כלומר, כל סדרות של טרמינלים שדקדוק G גוזר.

ניתן להציג גזירה של מילה בצורה גראפית בעזרת **עץ גזירה**



לכל גזירה של המילה מתאים עץ גזירה אחד. אבל לעץ גזירה אחד יכולים להתאים מספר גזירות של המילה.

אם בשפה של דח"ה קיימת מילה שיש לה יותר מעץ גזירה אחד, דקדוק כזה נקרא **רב-משמעי**.

אם לכל מילה בשפה של דח"ה יש רק עץ גזירה אחד, נקרא לדקדוק **חד-משמעי**.

יתכן מצב בו לשפה יש גם דקדוקים רב-משמעיים וגם חד-משמעיים. אבל קיימות שפות שכל דקדוקים שגוזרים אותן הם רב-משמעיים.

משפט: שפה L שמתקבלת ע"י אוטומט מחסנית $\Leftrightarrow$ קיים דקדוק חסר הקשר שגוזר L.

הגדרה:

- **שפה חסרת הקשר** – קיים דקדוק חסר הקשר שגוזר אותה.

טענה: אם L שפה חסרת הקשר ו- R שפה רגולרית, אזי $L \cap R$ שפה חסרת הקשר.

חשוב:

- חיתוך של שתי שפות חסרות לא בהכרח שפה חסרת הקשר.
- משלים של שפה חסרת הקשר לא בהכרח שפה חסרת הקשר.

הגדרה:

- דקדוק חסר הקשר (V,Σ,R,S) **בצורה נורמאלית של חומסקי** אם מתקיים:
  - כל כלל גזירה מהצורה $A \to BC$ או $A \to \alpha$ כאשר $A,B,C \in V,\ \ B,C \neq S,\ \alpha \in \Sigma$
  - ε יכול להיגזר רק מהנונטרמינל התחלתי. כלומר, $S \to \varepsilon$.

משפט: לכל דקדוק חסר הקשר G קיים דקדוק חסר הקשר בצורה נורמאלית של חומסקי G' כך שמתקיים $L(G') = L(G)$ (קיים אלגוריתם יעיל לבניית G' בהינתן G).

תרגיל 1: נתונה שפה $L_1 = \{a^n b^m c^{n-m} : n \geq m \geq 0\}$

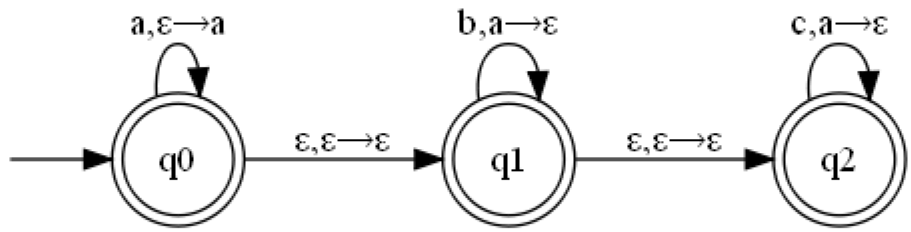
תכננו א"מ ודח"ה לשפה.

פתרון:

א"מ:

מתקיים $a^n \underbrace{b^m c^{n-m}}_{n}$. כלומר, מילה מהצורה $a^* b^* c^*$ מתקבלת אמ"ם מספר a-ים שווה למספר b-ים ו- c-ים ביחד.

לכן, א"מ מחסנית שנבנה יפעל באופן הבא: על כל a שנקרא נכניס a למחסנית ועל כל b או c נוציא a מראש המחסנית.



דח"ה:

מתקיים $\underbrace{a^{n-m}a^m}_{n} b^m c^{n-m}$. דח"ה לשפה:

$$S \to aSc \mid A$$
$$A \to aAb \mid \varepsilon$$

תרגיל 2: נתונה שפה $L_2 = \{\alpha : \alpha$ ביטוי רגולרי מעל א"ב $\{0,1\}\}$

האם שפה רגולרית? אם לא, האם ח"ה?

פתרון:

חיתוך של שפה L_2 עם שפה רגולרית $\{(^n\varnothing(*))^m : n,m>0\} = \{(\varnothing^*),(\varnothing^*)^*),(\varnothing^*)^*)^*),\ldots,((\varnothing^*),\ldots\}$

הוא שפה לא רגולרית $\{(^n\varnothing(*))^n : n>0\}$ (הוכחה בעזרת למת הניפוח לשפות רגולריות).

לכן שפה L_2 לא רגולרית.

דח"ה לשפה L:

$$S \to 0,1,\varepsilon,\varnothing$$
$$S \to (S \cup S)$$
$$S \to (S \circ S)$$
$$S \to (S*)$$

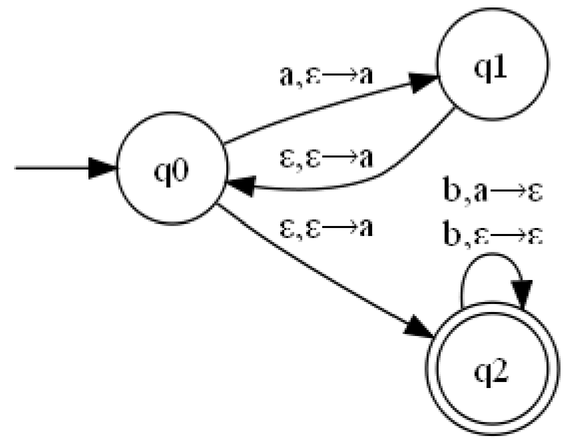
הערה: תווים $0,1,\varepsilon,\varnothing,(,),\cup,\circ,*$ הם טרמינלים.

תרגיל 3: נתונה שפה $L_3 = \{a^i b^j : i \neq j \ \wedge \ 2i \neq j\}$ האם שפה ח"ה?

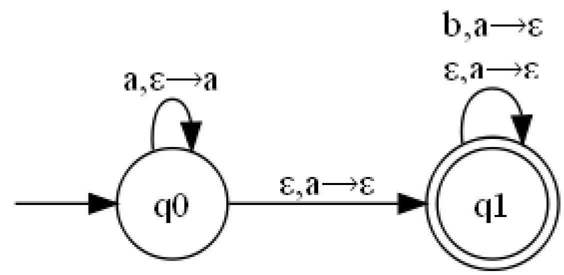
פתרון:

מתקיים $L_3 = \{a^i b^j : i \neq j \ \wedge \ 2i \neq j\} = \{a^i b^j : j > 2i\} \cup \{a^i b^j : j < i\} \cup \{a^i b^j : i < j < 2i\}$

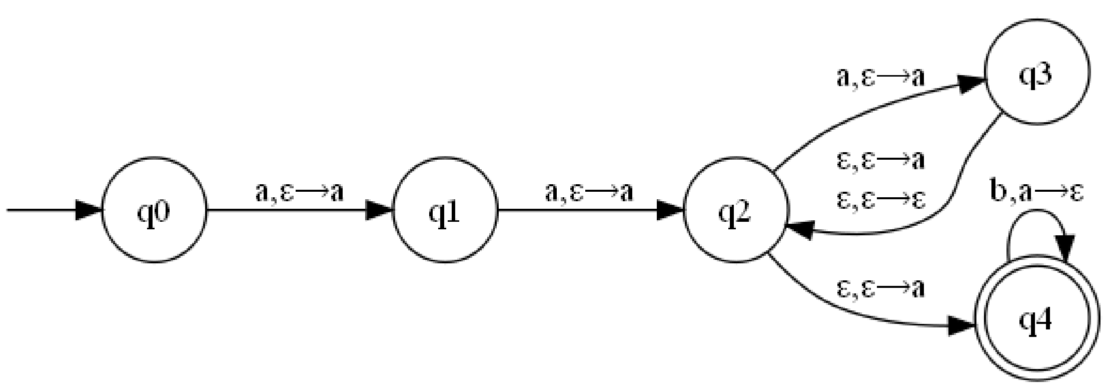
$\{a^i b^j : j > 2i\}$ שפה ח"ה:



$\{a^i b^j : j < i\}$ שפה ח"ה:



$\{a^i b^j : i < j < 2i\}$ שפה ח"ה:



לכן L_3 שפה חסרת הקשר כאיחוד סופי של שפות חסרות הקשר.

תרגיל 4: נתונה שפה $L_4=\{w\in\{0,1\}^*:\#0=\#1\}$

האם שפה ח"ה?

פתרון:

דח"ה לשפה: $S\to 0S1S\,|\,1S0S\,|\,\varepsilon$ נקרא לו G.

הוכחת נכונות:

$L(G)\subseteq L_4$:

טענה: לאחר $n=1,2,\ldots$ צעדי גזירה תמיד נקבל מחרוזת של טרמינלים ונונטרמינלים שמכילה מספר שווה של 0-ים ו-1-ים. מסקנה: כל מחרוזת טרמינלים שנקבל לאחר מספר סופי של צעדי גזירה מכילה מספר שווה של 0-ים ו-1-ים.

אם $n=1$ מחרוזות שנקבל: $0S1S,1S0S,\varepsilon$.

אם לאחר n צעדי גזירה קיבלנו מחרוזת עם מספר שווה של 0-ים ו-1-ים. אזי בצעד גזירה הבא:

- אם נשתמש בכלל גזירה $S\to\varepsilon$ מספר 0-ים ו1-ים לא השתנה.
- אם נשתמש בכלל $S\to 0S1S$ מספר 0-ים גדל ב-1 וגם מספר 1-ים גדל ב-1. לכן שוויון נשמר.
- אם נשתמש בכלל $S\to 1S0S$ מספר 0-ים גדל ב-1 וגם מספר 1-ים גדל ב-1. לכן שוויון נשמר.

לכן גם לאחר $n+1$ צעדי גזירה מספר נקבל מחרוזת עם מספר שווה של 0-ים ו-1-ים.

$L(G)\supseteq L_4$:

נוכיח באינדוקציה על אורך המילה k ש-G גוזר כל מילה ב-L_4.

הערה: כל מילה ב-L_4 מאורך זוגי.

$k=0$

$S\Rightarrow\varepsilon$ **לכן** $\varepsilon\in L(G)$

$k=2$

$S\Rightarrow 0S1S\Rightarrow 0\varepsilon 1S\Rightarrow 0\varepsilon 1\varepsilon=01$ **לכן** $01\in L(G)$

$S\Rightarrow 1S0S\Rightarrow 1\varepsilon 0S\Rightarrow 1\varepsilon 0\varepsilon=10$ **לכן** $10\in L(G)$

נניח שכל מילה ב-L_4 מאורך n ב-$L(G)$ נוכיח שגם כל מילה ב-L_4 מאורך $n+2$ ב-$L(G)$:

נתונה מילה w ב-L_4 מאורך $n+2$. בה"כ נניח שהיא מתחילה ב-0.

נגדיר: w_j שווה ל-$\#0-\#1$ ברישא מגודל j של w.

$w(j)$ - תו במקום j ב- w.

יהי $1<i\le|w|$ מספר קטן ביותר כך שמתקיים $w_i=0$ (i קיים כי ב- w מספר שווה של 0-ים ו-1-ים)

טענת עזר: אם w מתחילה ב- 0 אזי מתקיים $w(i)=1$ (אם $w(i)=0$ מתקיים $w_{i-1}=-1$ וגם $w_1=1$
לכן, קיים $i'<i$ כך ש-$w_{i'}=0$ בסתירה למינימאליות של i).

נקבל חלוקה של מילה $w=0\circ w(2,i-1)\circ 1\circ w(i+1,n+2)$ כאשר $w(2,i-1),w(i+1,n+2)\in L_4$.

לפי הנחת האינדוקציה $S\Rightarrow^* w(2,i-1), S\Rightarrow^* w(i+1,n+2)$ לכן
$S\Rightarrow 0S1S\Rightarrow^* 0\circ w(2,i-1)\circ 1\circ w(i+1,n+2)=w$

באופן דומה אם w מתחילה ב-1.

תרגיל 5 (מתוך מבחן סמסטר ב' 2010):

$\Sigma=\{0,1,2,3,4,5,6,7,8,9\}$. עבור מילה w נגדיר $e(w)$ להיות המילה המתקבלת מ-w ע"י השמטת התווים שערכם אי-זוגי. למשל $e(0073423982)=004282$

עבור שפה L מעל הא"ב Σ נגדיר $e(L)=\{e(w):w\in L\}$

הוכיחו: אם L שפה חסרת הקשר אז $e(L)$ חסרת הקשר.

פתרון:

בהינתן א"מ M לשפה L נגדיר א"מ M_e לשפה $e(L)$.

מבנה של M_e יהיה זהה למבנה של M פרט לשינוי הבא:

כל מעבר ב- M מהצורה $x,y\to z$ כאשר x תו שערכו אי-זוגי נחליף ב-M_e במעבר מהצורה $\varepsilon,y\to z$.

למעברי ε שהוספנו נקרא מעברי ε "אי-זוגיים" כדי להבדיל אותם ממעברי ε שהיו ב- M (ונשארו גם ב-M_e)

סקיצה של הוכחה:

אם $v\in e(L)$ אזי קיימת $u\in L$ כך שמתקיים $v=e(u)$.

$u \in L$ לכן קיים מסלול חישוב של M על u שמסתיים במצב מקבל עם מחסנית ריקה. אם בזמן הריצה של M_e על v נשתמש באותם מעברים על תווים שערכם זוגי ובמעברי ε "אי-זוגיים" שתואמים למעברים על תווים ב- u שערכם אי-זוגי, נסיים את הריצה במצב מקבל של M_e ומחסנית ריקה (כי פעולות על המחסנית היו זהות לפעולות של M על u). כלומר, א"מ M_e מקבל את v.

אם מילה v מתקבלת ב- M_e אזי קיים מסלול חישוב מקבל. אבל בהינתן מסלול חישוב מקבל ב- M_e אפשר להתאים מסלול חישוב מקבל ב- M: במקום שימוש במעברי ε "אי-זוגיים" נשתמש במעברים מקוריים שתואמים למעברים אלה. מילה u שתואמת למסלול חישוב זה מקיימת: $u \in L$ ובנוסף, $v = e(u)$.

סיכום תכונות סגור של שפות חסרות הקשר:

אם L_1, L_2 שפות חסרות הקשר ו- R שפה רגולרית , אזי גם

- $L_1 \cup L_2$
- $L_1 \cap R$
- $L_1 \circ L_2$
- L_1^*
- L_1^R

שפות חסרות הקשר.

שפות לא חסרות הקשר

למת הניפוח לשפות חסרות הקשר: לכל שפה ח"ה L קיים מספר טבעי $n_0 \geq 1$ ומתקיים:

לכל מילה $w \in L$ כך ש- $|w| \geq n_0$ קיימת חלוקה $w = uvxyz$ שמקיימת:

4. $|vy| \geq 1$
5. $uv^k xy^k z \in L$ לכל $k \geq 0$
6. $|vxy| \leq n_0$

תרגיל 1 (מתוך בוחן סמסטר א' 2010):

$b_i \in \{0,1\}^*, u_i \in \{1\}^*$ הייצוג אונארי ובינארי של מספר $i \in \mathbb{N}$.

נגדיר: $L_1 = \{u_i b_i : i \in \mathbb{N}\}$

דוגמה למילים בשפה: $u_1 b_1 = 11,\ u_2 b_2 = 1110,\ u_3 b_3 = 11111, \ldots$

האם שפה L_1 ח"ה ?

פתרון:

נניח בשלילה כי L_1 ח"ה, אזי $L_1 \cap \{1\}^* = \{1^{2^i - 1 + i} : i \in \mathbb{N}\}$ שפה ח"ה (כחיתוך של שפה רגולרית ושפה ח"ה).

נוכיח בעזרת למת הניפוח לשפות ח"ה כי השפה $L_1 \cap \{1\}^* = \{1^{2^i - 1 + i} : i \in \mathbb{N}\}$ לא ח"ה.

נניח בשלילה כי שפה זו ח"ה $\Leftarrow$ לפי למת הניפוח, קיים מספר טבעי $n_0 \geq 1$ ומתקיים: לכל מילה $w \in L_1$ כך ש- $|w| \geq n_0$ קיימת חלוקה $w = uvxyz$ ומתקיימים תנאים 1,2,3 של הלמה.

נבחר מילה ב- $L_1 \cap \{1\}^*$ $w = 1^{2^{n_0} - 1 + n_0}$. מתקיים $|w| = 2^{n_0} - 1 + n_0 \geq n_0$ לכן קיימת חלוקה $w = uvxyz$ שמקיימת תנאים 1,2,3 של הלמה.

בכל חלוקה vy סדרה של 1-ים ומתנאי 3 ו- 1 נובע $1 \leq |vy| \leq n_0$.

אם נבחר $k = 2$ נקבל $1^{2^{n_0} - 1 + n_0 + |vy|}$

מתקיים: $2^{n_0+1}-1+(n_0+1) \;>\; 2^{n_0}-1+n_0+|vy| \;>\; 2^{n_0}-1+n_0$.

לכן, לכל $i \in \mathbb{N}$ $\;1^{2^{i}-1+i} \neq 1^{2^{n_0}-1+n_0+|vy|}$.

$\Leftarrow$ המילה $1^{2^{n_0}-1+n_0+|vy|}$ לא בשפה $L_1 \cap \{1\}^* = \{1^{2^i-1+i} : i \in \mathbb{N}\}$ בסתירה לנכונות הלמה.

מסקנה: שפה $L_1 \cap \{1\}^*$ לא ח"ה ולכן גם L_1 לא ח"ה.

תרגיל 2: נתונה שפה $L_2 = \{a^n b^m :$ מספר ראשוני $n+m$ ומתקיים $n,m \ge 0\}$

האם שפה ח"ה ?

פתרון:

נוכיח בעזרת למת הניפוח לשפות ח"ה כי השפה L_2 לא ח"ה.

נניח בשלילה כי שפה זו ח"ה $\Leftarrow$ לפי למת הניפוח, קיים מספר טבעי $n_0 \ge 1$ ומתקיים: לכל מילה $w \in L_2$ כך ש- $|w| \ge n_0$ קיימת חלוקה $w = uvxyz$ ומתקיימים תנאים 1,2,3 של הלמה.

נבחר מילה ב- L_2 $w = a^p$ כאשר p מספר ראשוני מינימאלי שמקיים $p \ge n_0$ (p קיים כי יש מספר אינסופי של מספרים ראשוניים). מתקיים $|w| = p \ge n_0$ לכן קיימת חלוקה $w = uvxyz$ שמקיימת תנאים 1,2,3 של הלמה.

מתנאי 1 נובע כי בכל חלוקה vy סדרה של a-ים ומתקיים $1 \le |vy|$.

אם נבחר $k = p+1$ נקבל $a^{p+p\cdot|vy|}$ אבל מספר $p + p\cdot|vy| = p\underbrace{(1+|vy|)}_{\ge 2}$ לא ראשוני ולכן מילה $a^{p+p\cdot|vy|}$ לא בשפה בסתירה לנכונות הלמה.

מסקנה: שפה L_2 לא ח"ה.

תרגיל 3: נתונה שפה $L_3 = \{0^i \# 0^{2i} \# 0^{3i} : i \in \mathbb{N}\}$

האם שפה ח"ה ?

פתרון:

נוכיח בעזרת למת הניפוח לשפות ח"ה כי השפה L_3 לא ח"ה.

נניח בשלילה כי שפה זו ח"ה $\Leftarrow$ לפי למת הניפוח, קיים מספר טבעי $n_0 \geq 1$ ומתקיים: לכל מילה $w \in L_3$ כך ש- $|w| \geq n_0$ קיימת חלוקה $w = uvxyz$ ומתקיימים תנאים 1,2,3 של הלמה.

נבחר מילה ב- L_3 $w = 0^{n_0} \# 0^{2n_0} \# 0^{3n_0}$. מתקיים $|w| = 6n_0 + 2 \geq n_0$ לכן קיימת חלוקה $w = uvxyz$ שמקיימת תנאים 1,2,3 של הלמה.

קטעי הניפוח לא יכולים להכיל תו # (כי לאחר הניפוח נקבל מילה מהצורה לא נכונה) לכן כל קטע הניפוח נמצא רק ברצף 0-ים אחד.

מספר קטעי הניפוח לא ריקים הוא לכל היותר 2. לכן, בכל חלוקה של המילה יהיה לפחות רצף 0-ים אחד שלא מכיל קטע ניפוח (ולפחות רצף אחד שמכיל קטע ניפוח).

לאחר הניפוח (לדוגמה, $k = 2$) נקבל מילה בה יחס בין מספר 0-ים לא נשמר. כלומר, מילה לא בשפה.

מסקנה: שפה L_3 לא ח"ה.

תרגיל 4: נתונה שפה $L_4 = \{\alpha \# \beta : \alpha, \beta \in \{0,1\}^*, |\alpha| = |\beta|, \alpha \neq \beta\}$

האם שפה ח"ה ?

פתרון:

למרות ששפה דומה $\{\alpha\beta : \alpha, \beta \in \{0,1\}^*, |\alpha| = |\beta|, \alpha \neq \beta\}$ היא חסרת הקשר, נראה בעזרת למת הניפוח לשפות ח"ה כי שפה L_4 לא.

נניח בשלילה כי שפה זו ח"ה $\Leftarrow$ לפי למת הניפוח, קיים מספר טבעי $n_0 \geq 1$ ומתקיים: לכל מילה $w \in L_4$ כך ש- $|w| \geq n_0$ קיימת חלוקה $w = uvxyz$ ומתקיימים תנאים 1,2,3 של הלמה.

נבחר מילה ב- L_4 $w = 1^{n_0!+n_0} 0^{n_0} \# 1^{n_0} 0^{n_0!+n_0}$ מתקיים $|w| = 2 \cdot n_0! + 4n_0 + 1 \geq n_0$ לכן קיימת חלוקה $w = uvxyz$ שמקיימת תנאים 1,2,3 של הלמה.

קטעי הניפוח לא יכולים להכיל תו # (כי לאחר הניפוח נקבל מילה מהצורה לא נכונה).

בנוסף, אם גדלים של קטעי הניפוח בשני חלקי מילה לא שווים, לאחר הניפוח נקבל מילה לא בשפה.

מסקנה: v נמצא בחלק שמאלי של המילה, y נמצא בחלק ימני של המילה ומתקיים $|v|=|y|$.

מתנאי 3 ו- 1 נובע $1\le|vy|\le n_0$ לכן v מכיל רק 0-ים, y מכיל רק 1-ים.

אם נבחר $k=\frac{n_0!}{|v|}+1$ נקבל $1^{n_0!+n_0}0^{n_0+\frac{n_0!}{|v|}|v|}\#1^{n_0+\frac{n_0!}{|v|}|y|}0^{n_0!+n_0}=1^{n_0!+n_0}0^{n_0!+n_0}\#1^{n_0!+n_0}0^{n_0!+n_0}$

מילה לא בשפה בסתירה לנכונות הלמה.

מסקנה: שפה L_4 לא ח"ה.

הערה: בתרגילים קודמים ראינו שקיים k מסוים (לדוגמה, שווה ל-2) כך שלכל חלוקה של המילה תנאי 2 של הלמה לא מתקיים. בתרגיל זה לכל חלוקה קיים k שונה!

מכונת טיורינג (דטרמיניסטית)

דוגמה למכונת טיורינג לשפה $\{w \in \{0,1\}^* : \#0 = \#1\}$.

רעיון: נפעל על פי האלגוריתם הבא:

1. 1. אם מילת הקלט היא מילה ריקה – נקבל.
2. אחרת,

1.2.1 אם תו הראשון הוא 0 נחליף אותו בסימן $, נזיז את הראש הקורא/כותב ימינה ונחליף 1 הראשון שנמצא בסימן x (אם הגענו לסוף הקלט ולא מצאנו 1 – נדחה).

1.2.2 אם תו הראשון הוא 1 נחליף אותו בסימן $, נזיז את הראש הקורא/כותב ימינה ונחליף 0 הראשון שנמצא בסימן x (אם הגענו לסוף הקלט ולא מצאנו 0 – נדחה).

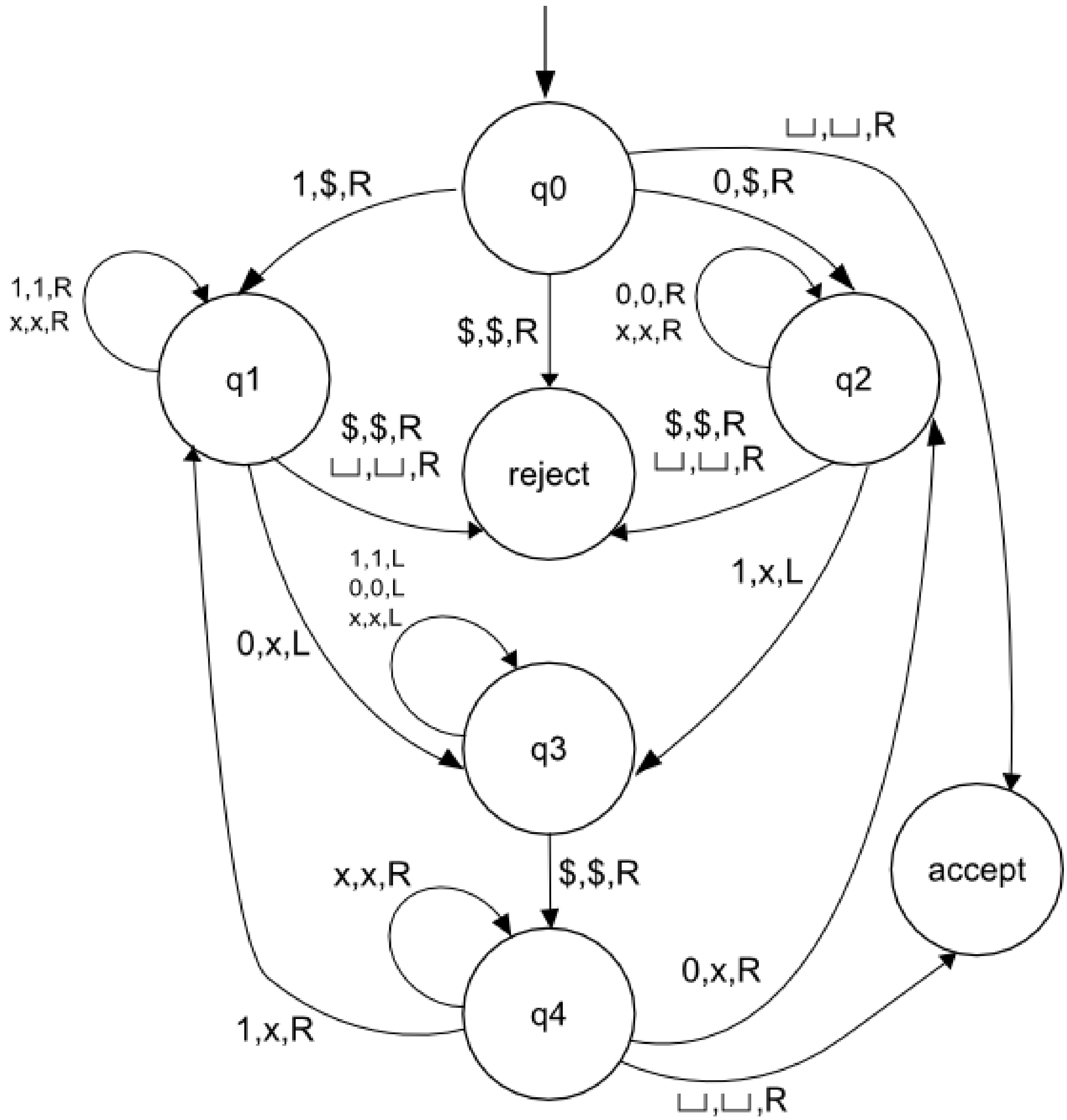
2. נחזיר את הראש הקורא/כותב לתחילת הסרט. כלומר, שמאלה עד לסימן $.
3. נחפש תו ראשון שונה מ-x או $:
   1. אם תו זה הוא 0 נחליף אותו בסימן x, נזיז את הראש הקורא/כותב ימינה ונחליף 1 הראשון שנמצא בסימן x (אם הגענו לסוף הקלט ולא מצאנו 1 – נדחה), נחזור לשלב 2.
   2. אם תו זה הוא 1 נחליף אותו בסימן x, נזיז את הראש הקורא/כותב ימינה ונחליף 0 הראשון שנמצא בסימן x (אם הגענו לסוף הקלט ולא מצאנו 1 – נדחה), נחזור לשלב 2.
   3. אם הגענו לסוף הקלט ולא מצאנו תו כזה – נקבל.

הרצת אלגוריתם על המילה 00011011:	הרצת אלגוריתם על המילה 10011011:
00011011	10011011
$0011011	$0011011
$00x1011	$x011011
$x0x1011	$xx11011
$x0xx011	$xxx1011
$xxxx011	$xxxx011
$xxxx0x1	$xxxxx11
$xxxxxx1	$xxxxxx1
$xxxxxxx	
נקבל לפי 3.3	נדחה לפי 3.2

תיאור גראפי של מכונת טיורינג שפועלת על פי האלגוריתם הנ"ל:

משמעות של סימון x, y, R על הקשת הוא שמעבר בין מצבים מתבצע אם קראנו על הסרט x ובנוסף, כותבים y במקום x והראש הקורא/כותב של מ"ט זז ימינה צעד אחד (x, y, L כאשר הוא זז שמאלה).

אחרי מילת הקלט על הסרט מופיעה סדרה אינסופית של ␣.


q0
1,$,R
0,$,R
⊔,⊔,R
1,1,R
x,x,R
q1
$,$,R
0,0,R
x,x,R
q2
$,$,R
⊔,⊔,R
reject
$,$,R
⊔,⊔,R
1,1,L
0,0,L
x,x,L
0,x,L
1,x,L
q3
$,$,R
accept
x,x,R
q4
0,x,R
1,x,R
⊔,⊔,R

שפות כריעות/ניתנות לקבלה ע"י מ"ט

הגדרות:

- מ"ט M **מקבלת** קלט x אם בזמן ריצה שלה על x היא עוצרת ונכנסת למצב q_{accept}.
- מ"ט M **דוחה** קלט x אם בזמן ריצה שלה על x היא עוצרת ונכנסת למצב q_{reject}.
- מ"ט M **לא מקבלת** קלט אם x בזמן ריצה שלה על x היא נכנסת ללולאה אינסופית או דוחה.
- **שפה של מ"ט** M: $L(M) = \{w \in \Sigma^* : w$ מקבלת $M\}$.

- מ"ט M **מקבלת את השפה** L אם מתקיים $L(M) = L$. כלומר, על כל $x \in L$ M עוצרת ונכנסת למצב q_{accept}, ועל כל $x \notin L$ עוצרת ונכנסת למצב q_{reject} או נכנסת ללולאה אינסופית.
- שפה נקראת **ניתנת לקבלה** (**Turing-acceptable/recognizable/Recursively Enumerable**) אם קיימת מ"ט שמקבלת אותה.

- מ"ט M **מכריעה את השפה** L אם מתקיים $L(M) = L$ <u>והיא עוצרת על כל קלט</u>. כלומר, על כל $x \in L$ M עוצרת ונכנסת למצב q_{accept}, ועל כל $x \notin L$ עוצרת ונכנסת למצב q_{reject}.
- שפה נקראת **כריעה** (**decidable/Recursive**) אם קיימת מ"ט שמכריעה אותה**.**

  הערה: כל שפה כריעה היא גם ניתנת לקבלה.

מתקיים: אם L_1, L_2 שפות ניתנות לקבלה , אזי גם

- $L_1 \cup L_2$
- $L_1 \cap R$
- $L_1 \circ L_2$
- L_1^*

שפות ניתנות לקבלה.

מתקיים: אם L_1, L_2 שפות כריעות , אזי גם

- $L_1 \cup L_2$
- $L_1 \cap R$
- $L_1 \circ L_2$
- L_1^*
- $\overline{L_1}$

שפות כריעות.

הגדרה: מ"ט טיורינג אוניברסאלית U מוגדרת באופן הבא:

1. בדיקה שקלט מהצורה $<M, w>$ כאשר $<M>$ קידוד חוקי של מ"ט ו- w מחרוזת מעל א"ב הקלט של M. אם לא, נדחה.
2. ביצוע סימולציית הריצה של מ"ט M על מחרוזת הקלט w:
   - אם M מקבלת w - נקבל.
   - אם M דוחה w - נדחה.

הערה: אם M לא עוצרת על w, אזי גם U לא עוצרת על קלט $<M, w>$.

$ACCEPT = \{M$ מ"ט, w מחרוזת. M מקבלת את $w : <M, w> \} = L(U)$

לכן לפי הגדרה, שפה $ACCEPT$ ניתנת לקבלה.

משפט 1: שפה $ACCEPT$ לא שפה כריעה. (הוכחה בעזרת ליכסון)

משפט 2: אם שפות $L, \overline{L}$ ניתנות לקבלה אזי הן כריעות.

מסקנה: אם L ניתנת לקבלה ולא כריעה, אזי $\overline{L}$ לא ניתנת לקבלה.

נגדיר: $\{M$ מ"ט, w מחרוזת. M לא מקבלת את $w : <M, w> \} = NOT - ACCEPT$

מתקיים: $\overline{ACCEPT} = NOT - ACCEPT \cup \{$קלט לא מהצורה תקינה$\}$

הערה: ניתן לבדוק ביעילות האם קלט בצורה תקינה. לכן נתייחס למשלים כ-

$\overline{L} = \{$קלט מהצורה תקינה$\} \setminus L$. לדוגמה, $\overline{ACCEPT} = NOT - ACCEPT$.

לפי המשפט מס' 2 שפה $NOT - ACCEPT$ לא ניתנת לקבלה. כלומר, לא ניתן לבנות מ"ט שמקבלת (עוצרת ונכנסת למצב q_{accept}) כל קלט בשפה.

רדוקציית מיפוי (mapping reduction/many-one reduction):

יש רדוקציית מיפוי משפה A לשפה B (סימון $A \leq_m B$) אם קיימת מ"ט R <u>שעוצרת על כל קלט</u> ומתקיים $x \in A \Leftrightarrow R(x) \in B$ (כאשר משמעות של סימון $R(x)$ - פלט של R על x).

משמעות של רדוקציה: מ"ט R מקבלת כקלט מחרוזת ומחזירה כפלט מחרוזת. היא מאפשרת "לתרגם" שאלה "האם x בשפה A?" לשאלה אחרת: "האם $R(x)$ בשפה B?".

טענה 1: אם מתקיים $A \leq_m B$ אזי:

1. אם שפה B כרעיה אזי A כריעה.
   מסקנה: אם שפה A לא כרעיה אזי B לא כריעה.

2. אם שפה B ניתנת לקבלה אזי A ניתנת לקבלה.
   מסקנה: אם שפה A לא ניתנת לקבלה אזי B לא ניתנת לקבלה.

משפט Rice:

אם שפה L מקיימת:

1. $L \subseteq TM = \{<M>:$ מ"ט $M\}$.
2. L לא טריוויאלית. כלומר, $L \neq \emptyset, TM$.
3. לכל שתי מ"ט M_1, M_2 כך ש- $L(M_1) = L(M_2)$ מתקיים: $<M_1> \in L \Leftrightarrow <M_2> \in L$.

אזי L לא כריעה.

הערה: שפה L שמקיימת תנאי 1 ו- 3 של משפט Rice נקראת "תכונה של שפות".

תזכורת:

ניתנת לקבלה	כריעה	שפה
v	x	$ACCEPT=\{<M,w>:$ M מ"ט, w מחרוזת. M מקבלת את $w\}$
x	x	$NOT-ACCEPT=\{<M,w>:$ M מ"ט, w מחרוזת. M לא מקבלת את $w\}$
v	x	$HALT=\{<M,w>:$ M מ"ט, w מחרוזת. M עוצרת על $w\}$
v	x	$NOT-EMPTY=\{<M>:$ M מ"ט. $L(M)\neq\varnothing\}$
x	x	$EMPTY=\{<M>:$ M מ"ט. $L(M)=\varnothing\}$
x	x	$EQUAL=\{<M_1,M_2>:$ M_1,M_2 מ"ט. $L(M_1)=L(M_2)\}$
x	x	$ALL=\{<M>:$ M מ"ט. $L(M)=\Sigma^*\}$
x	x	$ALL_{CFG}=\{<G>:$ G דח"ה. $L(G)=\Sigma^*\}$

תרגיל 1: $L_1=\{<M>:$ M מ"ט וקיימת מילה w כך ש- M מקבלת את w וגם $w^R\}$
האם שפה כריעה? ניתנת לקבלה?

פתרון:

נראה כי שפה ניתנת לקבלה ע"י בניית מ"ט T שמקבלת אותה.

הגדרת $T(<M>)$:
1. בדיקת תקינות הקלט
2. $i\leftarrow 1$
3. ביצוע סימולציה של M על כל מילים מאורך קטן שווה i למשך i צעדי חישוב:
   - אם קיימת מילה שהתקבלה וגם ה-reverse שלה התקבל – נקבל
   - אם לא, $i\leftarrow i+1$ וחוזרים לשלב 2.

נוכיח $L(T)=L_1$:
- אם $<M>\in L_1$ אזי קיימת מילה w כך ש- M מקבלת את w וגם w^R. נניח כי M מקבלת את w לאחר a צעדי חישוב, ואת w^R לאחר b צעדי חישוב.
  לכן, T על קלט $<M>$ כאשר $i\leftarrow \max(a,b,|w|)$ מריצה M על w וגם w^R למשך מספר גדול מספיק של צעדי חישוב כדי ש- M תקבל אותן. לכן, T מקבלת $<M>$.

- אם $<M>\notin L_1$ לכל i לא תהיה מחרוזת שהיא וגם ה-reverse שלה מתקבלת. לכן, מ"ט T מבצעת לולאה אינסופית. כלומר, לא מקבלת $<M>$.

- אם קלט לא קידוד חוקי של מ"ט הוא לא עובר בדיקה בשלב 1 ונדחה ע"י מ"ט T.

מסקנה: שפה ניתנת לקבלה.

האם שפה כריעה?

מתקיים:
1. $L_1 \subseteq TM$
2. שפה לא טריוויאלית: נגדיר: M_{accept} - מ"ט שמקבלת כל מילה. M_{reject} - מ"ט שדוחה כל מילה.
מתקיים: $<M_{accept}> \in L_1$ לכן $L_1 \neq \varnothing$. $<M_{reject}> \notin L_1$ לכן $L_1 \neq TM$.
3. אם מתקיים $L(M_1) = L(M_2)$ נקבל $<M_1> \in L_1 \Leftrightarrow$ קיימת מילה w כך ש- M_1 מקבלת את w וגם $w^R \Leftrightarrow$ קיימת מילה w כך ש- M_2 מקבלת את w וגם $w^R \Leftrightarrow <M_2> \in L_1$.

לכן לפי משפט Rice שפה לא כריעה.

אפשר להוכיח ששפה לא כריעה גם ע"י בניית רדוקציית מיפוי $ACCEPT \leq_m L_1$ (שפה $ACCEPT$ לא כריעה לכן גם L_1 לא כריעה).

נגדיר מ"ט R:

$R(<M,w>) = <M'>$
כאשר מ"ט M' מוגדרת באופן הבא:

$M'(x)$:
1. נריץ את M על w ונפעל כמוה

(כלומר, אם M מקבלת - M' מקבלת, אם M דוחה - M' דוחה, אם M לא עוצרת – גם M' לא עוצרת).

הערה: ההתנהגות של מ"ט M' שהגדרנו לא תלויה בקלט שהיא מקבלת, אלא רק ב- M ו- w.

חשוב: באופן פורמאלי צריך להגדיר התנהגות של R על קלט לא מהצורה $<M,w>$: על קלט כזה נחזיר מחרוזת שלא בשפה L_1. לדוגמה, לא קידוד חוקי של מ"ט או $<M_{reject}>$ שלא ב- L_1.

מתקיים: מ"ט R עוצרת על כל קלט: היא מקבלת כקלט קידוד של מ"ט ומחרוזת ופולטת קידוד של מ"ט.

R לא מריצה מ"ט אחרת, היא מבצעת רק פעולות על קידודים (מחרוזות). תפקידה דומה לתפקיד של מתכנת שמייצר קוד, לדוגמה, בשפת C. גם אם קוד מכיל לולאה אינסופית למתכנת לוקח זמן סופי לכתוב אותו.

- אם $<M,w> \in ACCEPT \Leftarrow$ M מקבלת $w \Leftarrow$ M' מקבלת כל קלט. כלומר, $L(M') = \Sigma^* \Leftarrow$ קיימת מילה w כך ש- M מקבלת את w וגם $w^R \Leftarrow R(x) = <M'> \in L_1$.
- אם $<M,w> \notin ACCEPT \Leftarrow$ M לא מקבלת $w \Leftarrow L(M') = \varnothing \Leftarrow R(x) = <M'> \notin L_1$

מסקנה: L_1 לא כריעה.
שפה L_1 לא כריעה וניתנת לקבלה (מזה נובע ש- $\overline{L_1}$ לא ניתנת לקבלה).

תרגיל 2: $L_2 = \{<M>:$ C מ"ט ולא מקבלת אף תוכנית בשפת $M\}$
האם שפה כריעה? ניתנת לקבלה?

פתרון:
$\overline{L_2} = \{<M>:$ מ"ט וקיימת תוכנית בשפת C שהיא מקבלת $M\}$

$\overline{L_2}$ ניתנת לקבלה.
$T(<M>)$:
1. בדיקת תקינות הקלט
2. $i \leftarrow 1$
3. נריץ מ"ט M על כל מחרוזות באורך קטן שווה i למשך i צעדי חישוב: אם קיימת מחרוזת שמתקבלת והיא תוכנית חוקית ב-C – נקבל
4. $i \leftarrow i+1$ וחוזרים לשלב 2.

מתקיים: $L(T) = \overline{L_2}$.
בנוסף, קיימת רדוקציית מיפוי $ACCEPT \leq_m \overline{L_2}$
$R(<M, w>) = <M'>$
$M'(x)$:
1. נריץ את M על w ונפעל כמוה

מסקנה: $\overline{L_2}$ לא כריעה וניתנת לקבלה. לכן L_2 לא ניתנת לקבלה.

תרגיל 3: $NOT-ALL = \{<M>: L(M) \neq \Sigma^*$,מ"ט $M\}$
האם שפה כריעה? ניתנת לקבלה?

פתרון:
מתקיים: $NOT-ACCEPT \leq_m NOT-ALL$

$R(<M, w>) = <M'>$
$M'(x)$:
1. נריץ את M על w ונפעל כמוה

מתקיים: מ"ט R עוצרת על כל קלט.

- אם $<M, w> \in NOT-ACCEPT \Leftarrow$ M לא מקבלת $w \Leftarrow$ M' לא מקבלת כל קלט.
  כלומר, $L(M') = \varnothing \Leftarrow L(M') \neq \Sigma^* \Leftarrow R(x) = <M'> \in NOT-ALL$.

- אם $<M, w> \notin NOT-ACCEPT \Leftarrow$ M מקבלת $w \Leftarrow L(M') = \Sigma^*$
  $\Leftarrow R(x) = <M'> \notin NOT-ALL$

מסקנה: $NOT-ALL$ לא ניתנת לקבלה.
הערה: ALL שפה לא ניתנת לקבלה וגם המשלים שלה $NOT-ALL$ לא ניתנת לקבלה.

תרגיל 4: $L_4 = \{<M>:$ 1-מ"ט ומקבלת את כל המילים שמתחילות ב $M\}$
האם שפה כריעה? ניתנת לקבלה?

פתרון:

מתקיים: $ALL \leq_m L_4$

$R(<M>) = <M'>$

$M'(x)$:

1. אם קלט x מתחיל ב-1 כלומר, מהצורה $x = 1 \circ y$ נריץ את M על y ונפעל כמוה
2. אחרת, נדחה

מתקיים: מ"ט R עוצרת על כל קלט.

- אם $<M> \in ALL \Leftarrow$ M מקבלת כל קלט $\Leftarrow$ M' מקבלת כל מילים שמתחילות ב-1 (ורק מילים כאלה) $\Leftarrow$ $R(x) = <M'> \in L_4$.
- אם $<M> \notin ALL \Leftarrow$ קיים קלט w ש- M לא מקבלת $\Leftarrow$ M' לא מקבלת קלט $1 \circ w \Leftarrow$ $R(x) = <M'> \notin L_4$

מסקנה: L_4 לא ניתנת לקבלה.

תרגיל 5: $EMPTY_{CFG} = \{<G>: L(G) = \varnothing$,דח"ה $G\}$
האם שפה כריעה? ניתנת לקבלה?

פתרון:

שפה כריעה. מ"ט שמריצה אלגוריתם הבא מכריעה אותה:

נתחזק שתי רשימות זרות של נונטרמינלים : l_1, l_2.
אתחול: כל נונטרמינלים ב-l_1, l_2 ריקה.
צעד 1: כל נונטרמינל ב-l_1 שיש לו כלל גזירה מהצורה "סדרה של טרמינלים $\rightarrow$ נונטרמינל" עובר ל-l_2.
צעד i: כל נונטרמינל ב-l_1 שיש לו כלל גזירה מהצורה
"סדרה של טרמינלים או נונטרמינלים שנמצאים ב-l_2 בשלב $i-1$ $\rightarrow$ נונטרמינל" עובר ל-l_2.

אם בצעד מסוים אין נונטרמינל שעבר ל-l_2 עוצרים ומחזירים:
- "שפה ריקה" אם נונטרמינל התחלתי S ב-l_1.
- "שפה לא ריקה" אם נונטרמינל התחלתי S ב-l_2.

נכונות:
אלגוריתם תמיד עוצר – מספר מכסימלי של צעדים שהוא מבצע כמספר נונטרמינלים ב- G (בכל צעד ל-l_2 עובר לפחות נונטרמינל אחד). זמן ריצה כולל שלו פולינומיאלי בייצוג של G.

נוכיח באינדוקציה שאם נונטרמינל עובר ל-l_2 בשלב i, קיימת מילה (סדרה של טרמינלים) שהוא גוזר:
$i = 1$
אם נונטרמינל עבר ל-l_2 בצעד מספר 1 קיים כלל גזירה שמאפשר לגזור מילה בצעד גזירה אחד.
נניח שטענה נכונה ל-$i = n$ ונוכיח נכונותה ל- $i = n+1$:
אם נונטרמינל עובר ל-l_2 בצעד $i = n+1$ קיים כלל גזירה שמורכב מטרמינלים ונונטרמינלים שנמצאים ב-l_2 בצעד מספר n ולפי הנחת האינדוקציה כל נונטרמינל כזה גוזר מילה. לכן אפשר לגזור מילה מנונטרמינל שעבר בצעד $n+1$.
מסקנה: כל נונטרמינל ב-l_2 גוזר מילה.

נניח בשלילה שלאחר עצירת אלגוריתם קיים נונטרמינל ב-l_1 שגוזר מילה. בכל עץ גזירה שמתאים למילה זו קיים מסלול שמתחיל בשורש, מורכב רק מנונטרמינלים שלא שייכים ל- l_2, ומסתיים בנונטרמינל שגוזר רק סדרה של טרמינלים. בסתירה לאלגוריתם שצריך להוסיף נונטרמינל כזה ל- l_2 בצעד מספר 1.
מסקנה: לאחר עצירת אלגוריתם, כל נונטרמינל ב-l_1 לא גוזר מילה.

מתקיים: פלט "שפה לא ריקה" $\Leftrightarrow$ נונטרמינל התחלתי S ב-l_2 $\Leftrightarrow$ S גוזר מילה $\Leftrightarrow L(G) \neq \varnothing$.

תרגיל 6: $DECIDE = \{<M>:$ מ"ט ועוצרת על כל קלט $M\}$
האם שפה כריעה? ניתנת לקבלה?

פתרון:

חשוב: לא ניתן להשתמש במשפט Rice כי שפה זו לא "תכונה של שפות":
M_{reject} - מ"ט שדוחה (ועוצרת) כל מילה. $\widehat{M}$ - מ"ט שלא עוצרת על כל מילה.
מתקיים: $L(\widehat{M}) = L(M_{reject}) = \varnothing$ אבל $<M_{reject}> \in DECIDE,\ <\widehat{M}> \notin DECIDE$.

מתקיים: $ALL \leq_m DECIDE$

$R(<M>) = <M'>$

$M'(x)$:

1. נריץ את M על x, אם M עוצרת ודוחה נריץ לולאה אינסופית (אם מקבלת – נקבל)

מתקיים: מ"ט R עוצרת על כל קלט.

- אם $<M> \in ALL \Leftarrow$ M מקבלת כל קלט $\Leftarrow$ M' עוצרת ומקבלת כל קלט$\Leftarrow$
$R(x) = <M'> \in DECIDE$.
- אם $<M> \notin ALL \Leftarrow$ קיים קלט w ש- M לא מקבלת $\Leftarrow$ M' לא עוצרת על w $\Leftarrow$
$R(x) = <M'> \notin DECIDE$

מסקנה: $DECIDE$ לא ניתנת לקבלה.

תרגיל 7 (מבחן מועד א' סמסטר א' 2004):

$L_7 = \{<M>:$ M מ"ט וקיימים ∞ קלטים עליהם היא עוצרת ו- ∞ קלטים עליהם היא לא עוצרת $\}$

האם שפה כריעה? ניתנת לקבלה?

פתרון:

מתקיים: $NOT-ACCEPT \leq_m L_7$

$R(<M,w>) = <M'>$

$M'(x)$:

1. אם קלט x מתחיל ב-w כלומר, מהצורה $x = w \circ y$ נריץ את M על w:
   - אם דוחה – נריץ לולאה אינסופית
   - אם מקבלת - נקבל
2. אחרת, נקבל

מתקיים: מ"ט R עוצרת על כל קלט.

- אם $NOT-ACCEPT \Leftarrow <M,w> \in NOT-ACCEPT$ ... אם $<M,w> \in NOT-ACCEPT \Leftarrow$ M לא מקבלת $w \Leftarrow$ M' לא עוצרת על כל קלט שמתחיל ב-w ויש ∞ כאלה. בנוסף, יש ∞ קלטים (שלא מתחילים ב- w) שאותם היא מקבלת ועוצרת $\Leftarrow$ $R(x) = <M'> \in L_7$.

- אם $<M,w> \notin NOT-ACCEPT \Leftarrow$ M מקבלת $w \Leftarrow$ M' מקבלת כל קלט $\Leftarrow$ $R(x) = <M'> \notin L_7$

מסקנה: L_7 לא ניתנת לקבלה.

האם $\overline{L_7}$ ניתנת לקבלה?

מתקיים: $NOT-ACCEPT \leq_m \overline{L_7}$

$R(<M,w>) = <M'>$

$M'(x)$:

1. אם קלט x מתחיל ב-w כלומר, מהצורה $x = w \circ y$ נריץ את M על w:
   - אם דוחה – נריץ לולאה אינסופית
   - אם מקבלת - נקבל
2. אחרת, נריץ לולאה אינסופית

מתקיים: מ"ט R עוצרת על כל קלט.

- אם $<M,w> \in NOT-ACCEPT \Leftarrow$ M לא מקבלת $w \Leftarrow$ M' לא עוצרת על כל קלט $\Leftarrow$ $R(x) = <M'> \in \overline{L_7}$.
- אם $<M,w> \notin NOT-ACCEPT \Leftarrow$ M מקבלת $w \Leftarrow$ M' מקבלת (ועוצרת) כל קלט שמתחיל ב-w ולא עוצרת על כל קלט שלא מתחיל ב-$w \Leftarrow$ $R(x) = <M'> \notin \overline{L_7}$

מסקנה: $\overline{L_7}$ לא ניתנת לקבלה.

תרגיל 8: $\{G$ דח"ה, $L(G)$ רגולרית $REGULAR_{CFG} = \{<G>:$
האם שפה כריעה? ניתנת לקבלה?

פתרון:

מתקיים: $ALL_{CFG} \leq_m REGULAR_{CFG}$

$R(<G>) = <G'>$

בהינתן דקדוק G כאשר S_G נונטרמינל התחלתי שלו נגדיר G' באופן הבא:

$$S_{G'} \to S_1 \# S_2 \mid S_2 \# S_G$$
$$S_1 \to aS_1b \mid \varepsilon$$
$$S_2 \to S_2S_2 \mid \varepsilon$$
$$S_2 \to \alpha \quad \forall \alpha \in \Sigma$$

מ"ט R עוצרת על כל קלט.

מתקיים: $L(G') = \{a^ib^i : i \geq 0\} \circ \{\#\} \circ \Sigma^* \cup \Sigma^* \circ \{\#\} \circ L(G)$

- אם $L(G) = \Sigma^* \Leftarrow <G> \in ALL_{CFG}$
$\Leftarrow L(G') = \{a^ib^i : i \geq 0\} \circ \{\#\} \circ \Sigma^* \cup \Sigma^* \circ \{\#\} \circ L(G) = \Sigma^* \circ \{\#\} \circ \Sigma^*$ $\Leftarrow$ שפה רגולרית
$\Leftarrow R(x) = <G'> \in REGULAR_{CFG}$

- אם $<G> \notin ALL_{CFG}$ אזי קיימת מילה $w \notin L(G)$. נניח בשלילה כי שפה של $L(G')$ רגולרית. לפי למת הניפוח לשפות רגולריות קיים מספר n_0 כך שתנאי הלמה מתקיימים...
נבחר מילה $a^{n_0}b^{n_0}\#w \in L(G')$ מתקיים $\left|a^{n_0}b^{n_0}\#w\right| \geq 2n_0 + 1$ לכן קיימת קטע ניפוח y לא ריק שמכיל רק a-ים. אבל מתקיים $a^{n_0+|y|}b^{n_0}\#w \notin L(G')$ בסתירה לנכונות של למת הניפוח לשפות רגולריות.

מסקנה: $R(x) = <G'> \notin REGULAR_{CFG}$

תרגיל 9: $L_9 = \{<M,k>:$ M מ"ט וקיים קלט ש- M עוצרת עליו לאחר k צעדי חישוב לכל היותר$\}$
האם שפה כריעה? ניתנת לקבלה?

פתרון:

התנהגות של M, שעוצרת לאחר k צעדי חישוב לכל היותר, על קלטים באורך יותר מ- k זהה להנהגות שלה על רישא בגודל k. לכן, כדי לדעת האם קיים קלט ש- M עוצרת עליו לאחר k צעדי חישוב לכל היותר מספיק להריץ M לאורך k צעדי חישוב רק על מילים באורך קטן שווה k (ויש רק מספר סופי של מלים כאלה).

$T(<M>)$:

1. $i = 0,1,2,\ldots,k$
   - נריץ M על כל המילים באורך i למשך k צעדי חישוב. אם מילה התקבלה ע"י M - נקבל
2. נדחה

מ"ט T מכריעה את השפה L_9.

האם שפה $\{<M>:$ M מ"ט, קיים מספר k וקלט ש- M עוצרת עליו לאחר k צעדי חישוב לכל היותר$\}$
האם שפה כריעה?

שפה זו שווה לשפה $\{<M>:$ M מ"ט וקיים קלט ש- M עוצרת עליו$\}$ ומתקיים:

$ACCEPT \leq_m \{<M>:$ M מ"ט וקיים קלט ש- M עוצרת עליו$\}$

$R(<M,w>) = <M'>$

$M'(x)$:

1. נריץ את M על w, אם M עוצרת ודוחה נריץ לולאה אינסופית (אם מקבלת – נקבל)

מ"ט R עוצרת על כל קלט.

אם $<M,w> \in ACCEPT \Leftarrow$ M מקבלת w $\Leftarrow$ M' עוצרת על כל קלט $\Leftarrow <M'>$ בשפה.

אם $<M,w> \notin ACCEPT \Leftarrow$ M לא מקבלת w $\Leftarrow$ M' לא עוצרת על כל קלט $\Leftarrow <M'>$ לא בשפה.

תרגיל 10 (מבחן מועד ב' סמסטר א' 2004):
האם קיימת מ"ט T שבהינתן קלט $<M>$ מדפיסה על הסרט כל הקידודים של מ"ט $<M'>,<M''>,<M'''>,\ldots$ כך שמתקיים: $L(M)=L(M')=L(M'')=L(M''')=\ldots$.

כלומר, אם מתקיים $L(M)=L(\widehat{M})$ בשלב מסוים T מדפיסה $<\widehat{M}>$ על הסרט.

פתרון:

אם T קיימת ניתן לבנות מ"ט שמקבלת את השפה $EQUAL$:
כדי לענות על השאלה "האם $<M_1,M_2>\in EQUAL$?" נריץ את T על $<M_1>$ אם בשלב מסוים $<M_2>$ מודפס על הסרט – נקבל.

נכונות:

- אם מתקיים $<M_1,M_2>\in EQUAL$ מובטח כי T על קלט $<M_1>$ בשלב מסוים תדפיס $<M_2>$ לכן נקבל לאחר מספר סופי של צעדים.

- אם מתקיים $<M_1,M_2>\notin EQUAL$ T על קלט $<M_1>$ לא מדפיסה $<M_2>$ לכן לא נעצור.

מסקנה: שפה $EQUAL$ לא ניתנת לקבלה, לכן T כפי שהוגדרה לא קיימת.

תרגיל 11:

$INFINITE=\{<M>:$ אינסופית $L(M)$,מ"ט $M\}$

הראו רדוקציות מיפוי $ALL\le_m INFINITE$ וגם $INFINITE\le_m ALL$.

פתרון:

$ALL\le_m INFINITE$

$R(<M>)=<M'>$

$M'(x)$:

- נריץ את M על $|x|+1$ מחרוזות ראשונות לפי סדר לקסיקוגרפי. אם כל מחרוזות התקבלו – נקבל.

R מ"ט שמבצעת רק פעולות על מחרוזות תמיד עוצרת.

מתקיים:

- $<M>\in ALL \Leftarrow$ M מקבלת כל מחרוזת $\Leftarrow$ M' מקבלת כל מחרוזת $\Leftarrow$ $L(M')$ אינסופית $\Leftarrow R(<M>)=<M'>\in INFINITE$
- $<M>\notin ALL \Leftarrow$ קיימת מחרוזת שלא מתקבלת ע"י M. נבחר מבין כל המחרוזות שלא מתקבלות ע"י M מחרוזת עם מיקום מינימאלי לפי סדר לקסיקוגרפי ונקרא למיקום זה $1\le min\_lex<\infty$. כלומר, כל מחרוזות שיש להן מיקום קטן מ- min_lex מתקבלות ומחרוזת במקום min_lex לא מתקבלת ע"י M $\Leftarrow$ כל מחרוזת קלט x שמקיימת $|x|+1<min\_lex$ מתקבלת ע"י M', וכל מחרוזת קלט x שמקיימת $|x|+1\ge min\_lex$ לא מתקבלת ע"י M' $\Leftarrow L(M')$ סופית.

$INFINITE\le_m ALL$

$R(<M>)=<M'>$

$M'(x)$:

1. $i\leftarrow 1$
2. נריץ את M על כל המחרוזות ממקום $|x|+1$ לפי סדר לקסיקוגרפי ועד למקום $|x|+i$ למשך i צעדי חישוב: אם קיימת מחרוזת שמתקבלת – נקבל
3. $i\leftarrow i+1$ וחוזרים לשלב 2

R מ"ט שמבצעת רק פעולות על מחרוזות תמיד עוצרת.

- $<M>\in INFINITE \Leftarrow$ בהינתן קלט x, קיימת מחרוזת שמתקבלת ע"י M ומיקומה לפי סדר לקסיקוגרפי גדול שווה $|x|+1$ (אם לא קיימת – סתירה לכך ש- $L(M)$ אינסופית). לכן על קלט x M' מקבלת. זה נכון לכל x ולכן M' מקבלת כל קלט $\Leftarrow L(M')=\Sigma^* \Leftarrow <M'>\in ALL$
- $<M>\notin INFINITE \Leftarrow$ M מקבלת מספר סופי של מחרוזות. נבחר מבין כל המחרוזות שמתקבלות ע"י M מחרוזת עם מיקום מכסימלי לפי סדר לקסיקוגרפי ונקרא למיקום זה $0\le max\_lex<\infty$ (משמעות של $max\_lex=0$ שאין מחרוזת ש- M מקבלת). כלומר, כל מחרוזות שיש להן מיקום גדול מ- min_lex לא מתקבלות $\Leftarrow$ לכן כל x שמקיים $|x|+1>max\_lex$ לא מתקבל ע"י M' (על x כזה M' מריצה M רק על מחרוזות ש- M לא מקבלת) $\Leftarrow L(M')\ne\Sigma^* \Leftarrow <M'>\notin ALL$

תרגיל 12:

$INFINITE_{DFA} = \{<A>:$ אינסופית $L(A)$,אס"ד $A\}$

האם שפה כריעה? ניתנת לקבלה?

פתרון:

ראינו שמתקיים: שפה של אס"ד $A = (Q, \Sigma, \delta, q_0, F)$ היא אינסופית $\Leftrightarrow$ קיימת מילה $w \in L(A)$ כך שמתקיים $|Q| \le |w| \le 2 \cdot |Q|$.

$T(<A>)$:

לכל מילה $w \in \Sigma^*$ שמקיימת $|Q| \le |w| \le 2 \cdot |Q|$ נבדוק האם $w \in L(A)$.

- אם קיימת מילה כזו – נקבל
- אחרת, נדחה

מ"ט T מכריעה את השפה.

תרגיל 13:

$INFINITE_{CFG} = \{<G>:$ אינסופית $L(G)$,דח"ה $G\}$

האם שפה כריעה? ניתנת לקבלה?

פתרון:

בהינתן $<G>$ האם ניתן לחשב מספר n_0 שקיים לפי למת הניפוח לשפות חסרות הקשר?
אם ניזכר בהוכחת של למת הניפוח, ראינו שאם נמצא (בעזרת אלגוריתם שלמדנו) דקדוק חסר הקשר $G' = (V', \Sigma, R', S')$ בצורה נורמאלית של חומסקי שמקיים $L(G) = L(G')$ ונגדיר $n_0 = 2^{|V'|}$ תנאי הלמה יתקיימו.

לכן, $L(G)$ אינסופית $\Leftrightarrow$ $\{w \in \Sigma^* : |w| \ge 2^{|V'|}\} \cap L(G) \ne \varnothing$.

- אם $\{w \in \Sigma^* : |w| \ge 2^{|V'|}\} \cap L(G) = \varnothing$ גודל של מילים ב-$L(G)$ חסום. לכן, שפה $L(G)$ סופית.
- אם $\{w \in \Sigma^* : |w| \ge 2^{|V'|}\} \cap L(G) \ne \varnothing$ קיימת ב-$L(G)$ מחרוזת ארוכה מספיק ולפי למת הניפוח ניתן לייצר ממנה אינסוף מחרוזות ב-$L(G)$. לכן, שפה $L(G)$ אינסופית.

$\{w \in \Sigma^* : |w| \ge 2^{|V'|}\} \cap L(G)$ שפה ח"ה כחיתוך של שפה רגולרית ושפה ח"ה. יתר על כן, בהינתן $<G>$ ניתן לבנות דקדוק חסר הקשר לשפה $\{w \in \Sigma^* : |w| \ge 2^{|V'|}\} \cap L(G)$ ולבדוק האם שפה שהוא מייצר היא ריקה.

מסקנה: שפה $INFINITE_{CFG}$ כריעה.

תרגיל 14 (מבחן מועד א' סמסטר ב' 2010): $L_{14}=\{<M_1,M_2>: <M_2>$ מקבלת M_1 ,מ"ט $M_2,M_1\}$

האם שפה כריעה? ניתנת לקבלה?

פתרון:

שפה ניתנת לקבלה: נריץ מ"ט על קלט $<M_2>$. אם מקבלת – נקבל.
שפה לא כריעה: $ACCEPT \leq_m L_{14}$

$R(<M,w>)=<M',M'>$

$M'(x)$:
1. נריץ את M על w ונפעל כמוה

R תמיד עוצרת.

- אם $<M',M'>\in L_{14} \Leftarrow <M'>\in L(M') \Leftarrow L(M')=\Sigma^* \Leftarrow <M,w>\in ACCEPT$
- אם $<M',M'>\notin L_{14} \Leftarrow <M'>\notin L(M') \Leftarrow L(M')=\varnothing \Leftarrow <M,w>\notin ACCEPT$

תרגיל 15 (מבחן מועד ב' סמסטר א' 2010):
$L_{15}=\{<M,w>:$ מ"ט, ותוך כדי עבודה על w , הראש של M בסרט עבודה שלה אי-פעם זז שמאלה $M\}$

האם שפה כריעה? ניתנת לקבלה?

פתרון:
אם נריץ את M על קלט w למשך $|w|+|Q|-2$ צעדי חישוב ($|Q|$- מספר מצבים של M).
1. אם שראש הכותב/קורא של M זז שמאלה – נדחה
2. אחרת, נקבל

אלגוריתם זה מכריע את השפה L_{15} .
הסבר: אם מ"ט M שרצה על קלט w למשך $|w|+|Q|-2$ צעדי חישוב לא עוצרת והראש כותב/קורא זז רק ימינה, אזי M לא עוצרת על קלט זה והראש כותב/קורא שלה זז רק ימינה.

אם M לא עוצרת אחרי $|w|+|Q|-2$ צעדים מתקיים:
לאחר $|w|$ צעדי חישוב הראש כותב/קורא נמצא באזור של הסרט שעליו כתובים רק רווחים.
במהלך $|Q|-2$ צעדים הבאים (כשאות הקלט תמיד רווח) M עוברת ב-$|Q|-1$ מצבים. אבל קיימים $|Q|-2$ מצבים שונים (q_{accept},q_{reject} לא מופעים כי M לא עצרה). לפי עקרון שובך היונים , קיים מצב שחוזר פעמיים. לכן, M נכנסת ללולאה אינסופית והראש זז רק ימינה.

תרגיל 16 (מבחן מועד א' סמסטר א' 2010):
מכונת טיורינג המקבלת קלטים אונריים , תקרא "מונוטונית" אם מספר צעדי חישוב שהיא מבצעת על הקלט 1^n הינה פונקציה מונוטונית לא יורדת ב-n.
לדוגמה, אם מתקיים:
$S_M(1)=5,\ S_M(11)=90,\ S_M(111)=90, S_M(1^n)=\infty \quad n>3$
כאשר S_M - מספר צעדי חישוב ש- M מבצעת על קלט.
אזי M מונוטונית.

$L_{16}=\{<M>:$ מ"ט מונוטונית $M\}$
האם שפה כריעה? ניתנת לקבלה?

פתרון:
$NOT-ACCEPT \le_m L_{16}$
$R(<M,w>)=<M'>$

$M'(x)$:
1. אם $x=1$ נריץ לולאה אינסופית
2. נריץ M על w
- אם דוחה – נריץ לולאה אינסופית
- אם מקבלת – נקבל

R תמיד עוצרת ומתקיים:
- אם $M' \Leftarrow <M,w> \in NOT-ACCEPT$ לא עוצרת על כל קלט ולכן מונוטונית $\Leftarrow <M'> \in L_{16}$.
- אם $M' \Leftarrow <M,w> \notin NOT-ACCEPT$ לא עוצרת על קלט $x=1$ ועוצרת על כל קלט אחר ולכן לא מונוטונית $\Leftarrow <M'> \notin L_{16}$.

מסקנה: L_{16} לא ניתנת לקבלה.

מבוא לסיבוכיות

בחלק זה של הקורס נתמקד בבעיות כריעות ע"י מכונת טיורינג. השאלה המרכזית - כמה משאבים (לדוגמה, זמן ריצה או זיכרון) דרושים למכונת טיורינג כדי להכריע את הבעיה.

תזכורת:

סימון אסימפטוטי:

$f\in O(g)$ - קיים קבוע c כך ש- $f(n)\le c\cdot g(n)$ לכל n גדול מספיק.

$f\in \Omega(g)$ - קיים קבוע c כך ש- $f(n)\ge c\cdot g(n)$ לכל n גדול מספיק.

$f\in \Theta(g)$ - קיימים קבועים c,c' כך ש- $c\cdot g(n)\ge f(n)\ge c'\cdot g(n)$ לכל n גדול מספיק.

$f\in o(g)$ - לכל $c>0$ $f(n)\le c\cdot g(n)$ לכל n גדול מספיק.

$f\in \omega(g)$ - לכל $c>0$ $f(n)\ge c\cdot g(n)$ לכל n גדול מספיק.

נוסחה בוליאנית בצורת CNF : פסוקיות מחוברות ע"י פעולת $\wedge$ - "וגם", כאשר כל פסוקית מורכבת מליטרלים מחוברים ע"י פעולת $\vee$ - "או".

לדוגמה, $(x_1\vee \overline{x}_2\vee x_3\vee \overline{x}_1)\wedge(x_5)\wedge(x_4\vee \overline{x}_5)$.

נוסחה בוליאנית בצורת $k-CNF$ כאשר היא בצורת CNF ובכל פסוקית יש בדיוק k ליטרלים.

זמן ריצה של מ"ט - פונקציה של אורך הקלט $t:\mathbb{N}\to\mathbb{N}$. כך שמתקיים: על כל קלט באורך n מתקבלת תשובה תוך $t(n)$ צעדי חישוב.

נאמר שמ"ט "יעילה" אם זמן ריצה שלה הוא פולינומי באורך הקלט. כלומר, $t\in O(n^c)$ כאשר c קבוע.

האם יעילות של מ"ט תלויה במספר סרטים שיש לה? כפי שראינו, לא.

בנוסף, אם זמן ריצה של מ"ט הוא $t(n)$, ניתן לבצע סימולציית ריצה שלה בעזרת מ"ט אוניברסאלית ב- $O(t(n)\log(t(n)))$.

הערה: יש חשיבות לאופן בו מקודד הקלט. לדוגמה, אלגוריתם שזמן ריצה שלו פולינומי באורך הקלט כאשר קלטים הם מספרים בקידוד אונארי, יכול להיות לא פולינומי אם קלטים מקודדים בקידוד בינארי.

הגדרות:

- $TIME(t(n))=\{L:$ ניתנת להכרעה ע"י מ"ט דטרמיניסטית שזמן ריצה שלה $\mathrm{O}(t(n))$ $L\}$
- $P=\bigcup_{c>0}TIME(n^c)=\{$ ניתנת להכרעה ע"י מ"ט דטרמיניסטית שזמן ריצה שלה פולינומיאלי $L\}$

נגדיר: שפה $2-SAT=\{\varphi:$ נוסחה ספיקה בצורת $2-CNF$ $\varphi\}$

נראה כי $2-SAT\in P$ (כלומר, קיימת מ"ט דטרמיניסטית פולינומית שמכריעה את השפה).

אלגוריתם:

קלט: φ נוסחה בצורת $2-CNF$. נניח כי מספר משתנים ב- φ הוא m: $x_1,x_2,\ldots,x_m$.

נבנה גרף מכוון G_φ עם $2m$ צמתים שנסמן ב- $x_1,x_2,\ldots,x_m,\overline{x}_1,\overline{x}_2,\ldots,\overline{x}_m$.

לכל פסוקית $(a\vee b)$ ב- φ נגדיר קשתות $(\overline{a},b)$ ו- $(\overline{b},a)$.

נבדוק האם קיים צומת x_i כך שקיים מסלול $x_i\xrightarrow{G_\varphi}\overline{x}_i$ וגם $\overline{x}_i\xrightarrow{G_\varphi}x_i$ (ניתן לבדוק בזמן פולינומי).

אם כן, "נוסחה φ לא ספיקה". אחרת, "נוסחה φ ספיקה".

הוכחת נכונות:

אלגוריתם תמיד עוצר וזמן ריצה שלו פולינומי באורך הקלט.

טענת עזר: אם קיים ב- G_φ מסלול $z_1\xrightarrow{G_\varphi}z_k$, אזי קיים מסלול $\overline{z}_k\xrightarrow{G_\varphi}\overline{z}_1$.

הוכחת טענה: אם קיים מסלול $z_1\to z_2\to\ldots\to z_k$ קיימות קשתות $(z_1,z_2),(z_2,z_3),\ldots,(z_{k-1},z_k)$. לפי בניית G_φ אם קיימת קשת (z_i,z_j) אזי קיימת גם קשת $(\overline{z}_j,\overline{z}_i)$. לכן קיימות ב- G_φ קשתות $(\overline{z}_k,\overline{z}_{k-1}),\ldots,(\overline{z}_3,\overline{z}_2),(\overline{z}_2,\overline{z}_1)$ כלומר, מסלול $\overline{z}_k\xrightarrow{G_\varphi}\overline{z}_1$.

טענה: נוסחה φ לא ספיקה $\Leftrightarrow$ קיים צומת x_i כך שקיים מסלול $x_i\xrightarrow{G_\varphi}\overline{x}_i$ וגם $\overline{x}_i\xrightarrow{G_\varphi}x_i$.

הוכחה:

($\Rightarrow$) נתון: קיימים $x_i \xrightarrow{G_\varphi} \overline{x}_i$, $\overline{x}_i \xrightarrow{G_\varphi} x_i$. נניח בשלילה כי קיימת השמה מספקת α לנוסחה φ.

- אם $\alpha(x_i)=T$ אזי $\alpha(\overline{x}_i)=F$

$$\begin{array}{ccccccc} x_i \rightarrow \ldots \rightarrow & x' & \rightarrow & x'' & \rightarrow \ldots & \overline{x}_i \\ T & T & & F & & F \end{array}$$

$\Leftarrow$ קיימת קשת (x',x'') ומתקיים: $\alpha(x')=T,\ \alpha(x'')=F$ $\Leftarrow$ קיימת פסוקית $(\overline{x}'\vee x'')$ ב- φ שערכה F תחת השמה α $\Leftarrow$ סתירה לכך ש- α השמה מספקת.

- אם $\alpha(x_i)=F$ אזי $\alpha(\overline{x}_i)=T$

$$\begin{array}{ccccccc} \overline{x}_i \rightarrow \ldots \rightarrow & y' & \rightarrow & y'' & \rightarrow \ldots & x_i \\ T & T & & F & & F \end{array}$$

$\Leftarrow$ קיימת קשת (y',y'') ומתקיים: $\alpha(y')=T,\ \alpha(y'')=F$ $\Leftarrow$ קיימת פסוקית $(\overline{y}'\vee y'')$ ב- φ שערכה F תחת השמה α $\Leftarrow$ סתירה לכך ש- α השמה מספקת.

מסקנה: φ לא ספיקה.

($\Leftarrow$) לכל צומת x_i אחד המסלולים $x_i \xrightarrow{G_\varphi} \overline{x}_i$ או $\overline{x}_i \xrightarrow{G_\varphi} x_i$ לא קיים. נוכיח קיום של השמה מספקת ע"י הצגת אלגוריתם שבונה אותה.

ניתן לכל צומת ב- G_φ ערך T/F.

כל עוד קיים צומת v שלא קיבל עדיין ערך T/F:

- אם לא קיים מסלול $v \xrightarrow{G_\varphi} \overline{v}$ ניתן ל- v ולכל צומת שניתן להגיע אליו מ- v ערך T.
- אחרת (לא קיים $\overline{v} \xrightarrow{G_\varphi} v$), ניתן ל- $\overline{v}$ ולכל צומת שניתן להגיע אליו מ- $\overline{v}$ ערך T.

הערה: כאשר צומת x מקבל ערך - צומת $\overline{x}$ מקבל מייד ערך נגדי ולהפך.

אלגוריתם זה תמיד עוצר. בנוסף, אלגוריתם הוא יעיל (אבל זה לא הכרחי להוכחה).

האם יתכן מצב בו צומת שכבר קיבל ערך בעבר, יקבל ערך שונה בשלבים הבאים ?

מקרה א':

בשלב מסוים צומת x' שלא היה לו ערך וקיים מסלול $x' \xrightarrow{G_\varphi} y$ מקבל ערך T. לכן, y מקבל ערך T.

בשלב מאוחר יותר צומת x'' שלא היה לו ערך ובנוסף קיים מסלול $\overline{y} \xrightarrow{G_\varphi} x''$ מקבל ערך T. לכן, y מקבל ערך F.

מתקיים: יש בגרף מסלול $y \xrightarrow{G_\varphi} x'$ ו- $\overline{y} \xrightarrow{G_\varphi} x''$. לפי טענת עזר אם קיים מסלול $\overline{y} \xrightarrow{G_\varphi} x''$ אזי קיים גם $\overline{x}'' \xrightarrow{G_\varphi} y$ $\Leftarrow$ קיים מסלול $\overline{x}'' \xrightarrow{G_\varphi} x'$ שעובר דרך y. לכן, x'' היה צריך לקבל ערך F בשלב בו x' קיבל ערך.

מסקנה: מצב שמתואר במקרה א' לא אפשרי.

באופן דומה אם y מקבל ערך F ובשלב מאוחר יותר ערך T.

מקרה ב':

צומת x' שלא היה לו ערך וקיימים $y \xrightarrow{G_\varphi} x'$ וגם $\overline{y} \xrightarrow{G_\varphi} x'$ מקבל ערך T. לפי טענת עזר קיים $\overline{x}' \xrightarrow{G_\varphi} y$. לכן, קיים $\overline{x}' \xrightarrow{G_\varphi} x'$ בסתירה לאופן בו נבחר x'.

השמה שאלגוריתם בונה היא השמה מספקת ל-φ:

לכל פסוקית $(a \vee b)$ קיימת קשת $(\overline{a}, b)$ ב-G_φ.

- אם ערך $\overline{a}$ הוא T, לפי בניית השמה ערך b הוא T (יש מסלול מ-$\overline{a}$ ל-b) $\Leftarrow$ ל-$(a \vee b)$ ערך T.
- אם ערך $\overline{a}$ הוא $F\Leftarrow$ ערך a הוא T $\Leftarrow$ ל-$(a \vee b)$ ערך T.

נכון לכל פסוקית, לכן ערך של φ תחת השמה שבנינו הוא T.

נגדיר: גרף לא מכוון $G = (V, E)$ הוא k-צביע אם קיימת פונקציה $c: V \to \{1, \ldots, k\}$ כך שלכל קשת $(u, v) \in E$ מתקיים $c(u) \neq c(v)$.

$2-COLOR = \{<G>: \text{ גרף } G \text{ 2-צביע}\}$

הערה: גרף 2-צביע $\Leftrightarrow$ גרף דו-צדדי.

מתקיים: $2-COLOR \in P$.

מכונת טיורינג לא דטרמיניסטית – מודל תיאורטי בו מאפשרים למכונה לבחור צעד חישוב הבא שלה מתוך קבוצה סופית של אפשרויות. קלט מתקבל ע"י מ"ט לא דטרמיניסטית אם קיים לפחות מסלול ריצה מקבל אחד.

זמן ריצה של מ"ט לא דטרמיניסטית– הוא פונקציה של אורך הקלט $t:\mathbb{N}\to\mathbb{N}$. כך שמתקיים: על כל קלט באורך n מתקבלת תשובה תוך $t(n)$ צעדי חישוב <u>בכל מסלולי חישוב</u>.

הגדרות:

- $NTIME(t(n))=\{L:$ מתקבלת ע"י מ"ט לא דטרמיניסטית שזמן ריצה שלה $O(t(n))$ $L\}$
- $NP=\bigcup_{c\geq 0} NTIME(n^c)=\{$ מתקבלת ע"י מ"ט לא דטרמיניסטית שזמן ריצה שלה פולינומיאלי $L\}$

הערה: $TIME(t(n))\subseteq NTIME(t(n))$ (ולכן, $P\subseteq NP$), $NTIME(t(n))\subseteq TIME(2^{t(n)})$.

מתקיים: $3-SAT,\ 3-COLOR\in NP$

מ"ט לא דטרמיניסטית לשפה $3-COLOR$:

1. בהינתן גרף, באופן לא דטרמיניסטי נרשום על הסרט צביעה של צמתים שלו ב-3 צבעים.
2. נבדוק שצביעה שהתקבלה היא צביעה חוקית של גרף הנתון ב-3 צבעים. אם כן, נקבל.

זמן ריצה של מכונה זו הוא פולינומי.
אם גרף 3-צביע, בשלב 1 ניתן לייצר צביעה שתתקבל בשלב 2. כלומר, קיים מסלול מקבל.
אם גרף לא 3-צביע, אזי כל צביעה לא תתקבל בשלב 2. כלומר, כל מסלול לא מקבל.

באופן דומה ניתן לבנות מ"ט לא דטרמיניסטית שמקבלת $3-SAT$:

1. בהינתן נוסחה, באופן לא דטרמיניסטי נרשום על הסרט השמה (כלומר, ערכי T/F למשתנים).
2. נבדוק שהשמה שהתקבלה מספקת את הנוסחה. אם כן, נקבל.

הגדרה: נאמר שלשפה יש **מוודא פולינומי** אם קיים קבוע c ומ"ט דטרמיניסטית ופולינומית V כך שמתקיים:

$$x\in L\Rightarrow \exists y: |y|<|x|^c\ \ V(x,y)=1$$
$$x\notin L\Rightarrow \forall y: V(x,y)=0$$

משפט: $L\in NP$ $\Leftrightarrow$ ל- L מוודא פולינומי.

תרגיל 1: אם $L_1, L_2 \in NP$ **,** אזי $L_1 \cup L_2 \in NP$.

פתרון:

אם שפה L_1 ב-NP, אזי קיים לה מוודא פולינומי. כלומר, מ"ט V_1 וקבוע c_1.

אם שפה L_2 ב-NP, אזי קיים לה מוודא פולינומי. כלומר, מ"ט V_2 וקבוע c_2.

נבנה מוודא פולינומי V ל-$L_1 \cup L_2$:

$V(x,y)$:

1. נריץ $V_1(x,y)$. אם פלט 1 - נחזיר 1.
2. נריץ $V_2(x,y)$. אם פלט 1 - נחזיר 1.
3. נחזיר 0.

מתקיים:

זמן ריצה של V פולינומי באורך הקלט.

$$x \in L \Rightarrow \exists y: |y| < |x|^{\max(c_1,c_2)} \; V(x,y)=1$$
$$x \notin L \Rightarrow \forall y: V(x,y)=0$$

הגדרה: יש **רדוקציה פולינומית** משפה A לשפה B (סימון $A \le_p B$) אם קיימת מ"ט דטרמיניסטית ופולינומית R ומתקיים $x \in A \Leftrightarrow R(x) \in B$ (כאשר משמעות של סימון $R(x)$ - פלט של R על x).

טענה 1: אם מתקיים $A \le_p B$ אזי אם $B \in P$ אזי $A \in P$. מסקנה: אם $A \notin P$ אזי $B \notin P$.

טענה 2: אם מתקיים $A \le_p B$ וגם $B \le_p C$ אזי $A \le_p C$.

הגדרה : שפה L נקראת $\boldsymbol{NP}$ **שלמה** אם מתקיים:

1. $L \in NP$
2. לכל שפה $L' \in NP$ מתקיים $L' \le_p L$

מסקנה ממשפט COOK-LEVIN: $3-SAT$ שפה NP שלמה.

טענה 3: אם שפה B ב-NP ומתקיים $A \le_p B$ כאשר A שפה NP שלמה, אזי B שפה NP שלמה.

תרגיל 3:

נגדיר:

$dHPATH=\{<G>:$ גרף מכוון ויש בו מסלול המילטוני $G\}$

$st-dHPATH=\{<G,s,t>:$ t ל- s גרף מכוון ויש בו מסלול המילטוני מ- $G\}$

הראו:

1. $dHPATH, st-dHPATH \in NP$
2. $dHPATH \le_p st-dHPATH,\ st-dHPATH \le_p dHPATH$

פתרון:

.2

$st-dHPATH \le_p dHPATH$

נגדיר: $R(<G,s,t>)=<G'>$ כאשר גרף G' מוגדר באופן הבא: נוסיף ל- G שני צמתים חדשים s',t' וקשתות מ- s' ל- s ומ- t ל- t'.

מתקיים: זמן ריצה של מ"ט R הוא פולינומי באורך הקלט.

- אם קיים מסלול המילטוני ב- G מ- s ל- t, אזי קיים מסלול המילטוני ב- G' (מ- s' ל- t').
- אם קיים מסלול המילטוני ב- G' הוא חייב להתחיל ב- s' ולהסתיים ב- t': $s',s,x_{i_1},\ldots,x_{i_n},t,t'$. המסלול $s,x_{i_1},\ldots,x_{i_n},t$ הוא מסלול המילטוני ב- G מ- s ל- t.

$dHPATH \le_p st-dHPATH$

נגדיר: $R(<G>)=<G',s,t>$ כאשר גרף G' מוגדר באופן הבא: נוסיף ל- G שני צמתים חדשים s,t וקשתות מ- s לכל צומת ב- G ומכל צומת ב- G ל- t.

מתקיים: זמן ריצה של מ"ט R הוא פולינומי באורך הקלט.

- אם קיים מסלול המילטוני ב- G: $x_{i_1},\ldots,x_{i_n}$, אזי קיים מסלול המילטוני ב- G' מ- s ל- t: $s,x_{i_1},\ldots,x_{i_n},t$.
- אם קיים מסלול המילטוני ב- G' מ- s ל- t: $s,x_{j_1},\ldots,x_{j_n},t$, אזי $x_{j_1},\ldots,x_{j_n}$ מסלול המילטוני ב- G.

טענה 4: מתקיים $3-SAT \leq_p st-dHPATH$.

מסקנה: $dHPATH, st-dHPATH$ שפות NP שלמות.

תרגיל 4:

$3-SAT \leq_p 4-SAT$

$R(\varphi)=\varphi'$ כאשר φ' מתקבלת מנוסחה בצורת $3-CNF$ φ על ידי שכפול של ליטרל ראשון בכל פסוקית. לדוגמה: $R((x_1 \vee \overline{x}_2 \vee x_3) \wedge (x_4 \vee \overline{x}_1 \vee \overline{x}_2)) = (x_1 \vee x_1 \vee \overline{x}_2 \vee x_3) \wedge (x_4 \vee x_4 \vee \overline{x}_1 \vee \overline{x}_2)$

מתקיים: השמה α מספקת φ $\Leftrightarrow$ השמה α מספקת $R(\varphi)=\varphi'$

מסקנה: לכל שפה $L' \in NP$ מתקיים: $L' \leq_p 4-SAT$.

$4-SAT \leq_p 3-SAT$

$R(\varphi)=\varphi'$ כאשר φ' מתקבלת מנוסחה בצורת $4-CNF$ φ באופן הבא: אם פסוקית מספר i ב-φ היא $(l_1 \vee l_2 \vee l_3 \vee l_4)$ נוסיף משתנה חדש a_i ונגדיר ב-φ': $(l_1 \vee l_2 \vee a_i) \wedge (l_3 \vee l_4 \vee \overline{a}_i)$.

לדוגמה:

$$R((x_1 \vee x_1 \vee \overline{x}_2 \vee x_3) \wedge (x_4 \vee x_4 \vee \overline{x}_1 \vee \overline{x}_2)) = (x_1 \vee x_1 \vee a_1) \wedge (\overline{x}_2 \vee x_3 \vee \overline{a}_1) \wedge (x_4 \vee x_4 \vee a_2) \wedge (\overline{x}_1 \vee \overline{x}_2 \vee \overline{a}_2)$$

- אם קיימת השמה מספקת α ל-φ, נגדיר השמה מספקת α' ל-φ' באופן הבא:
  $\alpha=\alpha'$ על משתנים משותפים לשתי נוסחאות וערך של משתנים a_i שהוספנו:
  $\alpha'(a_i)=T$ רק אם שני ליטרלים ראשונים בפסוקית מספר i קיבלו ערך F.
- אם קיימת השמה מספקת ל-φ' אזי היא גם השמה מספקת ל-φ:
  מתקיים: $\underbrace{(l_1 \vee l_2 \vee a_i)}_{T} \wedge \underbrace{(l_3 \vee l_4 \vee \overline{a}_i)}_{T}$.

  אם ערך של l_1 או l_2 הוא T, אזי ערך של $(l_1 \vee l_2 \vee l_3 \vee l_4)$ הוא T.
  אם ערך של l_1 וגם l_2 הוא F, אזי ערך של l_4 או l_3 הוא T. לכן, ערך של $(l_1 \vee l_2 \vee l_3 \vee l_4)$ הוא T.

מסקנה: $4-SAT \in NP$

ולכן $4-SAT$ שפה NP שלמה.

תרגיל 5:

נגדיר גרף "כמעט k-צביע" אם קיימת צביעה של צמתים שלו ב-k צבעים כך שלכל היותר קשת אחת מחברת בין שני צמתים שקיבלו אותו צבע.

הערה: גרף k-צביע הוא גם כמעט k-צביע.

נגדיר:

$k-COLOR=\{<G>:$ גרף G k-צביע$\}$

$k-COLOR'=\{<G>:$ גרף G כמעט k-צביע$\}$

מתקיים: $k-COLOR'\in NP$ כי ניתן לבדוק ביעילות האם צביעה נתונה של הגרף היא "כמעט k-צביעה" שלו.

$k-COLOR\leq_p k-COLOR'$

$R(<G>)=<G'>$ כאשר G' מתקבל מגרף G ע"י הוספת גרף שלם על $k+1$ צמתים (חדשים) כרכיב קשירות נוסף.

מתקיים: זמן ריצה של R פולינומי.

- אם G k-צביע, אזי G' כמעט k-צביע: קיימת צביעה של G ב-k צבעים ואת הגרף השלם על $k+1$ צמתים ניתן לצבוע ב-k צבעים כך שלכל היותר קשת אחת מחברת בין שני צמתים שקיבלו אותו צבע.
- אם G' כמעט k-צביע, אזי לפי הגדרה קיימת צביעה של צמתים שלו ב-k צבעים כך שלכל היותר קשת אחת מחברת בין שני צמתים שקיבלו אותו צבע. גרף שלם על $k+1$ צמתים הוא לא k-צביע. לכן, הקשת היחידה שמחברת בין שני צמתים שקיבלו אותו צבע שייכת לו. ל-G קיימת צביעה חוקית ב-k צבעים.

מסקנה: $k-COLOR'$ שפה NP שלמה.

תרגיל 6: נגדיר שפה

$3-SAT' = \{\varphi :$ <u>שבכל פסוקית</u> וקיימת לה השמה כך $3-CNF$ נוסחה בצורת $\varphi\}$
יש ליטרל שקיבל ערך T וליטרל שקיבל ערך F

הערה: $3-SAT' \subset 3-SAT$

מתקיים: $3-SAT' \in NP$

$3-SAT \leq_p 3-SAT'$

$R(\varphi) = \varphi'$ כאשר φ' מתקבלת מנוסחה בצורה $3-CNF$ φ באופן הבא: נגדיר משתנים חדשים $b, a_1, a_2, \ldots$. אם פסוקית מספר i ב-φ היא $(l_1 \vee l_2 \vee l_3)$. נוסיף משתנה חדש a_i ונגדיר ב-φ':
$(l_1 \vee l_2 \vee a_i) \wedge (l_3 \vee b \vee \overline{a}_i)$.

מתקיים: זמן ריצה של R פולינומי.

- אם קיימת השמה מספקת ל-φ ניתן להגדיר השמה ל-φ' כך שבכל פסוקית יש ליטרל שקיבל ערך T וליטרל שקיבל ערך F ע"י קביעת ערך F ל-b ול-a_i ערך T רק אם שני ליטרלים ראשונים בפסוקית מספר i קבלו ערך F.
- אם קיימת השמה α ל-φ' כך שבכל פסוקית יש ליטרל שקיבל ערך T וליטרל שקיבל ערך F, אזי השמה זו מספקת φ.

מקרה א': $\alpha(b) = F$

- אם $\alpha(l_3) = T$ אזי פסוקית מספר i $(l_1 \vee l_2 \vee l_3)$ ב-φ מקבלת ערך T.

- אם $\alpha(l_3) = F \Leftarrow \alpha(a_i) = F \Leftarrow$ l_1 או l_2 צרכים לקבל ערך T, אזי פסוקית מספר i $(l_1 \vee l_2 \vee l_3)$ ב-φ מקבלת ערך T.

מקרה ב': $\alpha(b) = T$. נבנה השמה $\overline{\alpha}$ ע"י הפיכת כל ערך בהשמה α. מתקיים: תחת השמה $\overline{\alpha}$ בכל פסוקית של φ' יש ליטרל שקיבל ערך T וליטרל שקיבל ערך F ובנוסף $\overline{\alpha}(b) = F$. לכן ניתן להפעיל מקרה א'.