אוניברסיטת חיפה
החוג למדעי המחשב

מודלים חישוביים

סיכומי ההרצאות של ד"ר מיכל דורי

נכתב על ידי בר וייסמן

סמסטר אביב תשפ"ג

תוכן העניינים

מבוא

◁ "הכרת המודלים החישוביים הפורמליים העומדים ביסודות של מדעי המחשב, ואפיון יכולות ומגבלות החישוב שלהם" (מתוך הסילבוס)

- מה זה מחשב?
- אילו בעיות מחשבים יכולים לפתור?

נתמקד בבעיות חישוב: בהינתן קלט x, יש להוציא פלט $f(x)$ עבור איזושהי פונקציה f.

קלט	פלט
מילה w	האם w מסתיימת ב-$'1'$
מספרים a, b	$a \cdot b$
מילה w	האם w מתארת תכנית C תקינה
מילה w	האם w מתארת תכנית C תקינה ואין קלט שעבורו היא נכנסת ללולאה אינסופית

טבלה 1: מספר דוגמאות לבעיות חישוב

הגדרה. בעיית חישוב שהתשובה עליה היא כן/לא נקראת <u>בעיית הכרעה</u>.
הקלטים יהיו מילים / מחרוזות, ונרצה אלגוריתם שמכריע האם מילה שייכת לשפה.

<u>נתעניין ב-2 שאלות מרכזיות:</u>

1. אילו בעיות מחשבים (לא) יכולים לפתור?
2. אילו בעיות מחשבים (לא) יכולים לפתור ביעילות?

1 אלפביתים ושפות

הגדרה. קבוצה Σ לא ריקה וסופית שמכילה אותיות תיקרא <u>אלפבית סופי</u>.

בהינתן א"ב סופי ניתן לבנות ממנו מילים (מילה - רצף תווים סופי, יכולה להיות ריקה - ε).

דוגמה. מספר דוגמאות לא"ב סופיים.

$$\{0,1\} \qquad \{a,b,c\}$$
$$\{0,\dots,9\} \qquad \{\text{א},\dots,\text{ת}\}$$

הגדרה. בהינתן שתי מילים w_1, w_2, <u>השרשור</u> שלהן יסומן ב-$w_1 \circ w_2$.

הגדרה. קבוצת כל המילים מעל א"ב סופי Σ תסומן ב-Σ^*.

הגדרה. יהי Σ א"ב סופי. קבוצה $L \subseteq \Sigma^*$ תיקרא <u>שפה</u> (יכולה להיות אינסופית).

דוגמה. מספר דוגמאות לשפות.

$$L = \emptyset \qquad \Sigma = \{0,1\}, L = \{0, 01, 110\}$$
$$L = \left\{ w \in \{0,1\}^* \mid \text{ב-}w\text{ מספר זוגי של 1-ים} \right\} \qquad L = \{\varepsilon\}$$

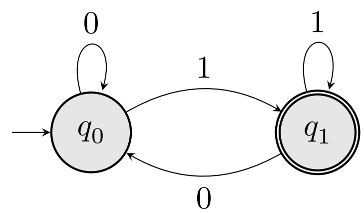
ניתן לחשוב על שפות כבעיות הכרעה: בהינתן שפה L מעל א"ב Σ, נגדיר את בעיית ההכרעה:

<u>קלט:</u> מילה $w \in \Sigma^*$.

<u>פלט:</u> האם $w \in L$.

חלק I
אוטומטים

1 אוטומט סופי דטרמיניסטי



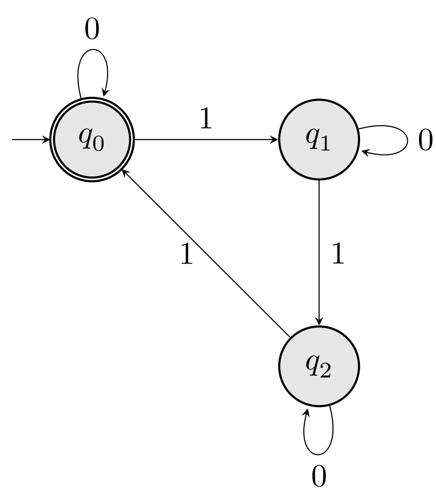
איור 1: אס"ד שמקבל את השפה $L = \left\{ w \in \{0,1\}^* \mid \text{מסתיימת ב-1 } w \right\}$.

- אס"ד מורכב ממצבים ומעברים ביניהם.
- חלק מהמצבים מסומנים ב-◎ (עיגול כפול) ונקראים <u>מצבים מקבלים</u>.
- מכל מצב יוצאים $|\Sigma|$ חצים, וכל חץ מסומן בתו אחד מ-Σ.

עבור כל מילה, נוכל להסתכל האם המצב בו נמצאים בסוף המילה הוא מקבל או לא, ובהתאם האם המילה התקבלה ע"י האוטומט.

- באס"ד לעיל, 001 תתקבל ו-10 תדחה.
- האוטומט מקבל את כל המילים שמסתיימות ב-1.

דוגמה. דוגמא נוספת לאס"ד.



איור 2: אס"ד שמקבל את השפה $L = \left\{ w \in \{0,1\}^* \mid \#_1 \in w \;\%\; 3 = 0 \right\}$

נסתכל על מספר מילים: 001 נדחית ו-010101 מתקבלת.
האוטומט מקבל את כל המילים שמספר ה-1ים בהן מתחלק ב-3.

$$L = \left\{ w \in \{0,1\}^* \mid \text{מספר ה-1ים ב-}w\text{ מתחלק ב-3} \right\}$$

הגדרה. <u>אוטומט סופי דטרמיניסטי</u> (אס"ד) הוא חמישייה $M = (Q, \Sigma, \delta, q_0, F)$.

- Q - קבוצה סופית של מצבים.
- Σ - א"ב סופי.
- q_0 - מצב התחלתי.
- $F \subseteq Q$ - קבוצת המצבים המקבלים.
- $\delta : Q \times \Sigma \to Q$ - פונקציית המעבר.

δ	0	1
q_0	q_0	q_1
q_1	q_1	q_2
q_2	q_2	q_0

טבלה 2: פונקציית המעברים עבור האס"ד מאיור 2

הגדרה. בהינתן אס"ד M ומילה w, <u>מסלול החישוב</u> של M על w הוא סדרת מצבים המוגדרת באופן הבא:

יהי $a_1, \ldots, a_n \in \Sigma$ כך ש-$w = a_1 \circ \ldots \circ a_n$.
נגדיר סדרה של $n+1$ מצבים $r_0, \ldots, r_n$ באופן הבא:

$$r_0 = q_0; \forall i : r_{i+1} = \delta\left(r_i, a_{i+1}\right)$$

המצב r_n יסומן על ידי $S_M(w)$.

דוגמה. נסתכל על המילה 0010, והאס"ד מאיור 2.

$$\begin{aligned} r_0 &= q_0 \\ r_1 &= \delta\left(r_0, 0\right) = q_0 \\ r_2 &= \delta\left(r_1, 0\right) = q_0 \\ r_3 &= \delta\left(r_2, 1\right) = q_1 \\ r_4 &= \delta\left(r_3, 0\right) = q_1 \end{aligned}$$

הגדרה. יהי M אס"ד. השפה של האוטומט תוגדר להיות: $L(M) = \{w \in \Sigma^* \mid S_M(w) \in F\}$.

הגדרה. נאמר כי שפה L מתקבלת ע"י אס"ד אם קיים אס"ד M כך ש-$L(M) = L$.

1.1 סגירויות

הגדרה. עבור שפה L, נגדיר את השפה המשלימה שלה להיות $\overline{L} = \Sigma^* \setminus L$.

משפט. אם שפה L מתקבלת ע"י אס"ד, אז $\overline{L}$ מתקבלת ע"י אס"ד. (סגירות שפות המתקבלות ע"י אס"ד תחת משלים).

הוכחה. יהי M אס"ד כך ש-$L = L(M)$, נרצה לבנות אס"ד M' כך ש-$L(M') = \overline{L(M)}$.
ניקח את M ונחליף בין המצבים המקבלים והלא-מקבלים.

<u>בנייה:</u> נגדיר $M' = (Q', \Sigma', \delta', q_0', F')$ באופן הבא

$$Q' = Q, \Sigma' = \Sigma, \delta' = \delta, q_0' = q_0, \boxed{F' = Q \setminus F}$$

נשים לב כי לכל $w \in \Sigma^*$ מתקיים $S_{M'}(w) = S_M(w)$, מאחר ולא שינינו את δ, q_0.

$$\begin{aligned} L(M') &= \{w \mid S_{M'}(w) \in F'\} \\ &= \{w \mid S_M(w) \in F'\} \\ &= \{w \mid S_M(w) \in Q \setminus F\} \\ &= \{w \mid S_M(w) \notin F\} \\ &= \overline{L(M)} \end{aligned}$$

$$\Longrightarrow \boxed{L(M') = \overline{L(M)}}$$

□

משפט. יהיו L_1, L_2 שפות המתקבלות ע"י אס"ד מעל א"ב Σ. אזי, $L_1 \cap L_2$ גם מתקבלת ע"י אס"ד. (סגירות תחת חיתוך).

נבנה אוטומט מכפלה + חיתוך.

הוכחה. יהיו M_1, M_2 אס"דים שמקבלים את השפות L_1, L_2 בהתאמה ($M_1 = (Q_1, \Sigma, \delta_1, q_0^1, F_1)$, $M_2 = (Q_2, \Sigma, \delta_2, q_0^2, F_2)$). נבנה אס"ד M כך ש-$L(M) = L(M_1) \cap L(M_2)$.

<u>בנייה:</u> נגדיר $M = (Q, \Sigma, \delta, q_0, F)$ באופן הבא:

$$Q = Q_1 \times Q_2 = \{(q_1, q_2) \mid q_1 \in Q_1 \wedge q_2 \in Q_2\}$$

$$q_0 = (q_0^1, q_0^2)$$

$$F = F_1 \times F_2 = \{(q_1, q_2) \mid q_1 \in F_1 \wedge q_2 \in F_2\}$$

$$\delta : Q \times \Sigma \to Q$$

$$\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$$

נראה כי $L\left(M\right)=L\left(M_{1}\right)\cap L\left(M_{2}\right)$.

למה. תהי מילה $w\in\Sigma^{*}$, אזי $S_{M}\left(w\right)=\left(S_{M_{1}}\left(w\right),S_{M_{2}}\left(w\right)\right)$.

הוכחה. נוכיח את הלמה באינדוקציה על $|w|$.
בסיס האינדוקציה: כאשר $|w|=0$, כלומר $w=\varepsilon$.

$$S_{M}\left(\varepsilon\right)=q_{0}=\left(q_{0}^{1},q_{0}^{2}\right)=\left(S_{M_{1}}\left(\varepsilon\right),S_{M_{2}}\left(\varepsilon\right)\right)$$

צעד האינדוקציה: נניח כי הטענה נכונה לכל מילה u מאורך n, נוכיח את נכונות הטענה לכל מילה w מאורך $n+1$. יהיו $a\in\Sigma$ ו-$u\in\Sigma^{*}$ כך ש-$w=u\circ a$. לפי הנחת האינדוקציה:

$$S_{M}\left(u\right)=\left(S_{M_{1}}\left(u\right),S_{M_{2}}\left(u\right)\right)$$

$$\begin{aligned}S_{M}\left(w\right)&=\delta\left(S_{M}\left(u\right),a\right)\\&=\delta\left(\left(S_{M_{1}}\left(u\right),S_{M_{2}}\left(u\right)\right),a\right)\\&=\left(\delta_{1}\left(S_{M_{1}}\left(u\right),a\right),\delta_{2}\left(S_{M_{2}}\left(u\right),a\right)\right)\\&=\left(S_{M_{1}}\left(w\right),S_{M_{2}}\left(w\right)\right)\end{aligned}$$

$$\Longrightarrow\boxed{S_{M}\left(w\right)=\left(S_{M_{1}}\left(w\right),S_{M_{2}}\left(w\right)\right)}$$

והוכחנו את צעד האינדוקציה. □

נשתמש בלמה לסיום ההוכחה:

$$\begin{aligned}L\left(M\right)&=\left\{w\in\Sigma^{*}\mid S_{M}\left(w\right)\in F\right\}\\&=\left\{w\in\Sigma^{*}\mid\left(S_{M_{1}}\left(w\right),S_{M_{2}}\left(w\right)\right)\in F\right\}\\&=\left\{w\in\Sigma^{*}\mid S_{M_{1}}\left(w\right)\in F_{1}\wedge S_{M_{2}}\left(w\right)\in F_{2}\right\}\\&=\left\{w\in\Sigma^{*}\mid S_{M_{1}}\left(w\right)\in F_{1}\right\}\cap\left\{w\in\Sigma^{*}\mid S_{M_{2}}\left(w\right)\in F_{2}\right\}\\&=L\left(M_{1}\right)\cap L\left(M_{2}\right)\end{aligned}$$

$$\Longrightarrow\boxed{L\left(M\right)=L\left(M_{1}\right)\cap L\left(M_{2}\right)}$$

□

משפט. אם L_{1},L_{2} שפות המתקבלות ע"י אס"ד, גם שפת האיחוד $L_{1}\cup L_{2}$ מתקבלת ע"י אס"ד.

הערה. נוכל להוכיח את המשפט באופן דומה להוכחת הסגירות תחת חיתוך, רק ש:

$$F=\left\{\left(q_{1},q_{2}\right)\mid q_{1}\in F_{1}\vee q_{2}\in F_{2}\right\}$$

אך לא נעשה זאת.

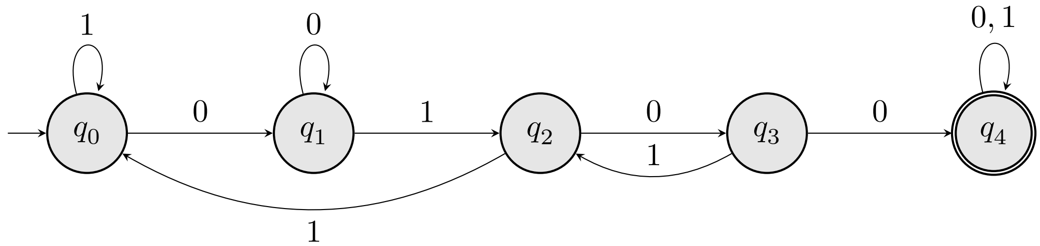
הוכחה. נבצע רדוקציה לסגירות תחת משלים וחיתוך.

$$L_1 \cup L_2 = \overline{\overline{L_1} \cap \overline{L_2}}$$

מאחר וקיימת סגירות תחת משלים וסגירות, השפה $\overline{\overline{L_1} \cap \overline{L_2}}$ מתקבלת ע"י אס"ד ולכן גם שפת האיחוד. □

ניתן להראות באופן דומה כי $L_1 \setminus L_2$ מתקבלת ע"י אס"ד: $L_1 \setminus L_2 = L_1 \cap \overline{L_2}$.

דוגמה. נסתכל על $\Sigma = \{0,1\}$ והשפה $L = \{w | 0100$ מכילה מופע של w$\}$.



איור 3: אסד עבור השפה L.
כל מצב מסמל איזו רישא של 0100 ראינו עד כה.

הגדרה. יהיו L_1, L_2 שפות מעל א"ב Σ_1, Σ_2 בהתאמה. שפת השרשור $L_1 \circ L_2$ מוגדרת להיות:

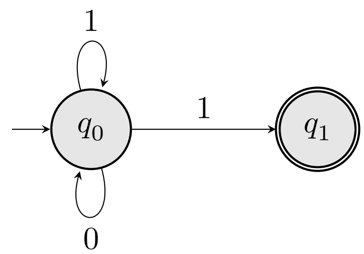
$$L_1 \circ L_2 = \{w = w_1 \circ w_2 \mid w_1 \in \Sigma_1 \wedge w_2 \in \Sigma_2\}$$

משפט. אם השפות L_1, L_2 מתקבלות ע"י אס"ד, גם $L_1 \circ L_2$ מתקבלת ע"י אס"ד.

לשם הוכחת משפט זה, נדבר על אוטומט (או חישוב) אי-דטרמיניסטי.

2 אוטומט סופי לא דטרמיניסטי

דוגמה. דוגמא לאוטומט אי-דטרמיניסטי.



איור 4: אוטומט לא דטרמיניסטי.

<u>מה מוזר באוטומט הזה?</u>

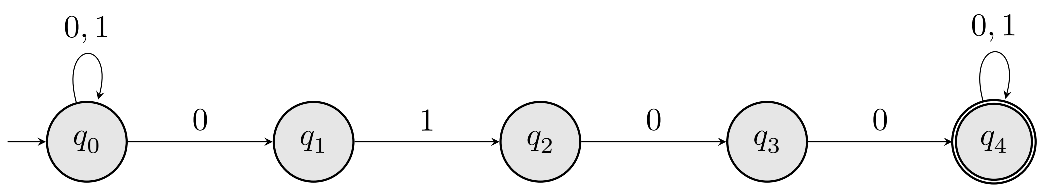
- שתי קשתות עם "1" שיוצאות מאותו המצב.
- מצב ללא קשתות יוצאות.

בחישוב לא-דטרמיניסטי יכולות להיות מספר אפשרויות מכל מצב ותו, ומסלולי חישוב שונים לאותה המילה.

הגדרה. יהי M אוטומט סופי לא דטרמינסטי (אסל״ד).
השפה של M מוגדרת להיות אוסף כל המילים שקיים מסלול חישוב של האוטומט על w שמסיים לקרוא את w ומגיע למצב מקבל.

בדוגמא לעיל, שפת האסל״ד היא אוסף כל המילים שמסתיימות ב-1 (הוכחה ע״י הכלה דו כיוונית).

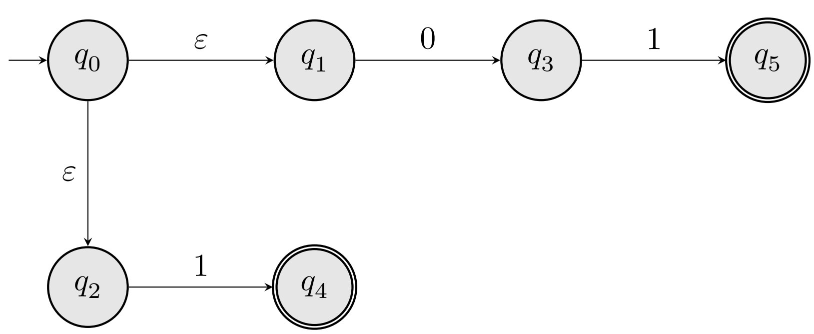
דוגמה. לעיתים יותר קל לצייר אסל״ד מאשר אס״ד.



איור 5: אסל״ד שמקבל את שפת כל המילים שמכילות 0100.

האס״ד שמתאר את אותה השפה מסובך יותר, ונוח לתאר את השפה באמצעות אסל״ד זה.

דוגמה. אסל״ד המכיל מעברי ε, בהם אפשר לעבור ואפשר לא לעבור.



איור 6: אסל״ד המכיל מעברי ε.

הגדרה. אסל"ד הוא חמישייה $M=(Q,\Sigma,\delta,q_0,F)$ כך ש:

- Q - קבוצה סופית של מצבים.
- Σ - א"ב סופי.
- $q_0 \in Q$ - מצב התחלתי.
- F - קבוצת המצבים המקבלים.
- $\delta : Q \times \Sigma \to P\left(Q\right)$ - פונקציית המעברים (δ מחזירה קבוצת מצבים, ולא מצב בודד כמו באס"ד).

למשל, $\delta\left(q,a\right)=\{q_7,q_{18}\}$, $\delta\left(q,a\right)=\emptyset$ או $\delta\left(q,a\right)=\{q_7\}$.
נותר להגדיר את מעברי ה-ε. לשם כך, נגדיר $\Sigma_\varepsilon=\Sigma\cup\{\varepsilon\}$.

- אם $\delta\left(q,\varepsilon\right)=\emptyset$ משמע שאין מעברי ε מהמצב q.

מהי השפה של אסל"ד?

- אין יותר "מסלול החישוב" - יש כמה מסלולי חישוב שונים.

הגדרה. יהיו M אסל"ד ו-$w\in\Sigma^*$ מילה.
נאמר כי סדרת מצבים $r_0,\dots,r_n$ היא מסלול חישוב של M על w אם קיימים $a_1,\dots,a_n$ כך ש-$w=a_1\circ\dots\circ a_n$, לכל i מתקיים $a_i\in\Sigma_\varepsilon$ וגם:

$$r_0=q_0, \forall i : r_{i+1}\in\delta\left(r_i,a_{i+1}\right)$$

הגדרה. יהי M אוטומט ותהי w מילה. נגדיר את קבוצת כל המצבים שניתן להגיע אליהם אחרי שקוראים את w:

$$S_M\left(w\right)=\{q\in Q\mid \text{קיים מסלול חישוב של } M \text{ על } w \text{ שמסתיים ב-} q\}$$

הגדרה. יהי M אסל"ד. השפה של M, $L\left(M\right)$, היא:

$$\begin{aligned}L\left(M\right)&=\{w\mid S_M\left(w\right)\cap F\neq\emptyset\}\\&=\{w\mid \text{קיים מסלול חישוב של } M \text{ על } w \text{ שמסתיים במצב מקבל}\}\end{aligned}$$

- למה אסל"ד טוב?
  - אסל"ד מגדיר שפה (טוב לספציפקציה).
  - טוב כדי להוכיח שמשהו לא אפשרי.

שאלה: האם יש שפות שאסל"ד יכול לקבל ואס"ד לא יכול?

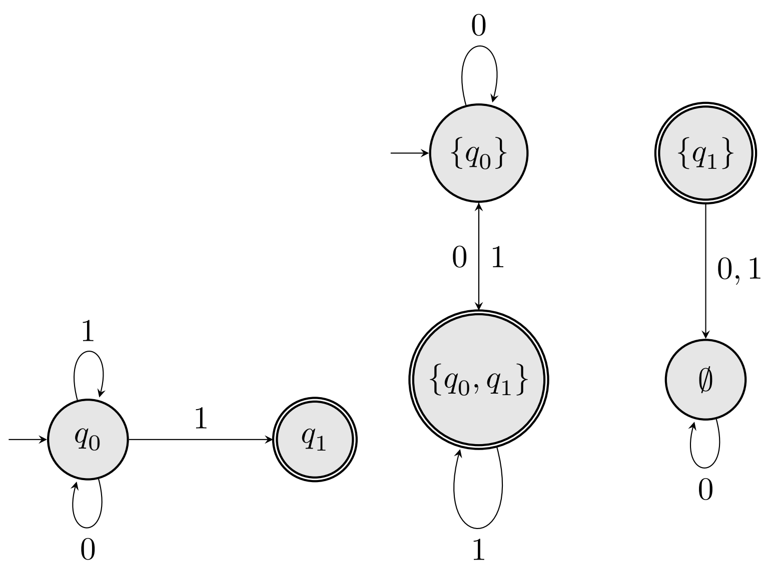
משפט. שפה מתקבלת ע"י אסל"ד אמ"מ היא מתקבלת ע"י אס"ד.

נרצה להוכיח את שני כיווני המשפט.

- תחילה, ברור כי כל שפה שמתקבלת ע"י אס"ד תתקבל ע"י אסל"ד.
- באופן פורמלי, האסל"ד יהיה זהה לחלוטין פרט ל-δ.
- נגדיר את $\delta'(q,a)=\{\delta(q,a)\}$, וכך השפה תתקבל ע"י אסל"ד.

כעת, נוכיח את הכיוון השני - נבנה אס"ד בהינתן אסל"ד.

דוגמה. בניית אס"ד בהינתן אסל"ד.



איור 7: משמאל אסל"ד ומימין אס"ד שמקבלים את שפת כל המילים שמסתיימות ב-1.

רעיון: מצב של אס"ד M' יהיה קבוצת מצבים של האסל"ד M. קבוצת המצבים של M' תהיה $Q'=P(Q)$, ומצביו יקראו סופר-מצבים.

ננסה לשמר את התכונה $S_{M'}(w)=S_M(w)$, לכל $w\in\Sigma^*$. כלומר, בכל שלב M' יחזיר בסופר-מצב הנוכחי את קבוצת המצבים בהם יכל M להגיע באותו השלב.

הוכחה. נוכיח (באמצעות בנייה) כי אם שפה מתקבלת ע"י אסל"ד היא מתקבלת ע"י אס"ד.
לצורך פשטות נניח כי ב-M אין מעברי ε.

$$Q'=P(Q),|Q'|=2^{|Q|}$$
$$q_0'=\{q_0\}$$
$$\delta'(S,a)=\bigcup_{q\in S}\delta(q,a)$$
$$F'=\{S\in Q'\mid S\cap F\neq\emptyset\}$$

נוכיח כי $L(M')=L(M)$. □

למה. לכל $w \in \Sigma^*$ מתקיים $S_{M'}(w) = S_M(w)$.

הוכחה. נוכיח את הלמה באינדוקציה על $|w|$.

- <u>בסיס:</u> $|w| = 0$, $w = \varepsilon$. מאחר ואין מעברי ε:

$$\begin{aligned} S_{M'}(\varepsilon) &= q'_0 = \{q_0\} \\ S_M(\varepsilon) &= q_0 \end{aligned}$$

- <u>צעד:</u> נניח את נכונות הטענה עבור מילים מאורך n, ונוכיח עבור מילים באורך $n+1$. $w = u \circ a$ כך ש-$n = |u|$.

$$\begin{aligned} S_{M'}(w) &= S_{M'}(u \circ a) \\ &= \delta\left(S_{M'}(u), a\right) \\ \text{(הנחת האינדוקציה)} &= \delta'\left(S_M(u), a\right) \\ \text{(הגדרת } \delta'\text{)} &= \bigcup_{q \in S_M(u)} \delta(q, a) \\ &= S_M(w) \end{aligned}$$

נכונות המעבר האחרון:

- $S_M(u)$ - קבוצת כל המצבים שניתן להגיע אליהם באסל"ד לאחר קריאת u.
- $\delta(q, a)$ - כל המצבים שפאשר להגיע אליהם מ-q אחרי שקראנו a.
- לכן, $\bigcup_{q \in S_M(u)} \delta(q, a)$ זו קבוצת כל המצבים שאפשר להגיע אליהם לאחר קריאת w, כלומר $S_M(w)$.

□

הוכחה. כעת, ניעזר בטענת העזר להוכחה:

$$\begin{aligned} L(M') &= \{w \mid S_{M'}(w) \in F'\} \\ \text{(טענת העזר + הגדרת } F'\text{)} &= \{w \in S_M(w) \cap F \neq \emptyset\} \\ \text{(הגדרת אסל"ד)} &= L(M) \end{aligned}$$

הוכחנו כי $L(M') = L(M)$, וכך כל שפה שמתקבלת ע"י אסל"ד מתקבלת ע"י אס"ד. □

הערה. מעברי ε.
ניתן להרחיב את ההוכחה למקרה שב-M יש מעברי ε.

$$\forall q \in Q : E(q) = \{q' \mid \text{ניתן להגיע מ-} q \text{ ל-} q' \text{ ע"י מעברי } \varepsilon \text{, כולל } q \text{ עצמו}\}$$

כעת, נגדיר את האוטומט בצורה הבאה:

$$q'_0 = E(q_0)$$

$$\forall S \subseteq Q : \quad \begin{aligned} E(S) &= \textstyle\bigcup_{q \in S} E(q) \\ \delta'(S, a) &= E\left(\textstyle\bigcup_{q \in S} \delta(q, a)\right) \end{aligned}$$

המשפט לעיל נותן לנו יכולת להוכיח שקיים אס"ד ע"י בניית אסל"ד (קל יותר).

משפט. יהיו L_1, L_2 שפות המתקבלות ע"י אס"דים. אזי, $L_1 \circ L_2$ מתקבלת ע"י אסל"ד (וכך גם ע"י אס"ד).

הוכחה. נבנה אסל"ד עבור שפת השרשור $L_1 \circ L_2$, בהינתן אס"דים M_1, M_2:
נחבר מעברי ε מכל מצב מקבל ב-M_1 ל-q_0^2, ונהפוך את המצבים המקבלים ב-M_1 ללא מקבלים.
המצבים המקבלים של אסל"ד השרשור הם המצבים המקבלים של M_2. □

<u>איך נכליל מעבר מאסל"ד לאס"ד עם מעברי ε?</u>

- המצב ההתחלתי יהיה קבוצה שמכילה את המצב ההתחלתי באס"ד, וכל מצב שאפשר להגיע אליו ממנו עם מעברי ε <u>בלבד</u>.

$$\delta(Q,\sigma) = \underbrace{\{\delta(q,\sigma) \mid q \in Q\}}_{Q'} \cup \{\delta(q,\varepsilon) \mid q \in Q'\}$$

משפטי עזר

משפט. אם קיים לשפה אסל"ד, קיים לה אסל"ד עם מצב מקבל יחיד.

הוכחה. נחבר מצב חדש - המקבל היחיד. מכל המצבים המקבלים הקודמים יהיו מעברי ε למעבר החדש, והם כבר לא יהיו מקבלים. □

משפט. אם שפה L מתקבלת ע"י אסל"ד, גם L^* מתקבלת ע"י אסל"ד.

הוכחה. נוסיף מצב מקבל - הוא יהיה המצב ההתחלתי החדש. נוסיף מעבר ε ממנו למצב ההתחלתי של האסל"ד של L. לבסוף, מהמצב המקבל של האסלד של L נוסיף מעבר ε למצב ההתחלתי של האסל"ד של L. □

3 ביטויים רגולריים

דוגמה. תחשיב הביטויים החשבוניים.

ביטוי חשבוני הוא מחרוזת מעל א"ב $\{0,\ldots,9,+,-,*,(,)\}$. למשל: $7+8-(3+5\cdot 2)$ הוא ביטוי חשבוני תקין.

סינטקס: תקינות של הביטוי כמחרוזת. למשל "$7(($" אינו ביטוי חשבוני תקין.

מהו ביטוי חשבוני תקין? משהו שאפשר לבנות מביטויים פשוטים ע"י פעולות הרכבה, אינדוקציה.

- "9", ... ,"0" הם ביטויים אטומים.
  - הגדרה אינדוקטיבית: בהינתן שני ביטויים חשבוניים תקינים α,β, גם "$\alpha+\beta$" הוא ביטוי תקין.
    * למשל, הביטויים התקינים $7+8$ ו-3, גם הביטוי $7+8+3$ הוא תקין.

סמנטיקה: מה משמעות $7+3$? משמעות: 10.

משמעות: $L(\text{"}0\text{"})=0$, $L(\text{"}\alpha+\beta\text{"})=L(\alpha)+L(\beta)$. גם את המשמעות מגדירים באופן אינדוקטיבי.

ביטוי רגולרי הוא מחרוזת.

משמעות: הביטוי הרגולרי מגדיר שפה.

דוגמה. ביטויים רגולריים מעל הא"ב $\Sigma=\{0,1\}$ שלא מכיל את התווים המיוחדים: $*,\cup,\emptyset,\varepsilon,(,),\circ$.

$$\left((0\cup 1)^*\circ 1\right)\Longrightarrow L=\{w\mid \text{מסתיימת ב-1 } w\}$$

$$\left(0\circ(0\cup 1)^*\circ 0\right)\Longrightarrow L=\{w\mid |w|\geq 2 \text{ מתחילה ונגמרת ב-0 וגם } w\}$$

הגדרה. ביטוי רגולרי.

בהינתן א"ב סופי Σ שלא מכיל את התווים המיוחדים $\varepsilon,\emptyset,\cup,*,\circ,(,)$, נגדיר באופן אינדוקטיבי:

- ביטויים רגולריים אטומיים הם המחרוזות הבאות:
  - התו a לכל $a\in\Sigma$. השפה שהוא מייצג היא השפה $\{a\}$.
  - הקבוצה הריקה $\emptyset$. השפה שהוא מייצג היא השפה הריקה $\{\}$.
  - ε הוא ביטוי רגולרי אטומי, שמייצג את $\{\varepsilon\}$.
- נגדיר ביטוי רגולרי באופן אינדוקטיבי: נניח כי α_1,α_2 הם ביטויים רגולריים שכבר הוגדרו. אזי:
  - $(\alpha_1\cup\alpha_2)$ הוא ביטוי רגולרי.
  - $(\alpha_1\circ\alpha_2)$ הוא ביטוי רגולרי.
  - (α_1^*) הוא ביטוי רגולרי.
- ואלה כל הביטויים הרגולריים.

דוגמה. מספר דוגמאות לביטויים רגולריים.

$$\emptyset \quad (\emptyset \cup 1) \quad \varepsilon \quad \varepsilon \circ 1 \quad \left(\left((0 \cup 1)^*\right) \circ 1\right)$$

<u>משמעות:</u> לכל ביטוי רגולרי α, נגדיר שפה $L(\alpha)$ מעל הא"ב Σ באופן אינדוקטיבי.

- מקרה בסיס - ביטויים רגולריים אטומיים:
  - לכל $a \in \Sigma$, $L(a) = \{a\}$.
  - $L(\emptyset) = \{\}$.
  - $L(\{\varepsilon\}) = \{\varepsilon\}$.
- מקרה כללי: בהינתן ביטוי רגולרי לא אטומי α, ייתכנו 3 מקרים (באינדוקציה, אנו מניחים כי $L(\alpha_1), L(\alpha_2)$ כבר הוגדרו):

α	$L(\alpha)$
$(\alpha_1 \cup \alpha_2)$	$L(\alpha_1) \cup L(\alpha_2)$
$(\alpha_1 \circ \alpha_2)$	$L(\alpha_1) \circ L(\alpha_2)$
$(\alpha_1)^*$	$L^*(\alpha_1)$

הערה. במידה ואין סוגריים, $*$ קודם ל-$\circ$ ושניהם קודמים ל-$\cup$.

דוגמה. מספר דוגמאות לייצוג שפות ע"י ביטויים רגולריים.

ביטוי רגולרי	שפה
$\left(((0 \cup 1))^* \circ 1\right)$	$\{0,1\}^* \circ \{1\}$
$1^* \circ 0 \circ 1^* \circ 0 \circ 1^*$	$\{w \mid \#_0 \in w = 2\}$
$(1^* \circ 0 \circ 1^* \circ 0 \circ 1^*)^* \circ 1^*$	$\{w \mid \#_0 \in w \equiv 0 \mod 2\}$

טבלה 3: מספר דוגמאות לייצוג שפות ע"י ביטויים רגולריים מעל הא"ב $\Sigma = \{0,1\}$

הגדרה. שפה L תיקרא <u>רגולרית</u> אם היא נוצרת ע"י ביטוי רגולרי.

3.1 רגולריות $\Longleftrightarrow$ אס"ד $\Longleftrightarrow$ אסל"ד

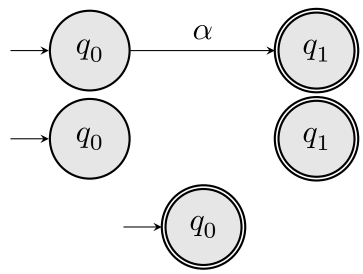
משפט. שלושת התנאים הבאים הם שקולים.

1. L מתקבלת ע"י אס"ד.
2. L מתקבלת ע"י אסל"ד.
3. L רגולרית.

טענה. כל שפה רגולרית מתקבלת ע"י אסל"ד (תנאי 3 $\Longleftarrow$ תנאי 2).

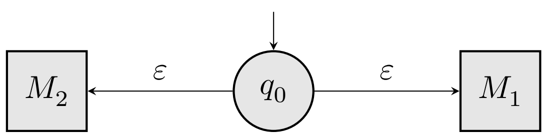
הוכחה. נוכיח את הטענה - בהינתן ביטוי רגולרי α, נבנה אסל"ד M כך ש-$L(M)=L(\alpha)$.
נבנה את האסל"ד באופן רקורסיבי.

- <u>בסיס:</u> α ביטוי רגולרי אטומי. נפריד למקרים:



איור 8: מלמעלה למטה: $\alpha\in\Sigma$ - $L(\alpha)=\{\alpha\}$; $\alpha=\emptyset$ - $L(\alpha)=\emptyset$; $\alpha=\varepsilon$ - $L(\alpha)=\varepsilon$.

- <u>צעד:</u> α ביטוי רגולרי שאינו אטומי (מורכב מ-α_1 ו-α_2). נניח כי קיימים אסל"דים M_1,M_2 כך ש-$L(M_1)=L(\alpha_1)\,,L(M_2)=L(\alpha_2)$, ונפריד למקרים:
  - $\alpha=\alpha_1\cup\alpha_2$, נובע מסגירות תחת איחוד כי $L(M)=L(M_1)\cup L(M_2)$. בנוסף, ניתן לבנות את האסל"ד באופן הבא.



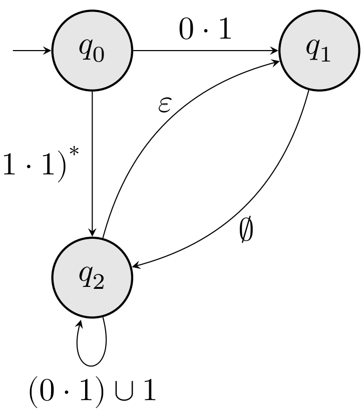
  - $\alpha=\alpha_1\circ\alpha_2$, הוכחנו סגירות תחת שרשור ולכן $L(M)=L(M_1)\circ L(M_2)$.
  - $\alpha=(\alpha_1)^*$, תרגיל! הראו כי $L(M)=L^*(M_1)$.

וכך סיימנו את ההוכחה. $\square$

טענה. כל שפה המתקבלת ע"י אס"ד היא רגולרית (תנאי 1 $\Longleftarrow$ תנאי 3).

הגדרה. <u>אוטומט מוכלל</u> הוא אוטומט בו במקום תווים, על הקשתות יהיו ביטויים רגולריים.
משמעות של חץ מ-q_1 ל-q_2 ע"י הביטוי α היא שמותר לעבור מ-q_1 ל-q_2 אם קוראים מהקלט מילה u כך ש-$u\in L(\alpha)$.
נובע מכך שחץ עם קבוצה ריקה זה כמו לא לשים חץ בכלל.

דוגמה. דוגמא לאוטומט מוכלל.



דוגמת הרצה עבור הקלט 1111 והאוטומט המוכלל לעיל.
דרך אחת לקרוא את דוגמא ההרצה, היא לעבור ע״י הביטוי הרגולרי $1111 \in L\left((1 \circ 1)^*\right)$ למצב התחתון ולעבור ע״י ε למצב העליון (המקבל).

הוכחה. נוכיח כי שפה המתקבלת ע״י אס״ד היא רגולרית.
בידוד וחיבור מקטעים באוטומט זה קשה.
במקום זאת, ניקח את האס״ד ונסבך אותו קצת - נשתמש באוטומט מוכלל.

- תחילה, כל שפה המתקבלת ע״י אס״ד מתקבלת ע״י אוטומט מוכלל (אס״ד הוא בפרט אוטומט מוכלל).

הוכחה. נוכיח את הטענה ע״י הוכחה שכל שפה המתקבלת ע״י אוטומט מוכלל היא רגולרית.
נרצה להפוך אוטומט מוכלל לאוטומט מוכלל המקיים:

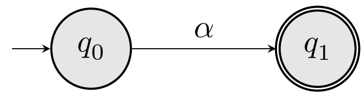
- מצב מקבל יחיד שאינו המצב ההתחלתי.
- יש מעבר בין כמעט כל זוג מצבים (לאו דווקא שונים).
- המעברים היחידים שלא נמצאים:
  - חצי כניסה למצב ההתחלתי.
  - חצי יציאה מהמצב המקבל.

כיצד נעשה זאת?

- בין כל זוג מצבים שאין ביניהם מעבר - נוסיף מעבר עם קבוצה ריקה $\xrightarrow{\emptyset}$.
- הוספת מצב מקבל אחד, ומעברי ε אליו מכל המצבים המקבלים הקודמים (שכעת כבר לא מקבלים).
- מצב התחלתי חדש וממנו מעבר ε למצב ההתחלתי הקודם.

$\square$

בנוסף, נראה כי ניתן להפוך אוטומט מוכלל בעל שני מצבים לביטוי הרגולרי.

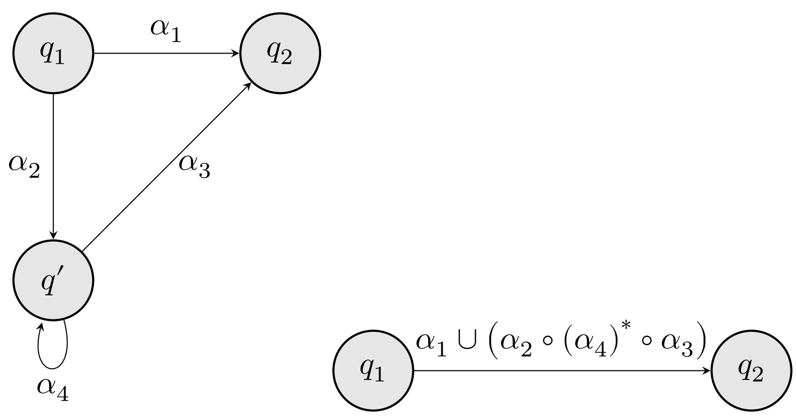


- האוטומט המוכלל חייב להיראות כך, לאור הדרישות הקודמות.

$$L(M) = L(\alpha)$$

כעת, נוכיח כי בהינתן אוטומט מוכלל בעל לפחות 3 מצבים, ניתן להפוך אותו לאוטומט מוכלל בעל 2 מצבים המקבל אותה שפה. לשם כך, נראה כי בהינתן אוטומט מוכלל בעל לפחות $q \geq 3$ מצבים ניתן להפוך אותו לאטוטמט מוכלל בעל $q-1$ מצבים, המקבל את אותה השפה. לאחר מכן, נוריד את מספר המצבים של האוטומט עד שנגיע ל-2 מצבים בלבד.

- מאחר ו-$q > 2$, קיים מצב q' שאינו התחלתי ואינו מקבל. רעיון ההוכחה:
  - נמחק את המצב הזה וננסה לעדכן את המעברים באוטומט כך שמילים שהתקבלו לפני המחיקה עדיין יתקבלו, ובו זמנית לא יתקבלו מילים חדשות.
  - נסתכל במקטע מהאוטומט, יהיו $q_1, q_2 \neq q'$ שני מצבים (לאו דווקא שונים). אילו מילים יכולות להגיע מ-q_1 ל-q_2?



איור 9: משמאל המצב ההתחלתי, ומימין לאחר הצמצום.
במצב ההתחלתי קיימות קשתות נוספות שאינן רלוונטיות.

  - נפעיל טרנספורמציה זו לכל זוג מצבים q_1, q_2 (וגם q_2, q_1), חוץ מהמקרה בו q_1 הוא המצב המקבל ו-q_2 הוא המצב ההתחלתי, וברור שנקבל את אותה השפה.
  - בסופו של דבר - צמצמנו את המצב q'.

קיבלנו כי ניתן להפוך אוטומט מוכלל בעל $q > 2$ מצבים לאוטומט מוכלל בעל $q-1$ מצבים. נפעיל את התהליך באופן איטרטיבי, עד למצב בו נותרו 2 מצבים בלבד.
לבסוף, מאחר וניתן להפוך אוטומט מוכלל בעל שני מצבים לביטוי רגולרי, קיבלנו כי האוטומט המוכלל ההתחלתי שקול לביטוי הרגולרי, וסיימנו את ההוכחה. □

הוכחנו את שתי הטענות. מאחר והראנו קודם לכן כי $2 \iff 1$, קיבלנו כי שלושת התנאים שקולים.

<u>אס"ד יודע לבצע:</u>

1. סגירויות: מאפשר לקחת אוטומטים פשוטים ולחבר אותם למסובכים יותר.
2. שקילויות (לאסל"ד וביטויים רגולריים).
3. כל שפה סופית היא רגולרית.

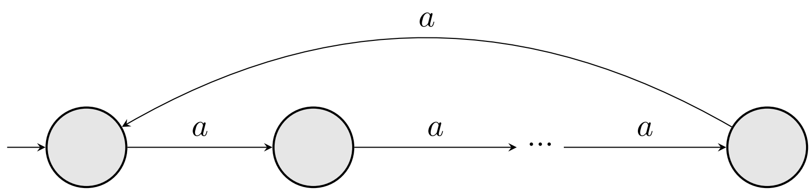
כעת נרצה לדעת מה אס"ד לא מסוגל לבצע - מה אי אפשר לחשב עם אס"ד, והאם יש שפות שאינן רגולריות?

4 שפות שאינן רגולריות

עד כה ראינו מודל חישובי אחד - אס"ד, וראינו כי אס"ד $\equiv$ אסל"ד $\equiv$ ביטוי רגולרי. אילו שפות לא ניתנות להכרעה במודל זה?

דוגמה. השפה $\Sigma = \{a, b\}$, $L = \{a^n b^n \mid n \geq 0\}$ אינה רגולרית.

<u>הרעיון:</u>



איור 10: אוטומט שמקבל (לכאורה) את השפה L.

תהי $w = a^n b^n$ כך ש-$|Q| \ll n$. חייב להיות n שבו האוטומט יחזור למצב בו היה קודם (מעגל - נסמן את אורכו ב-d). מאחר והאוטומט חסר זיכרון, אין דרך להבחין האם זו הפעם הראשונה, השנייה, או השלישית שבה הגיע למצב הזה. לכן, מאחר והמילה מתקבלת גם המילה $a^{n+d}b^n$ תתקבל - והאוטומט נכשל.

נרצה לנצל חולשה זו, באמצעות <u>למת הניפוח לשפות רגולריות</u>.

4.1 למת הניפוח לשפות רגולריות

למה. לכל שפה רגולרית L קיים מספר n_0, כך שלכל $w \in L$ עבורה $|w| \geq n_0$ קיימים $x, y, z \in \Sigma^*$ כך ש-$w = x \circ y \circ z$, ומתקיים:

1. $y \neq \varepsilon$.
2. לכל $k \geq 0$ מתקיים $xy^k z \in L$.
3. $|x \circ y| \leq n_0$.

הערה. הגרירה אינה דו כיוונית - קיימות שפות לא רגולריות שמקיימות את התנאי.

דוגמה. נשתמש בלמת הניפוח להוכחה כי $L = \{a^n b^n | n \geq 0\}$ לא רגולרית.

הוכחה. נוכיח כי L אינה רגולרית.
נניח בשלילה כי L רגולרית. נשתמש בלמת הניפוח, ונקבל כי קיים n_0 כך שלכל $w \in L, |w| \geq n_0$ מתקיימים תנאי הלמה.
תהי $w = a^{n_0} b^{n_0} \in L$. נשים לב כי $|w| = 2n_0 \geq n_0$. נובע מהלמה כי קיימים $x, y, z \in \Sigma^*$ כך ש-$w = xyz$ ומתקיימים תנאי הלמה.

$$w = \underbrace{a \circ a \circ \ldots \circ a}_{n_0} \circ \underbrace{b \circ b \circ \ldots \circ b}_{n_0}$$

נובע מתנאים $\#1, \#3$ כי $y \neq \varepsilon$ ו-$|xy| \leq n_0$. נובע מכך כי x ו-y נמצאים בחלק שמכיל a בלבד, ו-y מכיל אוסף לא ריק של aים: $y = a^c, c \geq 1$.

נשתמש בתנאי $2\#$: נבחר $k = 2$ ונקבל את המילה $xy^2z \in L$, וקיבלנו סתירה - מספר ה-a-ים ב-xyz שווה למספר ה-b-ים בה, והוספנו aים ע״י שרשור y. כלומר, מספר ה-a במילה שונה ממספר ה-b-ים.

על כן, לא ייתכן ש-$xy^2z \in L$ וקיבלנו סתירה - L אינה רגולרית. $\square$

דוגמה. השפה $L = \{1^p \mid p \text{ ראשוני}\}$ אינה רגולרית.

הוכחה. נוכיח כי L אינה רגולרית.

נניח בשלילה כי L רגולרית. נשתמש בלמת הניפוח, ונקבל כי קיים n_0 כך שלכל $w \in L, |w| \geq n_0$ מתקיימים תנאי הלמה.

תהי $w = 1^p \in L$, כך ש-$p \geq n_0$ הוא ראשוני (קיים כזה). נובע מלמת הניבוח כי קיימים $x, y, z \in \Sigma^*$ כך ש-$w = xyz$ ומתקיימים תנאי הלמה. כלומר:

$$w = 1^{p_0} = 1^{|x|}1^{|y|}1^{|z|}$$

נבחר $k = p + 1$, ונקבל את המילה $xy^kz \in L$. אורך המילה הוא $|xyz| + \left|y^{k-1}\right| = p + (k-1)\,|y| = p\,(1 + |y|)$, כלומר אינו ראשוני - בסתירה לכך ש-$xy^kz \in L$.

הגענו לסתירה ולכן L אינה רגולרית. $\square$

הוכחה. נוכיח את נכונות הלמה.

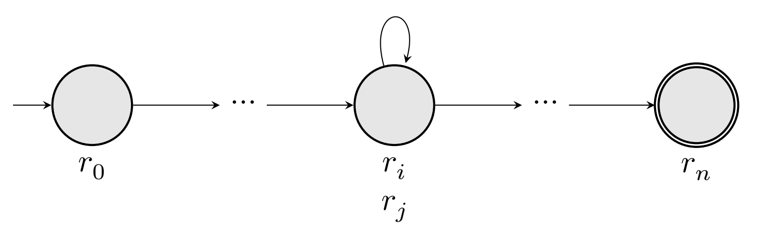
תהי L שפה רגולרית. ידוע ש-L מתקבלת ע״י אס״ד $M = (Q, \Sigma, \delta, q_0, F)$ ונבחר $n_0 = |Q|$.

תהי $w \in L$ כך ש-$|w| \geq n_0$, ונראה כי ניתן לפרק את w כך שיתקיימו תנאי הלמה.

נסתכל על מסלול החישוב של M על w. נסמן $|w| = n \geq n_0$, ונסמן את מסלול החישוב להיות:

$$\underbrace{r_0, r_1, \dots, r_{n_0}}_{n_0+1 \text{ מצבים}}, \dots, r_n$$

לפי עיקרון שובך היונים, ב-$|Q| < n_0 + 1$ המצבים הראשונים יש מצב שחוזר פעמיים: $\exists 0 \leq i < j \leq n_0 : r_i = r_j$.



איור 11: מסלול החישוב של M על w.

התחלנו מ-r_0, הגענו ל-r_i בפעם הראשונה, חזרנו לאותו המצב r_j, ולבסוף ל-r_n המצב המקבל (מאחר ו-$w \in L$). להשלמת ההוכחה, נגדיר את x להיות האותיות במסלול מ-r_0 ל-r_i, את y להיות מ-r_i ל-r_j, ואת z להיות מ-r_j ל-r_n.

נסמן $w = a_1 \dots a_n$ כך ש-$a_i \in \Sigma$. נגדיר:

- $x = a_1 \dots a_{i-1},$, (מה שקראנו עד r_i).
- $y = a_i, \dots, a_{j-1}$ (מה שקראנו בין r_i ו-r_j).
- $z = a_j \dots a_n$ (מה שקראנו אחרי r_j).

נוכיח כי חלוקה זו מקיימת את התנאים:

1. מאחר ו-$i < j$ מתקיים $|y| \geq 1$, וכך $y \neq \varepsilon$.
2. לכל $k \geq 0$, נסתכל על מסלול החישוב של M על המילה xy^kz, ונשים לב כי מסלול זה מסתיים ב-r_n (שהוא מצב מקבל) ולכן $xy^kz \in L$.
3. מהחלוקה קיבלנו כי $xy = a_1 \dots a_{j-1}$. מאחר ובחרנו $j \leq n_0$ מתקיים $|xy| = j - 1 \leq n_0$.

בסך הכל, קיבלנו כי שלושת תנאי הלמה מתקיימים, והוכחנו את נכונות הלמה. □

ניתן להוכיח ע"י רדוקציה ששפה כלשהי אינה רגולרית.

דוגמה. נוכיח כי השפה $L = \{w \mid \#_a \in w = \#_b \in w\}$ אינה רגולרית.

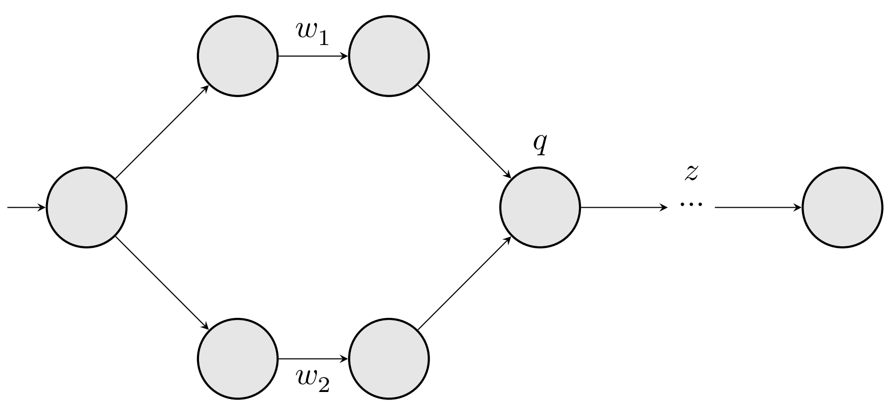
הוכחה. נניח בשלילה כי L רגולרית.
נסתכל על השפה $L' = L(a^*b^*)$, שהיא רגולרית (מתוארת ע"י ביטוי רגולרי).
נגדיר $L'' = L' \cap L = \{a^nb^n \mid n \geq 0\}$.

- מצד אחד, נובע מסגירות של שפות רגולריות תחת חיתוך ש-L'' רגולרית.
- מצד שני, $L'' = \{a^nb^n \mid n \geq 0\}$ ואינה רגולרית.

הגענו לסתירה, ולכן L אינה רגולרית. □

4.2 משפט מייהיל-נרוד

נרצה לנצל חולשה נוספת של אס"ד.



איור 12: שתי מילים w_1, w_2 הגיעו לאותו מצב q.

האוטומט "לא זוכר" האם קרא w_1 או w_2.

הגדרה. בהינתן שפה L, נגדיר שני יחסים על Σ^*.
יחס ההסכמה $\sim_L$: $w_1 \sim_L w_2 \iff w_1, w_2 \in L \vee w_1, w_2 \notin L$.
יחס השקילות $\equiv_L$: $w_1 \equiv_L w_2 \iff \forall z \in \Sigma^* : w_1z \sim_L w_2z$.

הערה. מספר הערות ביחסים ליחסים שהוגדרו לעיל.

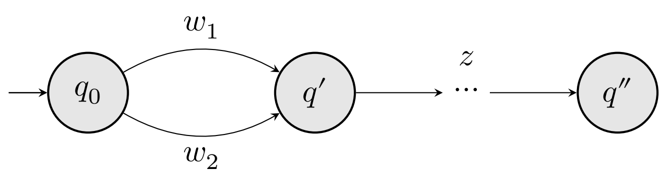
1. שני היחסים הם יחסי שקילות (רפלקסיבי, סימטרי, טרנזיטיבי).
2. אם $w_1 \equiv_L w_2$ אז $w_1 \sim_L w_2$ (נוכל לבחור $z = \varepsilon$ ונקבל $w_1 \sim_L w_2$).
3. יחס ההסכמה לא גורר את יחס השקילות.

דוגמה. יחס ההסכמה לא גורר את יחס השקילות.
נבחר $L = \{w \mid \#_1 \in w \equiv 0 \mod 3\}$.
נסתכל על המילים $100, 110$.
תחילה, $100, 110 \notin L$ ולכן $110 \sim_L 100$. לעומת זאת, עבור $z = 1$ נקבל $1101 \in L$ לעומת $1001 \notin L$.
מכאן, $110 \circ 1 \nsim_L 100 \circ 1$ ולכן $110 \not\equiv_L 100$.

הערה. כל שתי מילים שכמות ה-1ים בהן שווה $\mod 3$ הן שקולות.
יחסי השקילות שמושרים מהיחס מוגדרים ע"פ $\#_1 \mod 3$.

טענה. תהי L שפה המתקבלת ע"י אס"ד M. אם $w_1 \not\equiv_L w_2$ אז $S_M(w_1) \neq S_M(w_2)$.

הוכחה. נוכיח את הטענה.
יהיו w_1, w_2 מילים כך ש-$w_1 \not\equiv_L w_2$: נניח בשלילה כי $S_M(w_1) = S_M(w_2) = q'$.



איור 13: לאחר קריאת w_1 ו-w_2 נמצאים במצב q'.

מאחר ו-$w_1 \not\equiv_L w_2$, קיים $z \in \Sigma^*$ כך ש-$w_1 z \nsim_L w_2 z$.

נניח בה"כ כי $w_1 z \in L \wedge w_2 z \notin L$.
אם $w_1 z \in L$, קיבלנו כי q'' הוא מצב מקבל. מכאן נובע כי גם $w_2 z \in L$, והגענו לסתירה.
לכן $S_M(w_1) \neq S_M(w_2)$ והטענה הוכחה. □

מסקנה. מספר מסקנות מהטענה לעיל.

1. אם L שפה ו-$w_1, \dots, w_t \in \Sigma^*$ כך ש-$w_i \not\equiv_L w_j$ לכל $i \neq j$, אז ל-L אין אס"ד עם פחות מ-t מצבים.
   מאחר ואף זוג אינו שקול כל w_i צריך מצב סיום ייחודי, והאוטומט חייב להכיל לפחות את t מצבי הסיום.

2. תהי L שפה ו-$\{w_i\}_{i=1}^{\infty}$ מילים ב-Σ^* כך ש-$w_i \not\equiv_L w_j$ לכל $i \neq j$. אזי, L אינה רגולרית.
   מהמסקנה הקודמת נובע כי מספר המצבים של האוטומט יהיה אינסופי. לכן, לא ניתן לקבל את השפה ע"י אס"ד והיא אינה רגולרית.

דוגמה. נוכיח באמצעות המסקנות האחרונות כי $L = \{a^n b^n \mid n \geq 0\}$ אינה רגולרית.

הוכחה. נגדיר קבוצת מילים אינסופית שכל זוג מילים אינו שקול: $\{a^n\}_{n=1}^{\infty}$.
אף זוג אינו שקול: נסתכל על זוג המילים a^i, a^j כך ש-$i \neq j$. נסתכל על $z = b^i$, וכך $a^i \not\equiv_L a^j \Longleftarrow a^i b^i \in L \wedge a^j b^i \notin L$.
קיבלנו קבוצה אינסופית של מילים בה כל זוג מילים אינו שקול, ולכן L אינה רגולרית. $\square$

דוגמה. נוכיח כי $L = \{w \mid \exists u : w = uu^R\}$ (u^R היא המילה u כתובה בסדר אותיות הפוך).
למשל, $w = \underbrace{aab}_{u}\underbrace{baa}_{u^R} \in L$.

הוכחה. נוכיח כי L אינה רגולרית.
נסתכל על קבוצת המילים $\{a^n b\}_{n=1}^{\infty}$.
תחילה, לכל $i \neq j$ מתקיים $w_i = a^i b \not\equiv_L a^j b = w_j$: עבור $z = ba^i$ נקבל $w_i z = a^i b \circ ba^i \in L$ ו-$w_j z = a^j b \circ ba^i \notin L$.
כעת, קיבלנו קבוצה אינסופית של מילים בה כל זוג מילים אינו שקול, ולכן L אינה רגולרית. $\square$

<u>סימון:</u> נסמן ב-$\#_L$ את מספר מחלקות השקילות של $\equiv_L$.

- למשל, בשפה $L = \{w \mid \#_1 \in w \equiv 0 \mod 3\}$ מתקיים $\#_L = 3$, ומחלקות השקילות הן קבוצות מילים בעלות מספר 1ים שווה ($\mod 3$).
  - המילים $\varepsilon, 1, 11$ מייצגות את מחלקות השקילות - כל זוג מילים לא שקולות.
- בשפה $L = \{w \mid \#_1 \in w \equiv 0 \mod 1000\}$ מתקיים $\#_L = 1000$, והמילים $\{1^i\}_{i=1}^{1000}$ מייצגות את מחלקות השקילות.

משפט. מייהיל נרוד: תהי L שפה.

1. L רגולרית $\Longleftrightarrow$ $\#_L < \infty$.

2. האוטומט המינימלי עבור L הוא בעל t מצבים $\Longleftrightarrow$ $\#_L = t < \infty$.

הוכחה. נוכיח את המשפט.

1. ראינו כי אם $\#_L = \infty$ אז L אינה רגולרית (מסקנה $\#2$).
מכאן נובע כי אם L רגולרית אז $\#_L < \infty$.

2. ראינו כי אם $\#_L = t < \infty$ האוטומט המינימלי לפחות t מצבים (מסקנה $\#1$).

להשלמת ההוכחה נראה כי אם $\#_L = t < \infty$ אז יש ל-L אס"ד עם t מצבים.
נסמן ב-$[w]_L$ את מחלקת השקילות של המילה w לפי $\equiv_L$.

$$[w]_L = \{w' \in \Sigma^* \mid w' \equiv_L w\}$$

נבנה אסד $M = (Q, \Sigma, \delta, q_0, F)$, המוגדר כך:

- Q יהיו מחלקות השקילות של $\equiv_L$ ($|Q| = t$).
  - נרצה לשמר לכל $w \in \Sigma^*$ ש-$S_M(w) = [w]_L$.
- $q_0 = [\varepsilon]_L$.
- $F = \left\{[w]_L \mid w \in L\right\}$.
  - לכל שתי מילים $w_1, w_2 \in [w]_L$ מתקיים $w_1 \sim_L w_2$, ולכן F מוגדרת היטב וקיימת קביעה אחת לכל יחס שקילות: או שכל המילים מתקבלות והמצב מקבל, או שכל המילים לא מתקבלות והמצב אינו מקבל.
- נגדיר את δ באופן הבא, עבור נציג של מחלקת שקילות u:

$$\delta\left([u]_L, a\right) = [u \circ a]_L$$

  - נראה כי δ מוגדרת היטב, כלומר לכל $u \equiv_L u'$ מתקיים $[u \circ a]_L = [u' \circ a]_L$.
  - כלומר, נראה כי אם $u \equiv_L u'$ אז $u' \circ a \equiv_L u \circ a$ לכל $a \in \Sigma^*$.
  - הטענה נכונה, מאחר ולכל המשך $z \in \Sigma^*$ מתקיים $u' \circ (a \circ z) \sim_L u \circ (a \circ z)$ (נובע ישירות מהגדרת היחס $\equiv_L$).
  - ולכן, קיבלנו כי $u' \circ a \equiv_L u \circ a$ ולכן $[u \circ a]_L = [u' \circ a]_L$, כלומר δ מוגדרת היטב.

ניתן להוכיח באינדוקציה על $|w|$ ש-$S_M(w) = [w]_L$.

- בסיס האינדוקציה: $S_M(\varepsilon) = [\varepsilon]_L$.
- צעד האינדוקציה: תרגיל.

הוכחנו כי אם $\#_L = t < \infty$ אז יש ל-L אס"ד עם t מצבים, והשלמנו את ההוכחה:

1. אם $\#_L < \infty$ קיים אס"ד ל-L ולכן היא רגולרית - הכיוון השני של טענה 1.

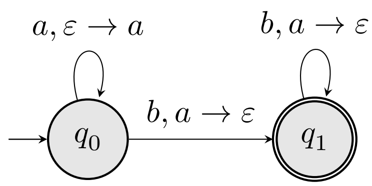
2. ראינו כי מספר המצבים באס"ד הוא לפחות $\#_L$, ולכן אס"ד עם $\#_L$ מצבים הוא מינימלי וסגרנו את טענה 2.

$\square$

5 אוטומט מחסנית

ניתן להסתכל על אס"ד בתור מחשב "ללא זיכרון", וכך למשל השפה $\{a^nb^n|n\geq 0\}$ אינה רגולרית. אוטומט מחסנית הוא אס"ד + מחסנית: זיכרון מוגבל ופשוט. נדבר על אוטומט מחסנית (א"מ) לא דטרמיניסטי - נרצה להבין מה אי אפשר לחשב במודל, ולכן נתרכז במודל חזק יותר. אין שקילות בין אוטומט מחסנית דטרמיניסטי ולא דטרמיניסטי.

דוגמה. נבנה אוטומט מחסנית עבור השפה $L=\{a^nb^n \mid n\geq 1\}$.



איור 14: אוטומט מחסנית עבור השפה L.

משמעות הלולאה העצמית של q_0 היא: אתה רשאי לקרוא a מהקלט, ובלבד שתכניס a למחסנית. הסימון $a\to\varepsilon\; a$ מסמל: הוצא כלום מהמחסנית ושים שם a.

שמורות:

1. המחסנית בהתחלה ריקה (נסמן ב-ε).
2. מסלול חישוב נקרא מקבל אם הוא סיים לקרוא את המילה, הגיע למצב מקבל **וגם** המחסנית ריקה.

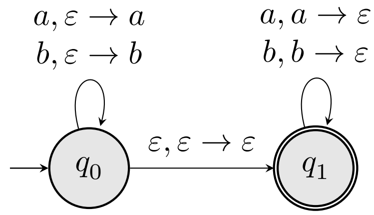
דוגמה. נסתכל על מסלול החישוב של המילה $w=aabb$ באוטומט.

מחסנית	מצב	קריאת הקלט
ε	q_0	$\overset{\downarrow}{a}abb$
a	q_0	$a\overset{\downarrow}{a}bb$
aa	q_0	$aa\overset{\downarrow}{b}b$
a	q_1	$aab\overset{\downarrow}{b}$
ε	q_1	$aabb^{\downarrow}$

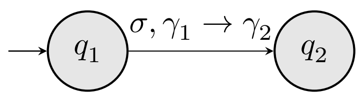
טבלה 4: מסלול החישוב של המילה $aabb$

המסלול מסתיים ב-q_1 והמחסנית ריקה, ולכן המילה מתקיימת. המילה aab לא תתקבל: למרות שמצב הסיום הוא מקבל המחסנית אינה ריקה.

דוגמה. נסתכל על השפה $L = \{w \mid \exists u : w = uu^R\}$ מעל $\Sigma = \{a, b\}$.
ראינו כי השפה אינה רגולרית. נרצה להראות אוטומט מחסנית שמקבל את L.



איור 15: א"מ שמקבל את L.
קליטת u נעבור במעבר ε ל-q_1, ונשמיט מהמחסנית לפי אותיות u^R.



איור 16: מעברי הא"מ.
σ היא תו או ε, ו-γ_1, γ_2 הם תווים של המחסנית. ניתן לעבור מ-q_1 ל-q_2 בתנאי שקראנו σ מהקלט, הוצאנו γ_1 מהמחסנית והכנסנו γ_2 (אם $\gamma = \varepsilon$ לא מוציאים/מכניסים מהמחסנית כלל).

הגדרה. אוטומט מחסנית הוא שישייה $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$, כך ש:

- Q - קבוצה סופית של מצבים.
- Σ - אלפבית הקלט.
- Γ - אלפבית המחסנית.
- $q_0 \in Q$ - מצב התחלתי.
- $F \subseteq Q$ - קבוצת המצבים המקבלים.
- $\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \to P\left(Q \times \Gamma_\varepsilon\right)$ - פונקציית המעברים (מקבלת מצב נוכחי, תו מהמילה וראש המחסנית. מוציאה קבוצה של זוגות מצבים ותו חדש למחסנית).
  <u>משמעות:</u> כאשר אנחנו במצב q_1, קוראים תו σ ובראש המחסנית נמצא γ_1, $(q_2, \gamma_2) \in \delta\left(q_1, \sigma, \gamma_1\right)$ אם אפשר לעבור ל-q_2 ולשים γ_2 בראש המחסנית.

הגדרה. בהינתן א"מ M ומילה w, נגדיר מסלול חישוב של M על w, המורכב משתי סדרות:

$$r_0, \dots, r_n \in Q; s_0, \dots, s_n \in \Gamma^*$$

כאשר s_i הוא תוכן המחסנית ברגע ה-i. הסדרת מהוות מסלול חישוב אם, עבור $w = a_1 \circ \cdots \circ a_n, a_i \in \Sigma_\varepsilon$:

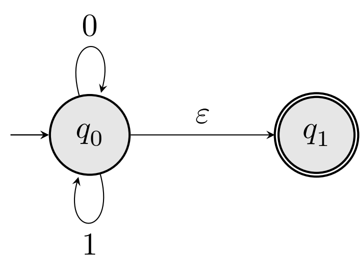
1. $r_0 = q_0, s_0 = \varepsilon$: בהתחלה המחסנית ריקה ומתחילים מ-q_0.
2. לכל $0 \le i < n$ קיים s כך ש-$s_i = \gamma_1 \circ s$, $s_{i+1} = \gamma_2 \circ s$, וגם $(r_{i+1}, \gamma_2) \in \delta\left(r_i, a_{i+1}, \gamma_1\right)$ (קיים מעבר מ-r_i ל-r_{i+1}).

הגדרה. בהינתן א"מ M, השפה של M מוגדרת להיות:

$$L(M)=\left\{w \mid r_n, s_n \text{ בזוג } w \text{ שמסתיים } \text{ על } w \text{ של } M \text{ יש מסלול חישוב } \wedge \underbrace{s_n=\varepsilon}_{\text{המחסנית ריקה}} \wedge \underbrace{r_n\in F}_{\text{הגענו למצב מקבל}}\right\}$$

5.1 סגירויות

אין סגירות תחת משלים: אפילו באסל"ד, הבנייה של הפיכת מצבים מקבלים ללא-מקבלים לא תעבוד. דוגמא:



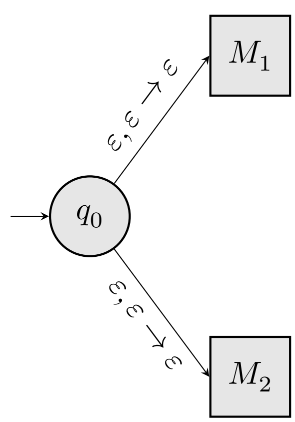
איור 17: דוגמא נגדית לסגירות א"מ תחת משלים.

שפת האסלד היא Σ^*, במידה ונהפוך את המצבים השפה החדשה תהיה גם Σ^*, ושיטה זו לא תעבוד.

בנוסף, אין סגירות א"מ לחיתוך (אינטואיציה - בניית מכפלה לא תעבוד כי אין לנו 2 מחסניות).

משפט. אם L_1, L_2 מתקבלות ע"י א"מ, אז גם $L_1 \cup L_2$.

הוכחה. נבנה אסל"ד באופן הבא.



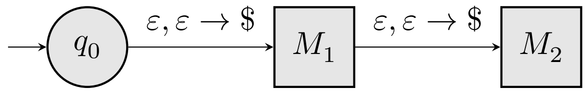
איור 18: אסל"ד האיחוד עבור $L_1 \cup L_2$.

נחבר מצב התחלתי למצבים ההתחלתיים של M_1 ו-M_2 עם מעברי ε.

□

משפט. אם L_1, L_2 מתקבלות ע"י א"מ, אז גם $L_1 \circ L_2$.

הוכחה. נבנה אסל"ד באופן הבא.



איור 19: אסל"ד השרשור עבור השפה $L_1 \circ L_2$.

נעזרנו בתו נוסף $\$$ כדי לוודא שהמחסנית ריקה. נחבר את המצב ההתחלתי q_0 למצב ההתחלתי של M_1, ומכל מצב מקבל ב-M_1 נחבר את המעבר למצב ההתחלתי של M_2.

$\square$

פעולה	משלים	חיתוך	איחוד	שרשור	כוכב
סגירות	X	X	✓	✓	✓

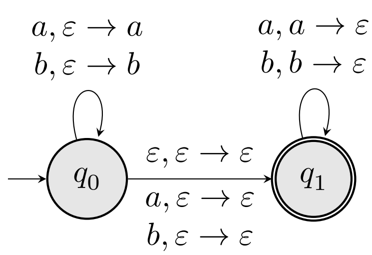
טבלה 5: סגירויות של א"מ

משפט. אם L_1 רגולרית ו-L_2 מתקבלת ע"י א"מ, אזי $L_1 \cap L_2$ מתקבלת ע"י א"מ.

דוגמה. מספר דוגמאות לשפות המתקבלות ע"י א"מ.

$$L_1 = \{w \mid w = u \circ u^R\}$$

$$L_2 = \{w \mid w = w^R\}$$



איור 20: א"מ שמקבל את השפה L_2.

המעברים מ-q_0 ל-q_1 שאינם מעברי ε גורמים לקבלת פלינדרומים מאורך אי זוגי - בלעדיהם נקבל פלינדרומיים זוגיים בלבד, כלומר את L_1.

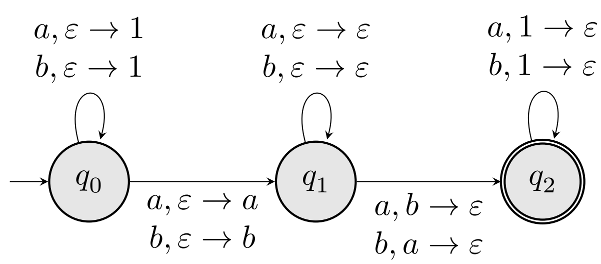
בהינתן השפה $L = \{w \mid w = w^R\}$, נרצה לבנות א"מ שמקבל את $\overline{L} = \{w \mid w \neq w^R\}$.

$$\overline{L} = \{w \mid \text{קיים } i \text{ כך שהמקום ה-} i \text{ מההתחלה ומהסוף שונים}\}$$

<u>רעיון:</u> ננחש באופן לא דטרמיניסטי את i.

- נקרא את התווים עד המקום ה-i.
- נכניס את התו ה-i למחסנית.
- נקרא את התווים שאחרי המקום ה-i, ננחש באופן לא דטרמיניסטי שהגענו למקום ה-i מהסוף.
- נשווה את התו שאנחנו קוראים לתו במחסנית, ואם הם שונים המילה תתקבל.

נבנה את הא"מ.



איור 21: א"מ עבור השפה $\overline{L}$.

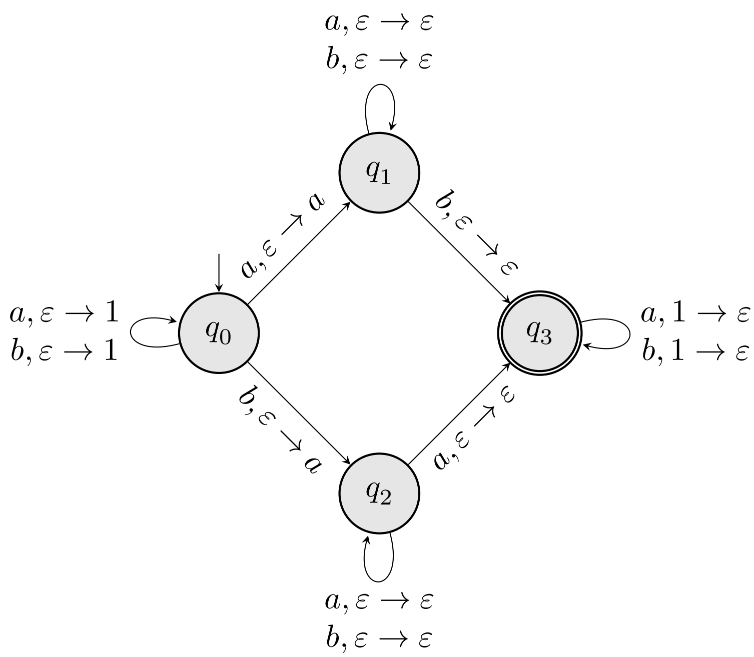
הלולאות העצמיות של q_0 אחראיים על ספירת התווים עד ה-i. התו ה-i נקרא במעבר מ-$q_0 \to q_1$. הלולאות העצמיות של q_1 עוברות על אמצע המילה, והמעבר $q_1 \to q_2$ יקרא את האות ה-i מהסוף - במידה והוא שונה מראש המחסנית המילה תתקבל.

הוכחה. נוכיח כי הא"מ מקבל את השפה $\overline{L}$ - נחלק את המילה $w = \sigma_1 \cdots \sigma_n$ לשלושה חלקים:

- הרישא בה נמצאים בלולאה העצמית של q_0.
- תת המילה בה נמצאים בלולאה העצמית של q_1.
- הסיפא בה נמצאים בלולאה העצמית של q_2.

אורך החלק הראשון והשלישי זהה, מאחר והראשון מכניס 1-ים למחסנית והשלישי מוציא 1-ים.
בנוסף, הבטחנו שהתו ה-i מההתחלה (שאיתו נעבור $q_0 \to q_1$) יהיה שונה מה-i מהסוף (שאיתו נעבור $q_1 \to q_2$).
וכך, w תתקבל אמ"מ היא אינה פלינדרום. $\square$

הערה. ניתן לבנות את האוטומט גם כך.



איור 22: א"מ עבור השפה $\overline{L}$.

נפצל את q_1 מהא"מ הקודם לשני מצבים, וכך לא נצטרך להוסיף תו נוסף למחסנית.

הערה. מה היה משתנה אם הא"ב היה $\{a, b, c\}$?

דוגמה. קיים א"מ לשפה $L = \{w \mid w = u_1 \circ u_2, u_1 \neq u_2 \wedge |u_1| = |u_2|\}$.
נסתכל על השפה $\{w \mid w = u_1 \circ u_2^R, u_1 \neq u_2 \wedge |u_1| = |u_2|\}$. זו שפת החיתוך של השפה שראינו קודם (מילים שאינן פלינדום), עם שפת המילים מאורך זוגי (רגולרית), ולכן יש לה א"מ.
כעת, נראה כי L מתקבלת ע"י א"מ.

$$L = \{w \mid \exists k, i : |w| = 2k \wedge w_i \neq w_{k+i}\}$$

<u>הרעיון:</u> לנחש באופן לא דטרמיניסטי את i.
<u>הבעיה:</u> אוטומט מחסנית יודע לבדוק רק אורכי אינטרוולים זהים. ננסה לייצג את הבעיה באמצעות אורכים זהים:

- ב-q_0, עבור כל תו שנקרא נכניס 1 למחסנית.
- באופן לא דטרמיניסיטי, נחליט שהגיע ה-i ונכניס את התו שאנחנו קוראים לרגיסטר (נשמור ע"י המצב) ונעבור ל-q_1.
- ב-q_1, על כל תו שנקרא נוציא איבר מהמחסנית, ורק אם המחסנית ריקה נעבור ל-q_2 (נבדוק האם במחסנית ריקה באמצעות תו נוסף %).
- ב-q_2, על כל תו שנקרא נכניס 1 למחסנית. באופן לא דטרמיניסיטי נניח שהגיע המקום ה-$k + i$, ואז נקרא תו מהקלט ונשווה אותו לתו ה-i - אם התווים שונים נעבור ל-q_3.
- ב-q_3, על כל תו שנקרא נוציא תו מהמחסנית, ואם היא ריקה נקבל.

דוגמה. נסתכל על השפה $L = \{w \mid w = u_1 \circ u_2, u_1 = u_2\}$.
נראה בהמשך ש-L לא ניתנת לקבלה ע"י א"מ.

6 דקדוק חסר הקשר

דקדוק חסר הקשר הוא רשימה של חוקים שגוזרת שפה. יש הבחנה בין שני סוגי תווים:

- טרמינלים - מסומנים באותיות קטנות.
- נונטרמינלים - מסומנים באותיות גדולות.

כל חוק הוא מהצורה $A \to \varepsilon$ כאשר A הוא נונטרמינל ו-α היא מילה שיכולה להכיל גם טרמינלים וגם נונטרמינלים.

דוגמה. נסתכל על החוקים $S \to aSb$ ו-$S \to \varepsilon$.

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb$$

התהליך מסתיים כאשר אין נונטרמינלים. השפה שנגזרת מדקדוק זה היא $L = \{a^n b^n \mid n \geq 0\}$.

הגדרה. השפה הנגזרת מדקדוק חסר הקשר (דח"ה) היא כל המילים שיכולות להיגזר ממנו ומורכבות רק מטרמינלים.

דוגמה. מספר דוגמאות לדקדוקים ולשפות המושרות מהן.

$$\begin{aligned} S &\to aSa \\ S &\to bSb \\ S &\to \varepsilon \end{aligned}$$

השפה היא שפת כל הפלינדרומים הזוגיים: $L = \{w \mid w = u \circ u^R\}$.
במידה ונרצה לקבל פלינדרומים מכל אורך, נוסיף את החוקים $S \to a, S \to b$.

דוגמה. S הוא נונטרמינל התחלתי והוא יחיד, אך ייתכנו נונטרמינלים נוספים.

$$\begin{aligned} S &\to AB \\ A &\to aAb \\ A &\to \varepsilon \\ B &\to bB \\ B &\to \varepsilon \end{aligned}$$

השפה היא $L = \{a^n b^m \mid m \geq n \geq 0\}$, מאחר ו-$A$ יכול לייצר $a^n b^n$, ו-B (מימינו) מוסיף b-ים בלבד. מילה לדוגמא:

$$\begin{aligned} S &\Rightarrow AB \\ &\Rightarrow aAbB \\ &\Rightarrow aAbbB \\ &\Rightarrow aa\underbrace{A}_{\to\varepsilon}bbb\underbrace{B}_{\to\varepsilon} \\ &\Rightarrow \boxed{aabbb} \end{aligned}$$

הגדרה. דקדוק חסר הקשר הוא רביעייה $G=(V,\Sigma,R,S)$, המוגדר כך:

- Σ - א"ב סופי שאיבריו נקראים טרמינלים, זה א"ב המילים הנגזרות.
- V - א"ב סופי שמכיל גם את א"ב הטרמינלים, Σ, וגם את הא"ב של הנונטרמינלים. כלומר $\Sigma\subset V$, ואיברי $V\setminus\Sigma$ נקראים נונטרמינלים.
- $R\subseteq(V\setminus\Sigma)\times V^*$ - קבוצה סופית של זוגות מהצורה (A,α), כאשר $A\in V\setminus\Sigma$ (נונטרמינל) ו-$\alpha\in V^*$.
  הזוג (A,α) מתפרש לחוק $A\to\alpha$.
- $S\in V\setminus\Sigma$ - נונטרמינל התחלתי.

הגדרה. עבור דח"ה $G=(V,\Sigma,R,S)$, יחס הגזירה $\underset{G}{\Rightarrow}$ על V^* מוגדר באופן הבא:

$$\forall w_1,w_2\in V^*: w_1\underset{G}{\Rightarrow}w_2\iff\exists x,y,\alpha\in V^*,A\in V\setminus\Sigma:\ \begin{matrix}w_1=xAy\\ w_2=x\alpha y\end{matrix}\ \wedge(A,\alpha)\in R$$

הגדרה. עבור דח"ה G נגדיר את השפה של G באופן הבא:

$$L(G)=\{w\in\Sigma^*\mid \text{קיימת סדרה סופית של צעדי גזירה שמתחילה ב-}S\text{ ומסתיימת ב-}w\}$$

הערה. רלוונטי בבלשנות, שפות תכנות, קומפיילרים ועוד.

מטרה: לקבוע אוסף חוקים שמגדיר מה זה משפט נכון (/תכנית תקינה).

בהקשר שלנו - נראה כי דח"ה $\iff$ א"מ (בהמשך נקרא לשפה המתקבלת ע"י א"מ שפה חסרת הקשר - ח"ה), וניעזר בזה בהמשך כדי להראות שפות שאינן מתקבלות ע"י א"מ.

דוגמה. בהינתן הא"ב $\Sigma=\{0,...9,+,*,(,)\}$, נרצה להגדיר את שפת הביטויים החשבוניים התקינים.

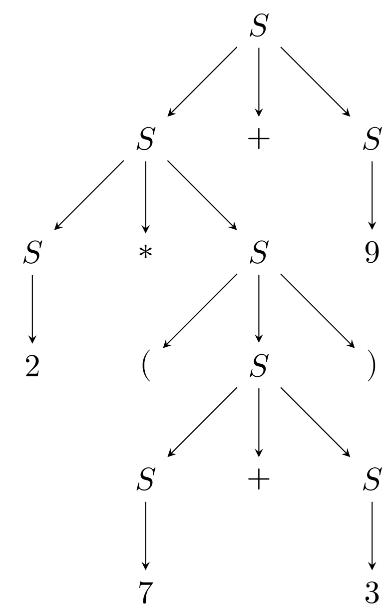
$$S\to S+S\mid S*S\mid(S)\mid0\mid1\mid\cdots\mid9$$

ביטוי חשבוני תקין לדוגמא:

$$\begin{aligned}S&\Rightarrow S+S\\&\Rightarrow S*S+S\\&\Rightarrow 2*S+S\\&\Rightarrow 2*(S)+S\\&\Rightarrow 2*(S)+9\\&\Rightarrow 2*(S+S)+9\\&\Rightarrow 2*(7+S)+9\\&\Rightarrow 2*(7+3)+9\end{aligned}$$

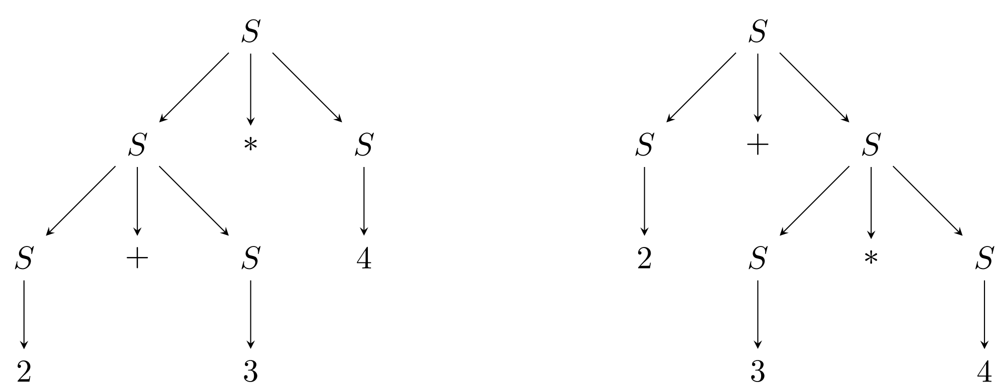
הגדרה. עץ גזירה הוא עץ שכל קודקוד בו מסומן ע"י תו ב-V (טרמינל או נונטרמינל), השורש מסומן ע"י הנונטרמינל S. לכל קודקוד פנימי, אם הקודקוד מסומן בנונטרמינל A, אז יהיו לו בנים $a_1, \dots, a_t \in V$ כך שקיים חוק $A \to a_1 \cdots a_t$.

דוגמה. דוגמא לעץ גזירה השקול לביטוי החשבוני התקין לעיל.



איור 23: עץ גזירה השקול לביטוי החשבוני $2 * (7 + 3) + 9$.

<u>עמימות:</u> נסתכל במילה $2 + 3 * 4$, למילה יש שני עצי גזירה שונים!



איור 24: שני עצי גזירה שונים, עבור הביטויים $2 + (3 * 4)$ ו-$(2 + 3) * 4$ (מימין לשמאל).

ניתן לפתור את בעיית העמימות באמצעות סוגריים.

הגדרה. שפה נקראת חסרת הקשר (ח"ה) אם יש דח"ה שגוזר אותה.

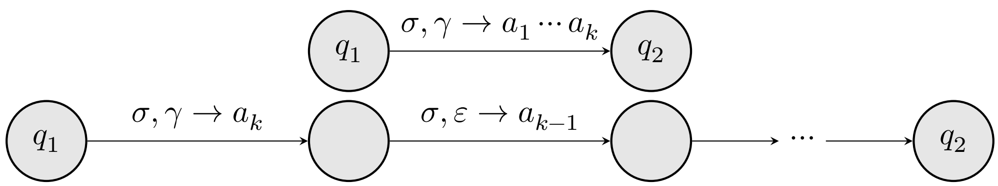
6.1 דח"ה $\Longleftrightarrow$ א"מ

משפט. שני התנאים הבאים שקולים.

1. L מתקבלת ע"י א"מ (לא דטרמיניסטי).
2. L חסרת הקשר.

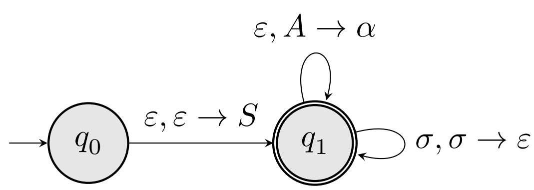
הוכחה. נוכיח את שקילות התנאים.
תחילה, נראה כי כל שפה חסרת הקשרת מתקבלת ע"י א"מ.

- עד כה לא הרשינו לדחוף בבת אחת שני תווים למחסנית - מעכשיו ניתן.



איור 25: דחיפה של מספר תווים למחסנית.
נעביר מעבר בו מכניסית מספר תווים למחסנית (מלמעלה) לסדרת המצבים (מלמטה).

- נתון דח"ה G, נרצה לבנות א"מ M כך ש-$L(M) = L(G)$.
  - הא"מ ינסה לחקות גזירה בדקדוק G, וינחש באופן לא דטרמיניסטי איך כדאי לגזור, כדי להצליח לגזור את המילה הנתונה כקלט.
  - א"מ בעל 2 מצבים בלבד, א"ב המחסנית הוא $\Gamma = V$ (V - א"ב שכולל את הטרמינלים ואת הנונטרמינלים בדקדוק).



איור 26: אוטומט המחסנית M.
תחילה, מאחסנים את המחסנית ב-S ע"י מעבר ה-ε הראשון. עבור כל חוק $A \to \alpha$ בדקדוק יהיה מעבר בלולאה העצמית של q_1: $\varepsilon, A \to \alpha$. בנוסף, לכל $\sigma \in \Sigma$ יהיה מעבר $\sigma, \sigma \to \varepsilon$ בלולאה העצמית של q_1. משמאל, דוגמא עבור הדח"ה $S \to$.

- יש להראות כי $w \in L(G)$ אמ"מ $w \in L(M)$.

דוגמה. דוגמא עבור הכיוון הראשון, עבור הדח"ה
$$\begin{array}{l} S \to AB \\ A \to aAb \mid \varepsilon \\ B \to bB \mid \varepsilon \end{array}$$
, שגוזר את השפה $L = \{a^n b^m \mid m \geq n \geq 0\}$.
נסתכל על המילה $abb \in L(G)$:

$$S \Rightarrow AB \Rightarrow aAbB \Rightarrow abB \Rightarrow abbB \Rightarrow abb$$

נראה כי האוטומט יקבל ע"פ גזירה זו:

מחסנית	מצב	קריאת הקלט
ε	q_0	$\overset{\downarrow}{a}bb$
S	q_1	$\overset{\downarrow}{a}bb$
AB	q_1	$\overset{\downarrow}{a}bb$
$aAbB$	q_1	$\overset{\downarrow}{a}bb$
AbB	q_1	$a\overset{\downarrow}{b}b$
bB	q_1	$a\overset{\downarrow}{b}b$
B	q_1	$ab\overset{\downarrow}{b}$
bB	q_1	$ab\overset{\downarrow}{b}$
B	q_1	$ab\overset{\downarrow}{b}$
ε	q_1	$abb^{\downarrow}$

טבלה 6: מסלול החישוב של המילה abb באוטומט

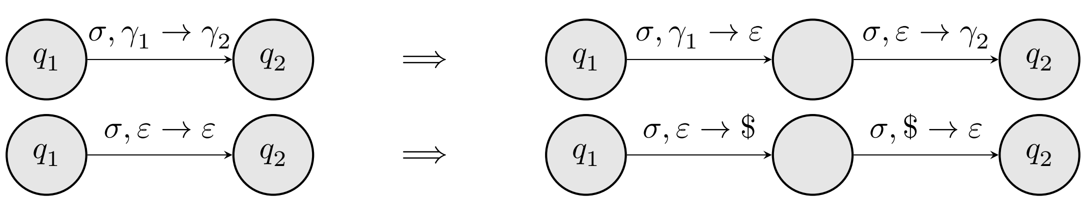
- אפשר להכליל את הדבר לכל דקדוק (תרגיל).

כעת, נראה כי אם המילה מתקבלת ע"י הא"מ M אז מסלול החישוב מגדיר תהליך גזירה של המילה ב-G.

נתון א"מ M, ונרצה לבנות דח"ה G כך ש-$L(M) = L(G)$.
נעביר את $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$ לצורה יותר נוחה:

1. נניח בה"כ כי ל-M מצב מקבל יחיד q' (ניתן להוסיף מעברי ε מהמצבים המקבלים למצב מקבל יחיד).

2. בכל מעבר האוטומט M דוחף <u>או</u> מוציא מהמחסנית, אך לא שניהם.



איור 27: נמיר מעבר בודד (משמאל) לשני מעברים עם מצב דמה (מימין).
למעלה, עבור מעברים בהם מוציאים וגם מכניסים, ומטה עבור מעבר בו לא מוציאים ולא מכניסים.

לכל זוג מצבים (לאו דווקא שונים) $p,q\in Q$ נגדיר שפה $L_{p,q}$:

$$L_{p,q}=\{w\mid \text{ניתן להתחיל ב-}p\text{ עם מחסנית ריקה, לקרוא }w\text{ ולהגיע ל-}q\text{ עם מחסנית ריקה}\}$$

$$\Longrightarrow L\left(M\right)=L_{q_0,q'}$$

נבנה דח"ה $G=\left(V,\Sigma,R,S\right)$ בשלבים - ממצבים קרובים לרחוקים.

- הטרמינלים הם Σ.
- נונטרמינלים: לכל זוג מצבים p,q (לאו דווקא שונים) יהיה נונטרמינל $A_{p,q}$.
- נבנה חוקים כך שהמילים הנגזרות ע"י $A_{p,q}$ יהיו בדיוק המילים ב-$L_{p,q}$.
  - נגדיר את $A_{q_0,q'}$ בתור הנונטרמינל ההתחלתי.
- יהיו $p,q\in Q$, ותהי $w\in L_{p,q}$ מילה, $p\overset{w}{\rightsquigarrow}q$.
- המחסנית ריקה ב-p וב-q. נפריד למקרים:
  - היה שלב בדרך בו המחסנית ריקה, וכך נוכל להוסיף את החוק $A_{p,q}\to A_{p,r}A_{r,q}$ לכל $r\in Q$.

$$w=w_1\circ w_2,w_1\in L_{p,r}\wedge w_2\in L_{r,q}$$

מניחים באופן רקורסיבי שיש לנו את $A_{p,r}$ ואת $A_{r,q}$.

  - המחסנית לא הייתה ריקה בשום שלב בדרך.
מכאן, התו הראשון שהוכנס הוא זה שהוצא אחרון. נוכל להסתכל תת המילה שקראנו פרט לקצוות - $w'\in L_{r,s}$, כך ש-r העוקב של p ו-s הקודם q.
לכל $r,s\in Q$, אם קיימים $\gamma\in\Gamma$ והמעברים $p\overset{\sigma_1,\varepsilon\to\gamma}{\longrightarrow}r,s\overset{\sigma_2,\gamma\to\varepsilon}{\longrightarrow}q$, נוסיף את החוק $A_{p,q}\to\sigma_1A_{r,s}\sigma_2$.

- מקרי בסיס: $A_{p,p}\to\varepsilon$ לכל $p\in Q$.

ניתן להראות שכל מילה המתקבלת ע"י האוטומט ניתנת לגזירה ע"י אחד משני המקרים.
בסך הכל, הוכחנו את שקילות שתי הטענות. $\square$

6.2 למת הניפוח לשפות ח"ה

מה שפות ח"ה לא יכולות לעשות? בדומה לשפות רגולריות, קיימת למת ניפוח לשפות ח"ה.

למה. לכל שפה ח"ה L קיים $n_0\in\mathbb{N}$ כך שלכל מילה $w\in L$ שמקיימת $|w|\geq n_0$, קיימים $u,v,x,y,z\in\Sigma^*$ כך ש-$w=u\circ v\circ x\circ y\circ z$, ומתקיים:

1. $v\circ y\neq\varepsilon$
2. לכל $k\geq 0$: $u\circ v^k\circ x\circ y^k\circ z\in L$
3. $|v\circ x\circ y|\leq n_0$

דוגמה. נשתמש בלמה כדי להראתו שהשפה $L = \{a^n b^n c^n \mid n \geq 0\}$ אינה ח"ה.

נניח בשלילה כי L ח"ה, ונסתכל על המילה $w = a^{n_0} b^{n_0} c^{n_0}$ שאורכה $3n_0 \geq n_0$.

מהלמה קיימים u, v, x, y, z שמקיימים את התנאים.

מאחר ו-$|vxy| \leq n_0$ $v \circ x \circ y$ לא יכול להכיל את שלושת סוגי התווים, בה"כ vxy לא מכיל a-ים.

מאחר ו-$|vy| > 0$, בניפוח $uv^2xy^2z \in L$ נוספו b-ים ו-c-ים אך לא a-ים, וכך לא אפשרי שבניפוח מספר שווה של a-ים ו-b-ים, והגענו לסתירה - L אינה ח"ה.

הערה. ראינו כי שפות ח"ה סגורות תחת איחוד ושרשור. עם זאת, אין סגירות תחת חיתוך. נסתכל על השפות:

$$L_1 = \{a^n b^n c^m \mid n, m \geq 0\}, L_2 = \{a^m b^n c^n \mid n, m \geq 0\}$$

L_1 ו-L_2 הן ח"ה - נובע משרשור של השפות ח"ה $\{a^n b^n \mid n \geq 0\}, \{c^n \mid n \geq 0\}$ ובאופן דומה עבור L_2. לעומת זאת, $L_1 \cap L_2 = \{a^n b^n c^n \mid n \geq 0\}$, שאינה רגולרית.

דוגמה. השפה $L = \left\{w \mid w = u \circ u, u \in \{0,1\}^*\right\}$ אינה ח"ה.

נניח בשלילה כי L ח"ה, ונסתכל על המילה $w = 0^{n_0} 1^{n_0} 0^{n_0} 1^{n_0} \in L$, שאורכה $4n_0 \geq n_0$.

כך, קיימים u, v, x, y, z כך ש-$w = u \circ v \circ x \circ y \circ z$ ומתקיימים תנאי הלמה.

$vy \neq \varepsilon$, ועבור $k = 0$ נקבל כי $uxz = 0^{n_1} 1^{n_2} 0^{n_3} 1^{n_4} \in L$. נפריד למקרים:

- vxy נמצאים בה"כ בחצי השמאלי של המילה - תת מחרוזת של $0^{n_0} 1^{n_0}$ השמאליים.

$$uxz = 0^{n_1} 1^{n_2} 0^{n_0} 1^{n_0}$$

  - מכאן, $n_0 \leq n_1 + n_2 < 2n_0$ (הורדנו עם vy לפחות 1 וכל היותר n_0 תווים מהחצי השמאלי).
  - נובע מכך שלא ניתן לחלק את uxz לשתי מילים שוות - אמצע המילה של uxz ימני מאמצע $uvxyz$ - החלק השמאלי יסתיים ב-0 והימני ב-1.
  - על כן, מקרה זה לא ייתכן.

- vxy עוברים באמצע המילה.

  - מאחר ו-$|vxy| \leq n_0$, vxy לא כולל את רישת ה-0-ים או את סיפת ה-1-ים. לכן:

$$uxz = 0^{n_0} 1^{n_2} 0^{n_3} 1^{n_0}, n_2 < n_0 \vee n_3 < n_0$$

  - בה"כ $n_2 < n_0$, וכך מספר ה-1-ים בחצי השמאלי של המילה קטן ממספר ה-1-ים בחצי הימני, ו-$uxz \notin L$ בסתירה ללמה.

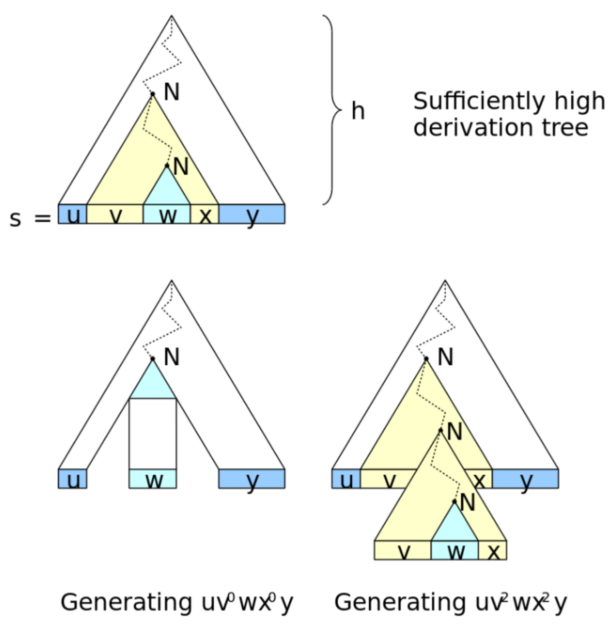
בסך הכל, קיבלנו כי בכל מקרה $uxz \notin L$ בסתירה ללמת הניפוח, ולכן L אינה ח"ה.

דוגמה. נסתכל על השפה $L_1=\{w\mid w=u_1\circ u_2,|u_1|=|u_2|,u_1\neq u_2\}$.

- זו שפה חסרת הקשר שלא משלימה את השפה הקודמת - מכילה מילים עם אורך זוגי בלבד.
- נגדיר $L_2=\{w\mid |w|\equiv 1 \mod 2\}$, שפה רגולרית ולכן ח"ה.
  - השפה $L_3=L_1\cup L_2$ היא ח"ה מסגירות של איחוד על ח"ה.
  - בנוסף, $\overline{L_3}=L=\{w\mid w=u\circ u,u\in\Sigma^*\}$.
- כך, L_3 היא דוגמא לשפה ח"ה שהמשלים שלה אינו ח"ה.

הערה. אינטואיציה להוכחת למת הניפוח לשפות ח"ה.

- בהינתן דח"ה, נבחר מילה w "ארוכה מספיק" (באורך הגדול מהחוק הארוך ביותר בחזקת מספר הנונטרמינלים), שעבורה אורך מסלול הגזירה גדול ממספר הנונטרמינלים.
  - קיים נונטרמינל שחוזר על עצמו.
- נסמן ב-vxy את תת המילה תחת תת העץ של המופע הראשון של A, וב-x את תת המילה תחת ה-A השני. כך, נוכל להחליף את x ב-vxy ולנפח את המילה, באופן דומה ללמת הניפוח לשפות רגולריות.



איור 28: עץ הגזירה, w מחולק ל-$uvxyz$.
נוכל להחליף את x ב-vxy - וכך לנפח את המילה ל-uv^kxy^kz. מתוך: ויקיפדיה.

חלק II

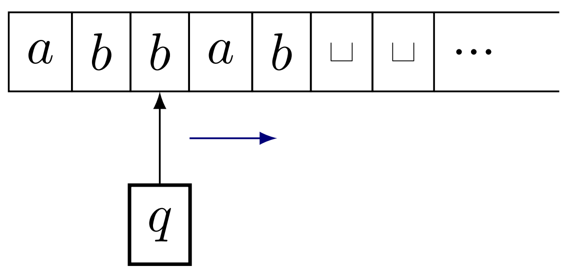
מכונת טיורינג

נעסוק במודל חזק יותר מאוטומט - מכונת טיורינג, השקול למחשב מבחינת כוח החישוב. נעסוק בשאלות:

- מה מחשבים יכולים לחשב?
- מה מחשבים יכולים לחשב באופן יעיל?

<u>אינטואיציה:</u>

- אמצעי קלט/פלט - סרט אינסופי בכיוון אחד.
- זיכרון - רשימה מקושרת דו-כיוונית.
- מצב, שנשמר במין רגיסטר.
- <u>פעולות:</u> קריאה, כתיבה, תזוזה ימינה/שמאלה.

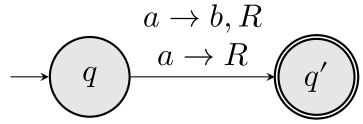


איור 29: סליל הקלט/פלט.

המצב הנוכחי q בתו הנוכחי. התו $\sqcup$ אינו חלק מהקלט, ומופיע לעד.

נדבר על קלט שכבר נמצא בזיכרון בתחילת הריצה.

- מתחילים כשהראש הקורא בראש המילה.
- בכל <u>צעד חישוב</u>:
  - הסתכל במצב וקרא את התו עליו מצביע הראש הקורא.
  - כפונקציה של המצב והתו שקראת - שנה את המצב, כתוב תו במקום שאליו מצביע הראש הקורא וזוז ימינה/שמאלה.



איור 30: מעבר בודד במ"ט.

משמעות: המעבר העליון - אם אתה במצב q וקראת a, החלף את a ב-b, עבור ל-q' והזז את הראש ימינה. המעבר התחתון - עבור למצב q' והזז את הראש ימינה (בלי לכתוב תו חדש).

הגדרה. מכונת טיורינג (מ"ט) M היא שביעייה $M=\left(Q,\Sigma,\Gamma,\delta,q_0,q_{accept},q_{reject}\right)$, כאשר:

- Q - קבוצה סופית של מצבים.
- Σ - א"ב הקלט, $\sqcup\notin\Sigma$.
- Γ - א"ב הסרט, $\Sigma\cup\{\ \}\subseteq\Gamma$.
- q_0 - מצב התחלתי.
- q_{accept} - מצב מקבל.
- q_{reject} - מצב דוחה ($q_{accept}\neq q_{reject}$).
- פונקציית המעברים $\delta:Q\times\Gamma\to Q\times\Gamma\times\{L,R\}$:

$$\delta\left(q,a\right)=\left(q',b,L\right)$$

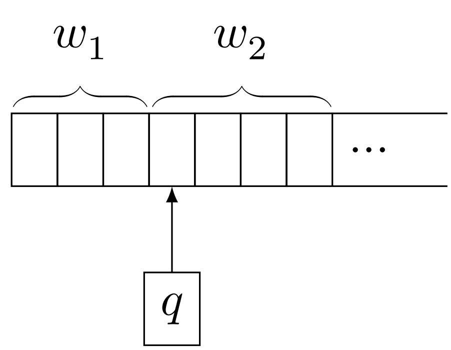
הערה. אם נמצאים בתא השמאלי ביותר של הסרט ורוצים לזוז שמאלה - נשארים במקום.

נרצה לדון במסלול החישוב של מ"ט M על מילה $w\in\Sigma^*$.

הגדרה. קונפיגורציה של מ"ט (תמונה רגעית של החישוב), מכילה:

- מצב נוכחי.
- תוכן הסרט (עד המקום ממנו יש רק רווחים).
- מיקום הראש הקורא.

באופן פורמלי, קונפיגורציה היא שלשה (w_1,q,w_2) כאשר $w_1,w_2\in\Gamma^*,q\in Q$.



איור 31: קונפיגורציה (w_1,q,w_2) בסרט.

המילה w_1 תכיל את התווים את הראש הקורא, ו-w_2 החל ממיקום הראש הקורא. q הוא המצב הנוכחי. כלומר, הסרט מכיל $w_1\circ w_2\circ\sqcup\circ\ldots$.

בהינתן מילה $w\in\Sigma^*$ ומ"ט M, <u>הקונפיגורציה ההתחלתית</u> של M על w היא (ε,q_0,w).

- הראש הקורא נמצא בתא השמאלי ביותר, נמצאים ב-q_0 ומימינו כתובה המילה w.

הגדרה. קונפיגורציות מיוחדות.

- קונפיגורציה בה המצב הוא $q_{accepct}$ תיקרא מקבלת.
- קונפיגורציה בה המצב הוא q_{reject} תיקרא דוחה.
- קונפיגורציה מקבלת או דחה תיקרא מסיימת.

הגדרה. נגדיר יחס $\vdash_M$ על קונפיגורציות, כך ש-$c_1 \vdash_M c_2$ אמ"מ ניתן לעבור בצעד אחד מ-c_1 ל-c_2.

דוגמה. יהיו $w_1, w_2 \in \Gamma^*, a_1, a_2 \in \Gamma, q \in Q$, ו-$c = (w_1 a_1, q, a_2 w_2)$.
נניח כי $\delta(q, a_2) = (q', b, R)$, ואז:

$$c \vdash_M (w_1 a_1 b, q', w_2)$$

הערה. אם c קונפיגורציה מסיימת, אזי $c \nvdash_M c'$ לכל c'.

הגדרה. מסלול החישוב של מ"ט M על מילה $w \in \Sigma^*$ הוא סדרת קונפיגורציות, כך שמתקיים:

- מתחילים מהקונפיגורציה ההתחלתית של M על w.
- לכל i: $c_i \vdash_M c_{i+1}$, כלומר:
  - $c_1 \vdash_M c_2 \vdash_M \cdots \vdash_M c$, כך ש-$c$ היא קונפיגורציה מסיימת.
  - $c_1 \vdash_M c_2 \vdash_M \cdots$ - לולאה אינסופית.

למ"ט, בהינתן קלט, יש 3 אפשרויות:

1. לעצור ולקבל (מגיעים ל-q_{accept}).
2. לעצור ולדחות (מגיעים ל-q_{reject}).
3. להיכנס ללולאה אינסופית.

הערה. אין קשר בין אורך המילה ומסלול החישוב.

1 שפות המתקבלות ע"י מ"ט

אינטואיטיבית, היינו רוצים שעל קלט בשפה נעצור ונקבל, ועל קלט לא בשפה נעזור ונדחה.

- עם זאת, קיים מצב שלישי - בו נכנסים ללולאה אינסופית.

הגדרה. נאמר שמ"ט M <u>מכריעה</u> שפה L אם:

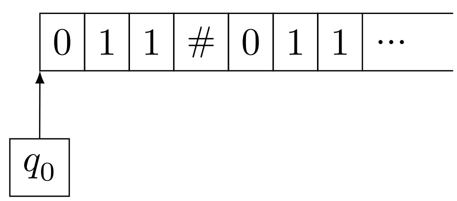
- לכל $w \in L$, מסלול החישוב של M על w מסתיים בקונפיגורציה מקבלת.
- לכל $w \notin L$, מסלול החישוב של M על w מסתיים בקונפיגורציה דוחה.

הערה. בפרט, אם M מכריעה שפה אז היא עוצרת על כל קלט.

הגדרה. נאמר שמ"ט M <u>מקבלת</u> שפה L אם:

- לכל $w \in L$, מסלול החישוב של M על w מסתיים בקונפיגורציה מקבלת.
- לכל $w \notin L$, מסלול החישוב של M על w אינו מסתיים בקונפיגורציה מקבלת (כלומר, נעצור ונדחה או ניכנס ללולאה אינסופית).

דוגמה. נסתכל על השפה $L = \left\{u\#u \mid u \in \{0,1\}^*\right\}$.
שפה זו אינה ח"ה.



איור 32: דוגמא לסרט ההתחלתי.

<u>הרעיון:</u>

- שמור את ערך התו הראשון במצב והחלף אותו ב-x.
- רוץ ימינה עד מציאת $\#$, ועקוף x-ים.
- קרא את התו הבא.
  - אם הוא שונה מערך התו הראשון, דחה.
  - אחרת, החלף את התו ב-x, ורוץ שמאלה עד התו הראשון (משמאל) שאינו x.

דוגמה. נסתכל על השפה $L = \left\{0^{2^n} \mid n \geq 0\right\}$, תחת הא"ב $\Sigma = \{0\}$.
ניתן להראות כי השפה אינה ח"ה, נבנה מ"ט M שמכריעה את L.
רעיון: נרצה לחלק ב-2 בכל פעם, עד שנגיע לתו בודד.

- נוסיף לא"ב תו x, ומילה היא רצף של 0-ים ו-x-ים.
  - כדי לבדוק איזו מילה מקודדת, נתעלם מה-x-ים.
- נבצע כל חלוקה ב-2 ע"י הפיכה של כל תו 0 שני ב-x.

– אם סיימנו ב-x, המספר זוגי - נרצה לחזור על התהליך עם המספר שנשאר.
– אחרת, המספר אי-זוגי - נדחה, אלא אם המספר הוא 1 ונקבל.

נשתמש בשני מצבים בשביל לחשב את זוגיות מספר ה-0-ים.

- אם בסיום המילה אנחנו במצב האי-זוגי.

– בנוסף, נטפל בתו הראשון באופן מיוחד (נחליף אותו בתו מיוחד $0'$, שמסמן את תחילת הסרט).
– אם בסיום קריאת המילה הגענו למילה שה-0 היחיד בה הוא $0'$, נקבל (חילקנו ב-2 עד שהגענו ל-1, ולכן המספר הוא חזקה של 2).
– אחרת, נדחה.

- אחרת, בסיום המילה אנחנו במצב הזוגי - נמשיך.

נסתכל על המילה 0^{12}. לאחר המעבר הראשון המילה תיראה כך:

$$000000000000 \Rightarrow 0'x0x0x0x0x0x$$

מספר ה-0-ים זוגי - נמשיך (כעת מיוצג המספר 6).

$$0'x0x0x0x0x0x \Rightarrow 0'xxx0xxx0xxx$$

מספר ה-0-ים זוגי, כעת מיוצג המספר 3.

$$0'xxx0xxx0xxx \Rightarrow 0'xxxxxxx0xxx$$

מספר ה-0-ים אינו זוגי (ה-0 האחרון אינו אדום) $\Longleftarrow$ נדחה.

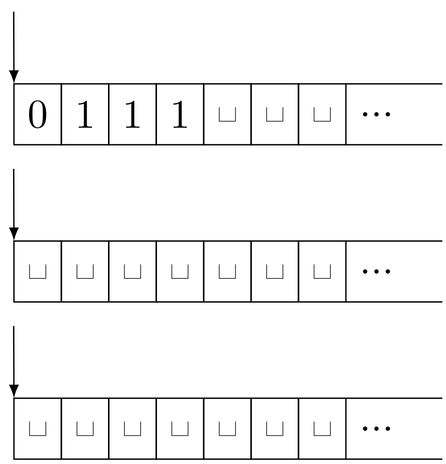
2 התזה של צרץ' וטיורינג

- כל מה שניתן לחישוב, ניתן לחישוב ע"י מ"ט.
  - כל מה שניתן לחישוב ע"י כל מחשב שאנו מכירים.
  - כל מה שניתן לחישוב ע"י כל מחשב שניתן לדמיין.
  - כל מה שניתן לתאר אינטואיטיבית כאלגוריתם.
- אינטואיציה - למה זה נכון?

משפט. קיים קומפיילר שלוקח תכנית מחשב בשפת C, ומתרגם אותה לטבלת מעבר של מ"ט שמקבלת/מכריעה את אותה השפה.

- באופן אמפירי, התזה בת יותר מ-80 שנה.
- חיזוק של מ"ט ע"י תוספת של רכיבים נוספים, לא מוסיף כוח חישוב.
  - למשל, מכונת טיורינג עם הרבה סרטים.

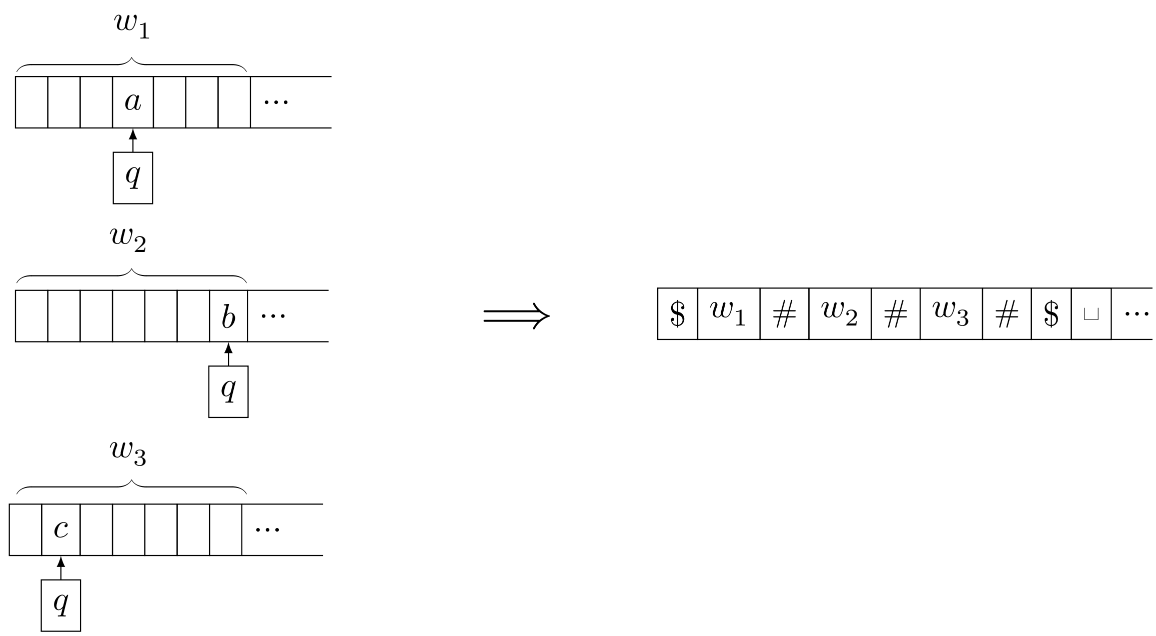
הגדרה. מ"ט k-סרטית היא מ"ט עם k סרטים, ו-k ראשי סרט קוראים/כותבים.
בהינתן המצב וכל אחד מהתווים, עוברים למצב הבא וכותבים/זזים בכל אחד מהסרטים.
הקלט בהתחלה בסרט הראשון.



איור 33: סרטים של מכונה 3-סרטית בהתחלה.

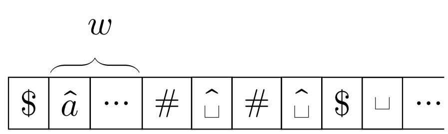
משפט. כל שפה L המתקבלת ע"י מ"ט k-סרטית מתקבלת ע"י מכונת טיורינג עם סרט בודד.

הוכחה. בהינתן מ"ט M k-סרטית שמקבל שפה L, נבנה מ"ט M' חד-סרטית שתקבל את L.
הרעיון: נקודד קונפיגורציה של מ"ט k-סרטית על סרט יחיד.
למשל, עבור מ"ט 3-סרטית.



איור 34: משמאל, מצב כלשהו של 3 הסרטים במ"ט. מימין, קידוד 3 הסרטים לסרט יחיד.
נוסיף לא"ב Γ' של M' את התווים המיוחדים $\$, \#$. בנוסף, לכל תו $a \in \Sigma$ נוסיף ל-Γ' תו מיוחד $\hat{a}$, שיסמל את מיקום הראש הקורא בכל אחד מהסרטים.

<u>סימולציה:</u>



איור 35: קידוד הקונפיגורציה ההתחלתית ב-M'.
באתחול, כאשר M מקבלת קלט w, קידוד הקונפיגורציה ב-M' יהיה (נסמן ב-a את התו הראשון ב-w)

<u>הרעיון:</u> כל צעד חישוב של ה-k סרטית יסומלץ ע"י (הרבה) צעדים של החד-סרטית. נשמור בעזרת המצב את התווים שהראש הקורא מצביע עליהם בכל אחד מהסרטים.

- נסרוק את הסרט משמאל לימין.
  - כאשר נראה תו מסומן בכובע, נשמור את זה בעזרת המצב.
- בסוף הסריקה אנו נמצאים ב-$\$$ הימני, ובמצב יש את התווים שה-k-סרטית רואה בראשים הקוראים.

אם המ"ט ה-k סרטית מתכוונת לעבור ממצב q למצב q' כאשר היא רואה a, b, c, לכתוב במקום a', b', c', ולזוז L, R, L, אז נשמור בעזרת במצב את q', a', b', c', L, R, L.

- כעת, נסרוק את הסרט מימין לשמאל ונבצע את השינויים המתבקשים.
  - למשל, כשנגיע לתו עם כובע, נוריד את הכובע ונזוז ע"פ המצב.
- אם המ"ט ה-k-סרטית רוצה להזיז את הראש הקורא שלה ימינה, לתו שלא מופיע על הסרט של החד-סרטית, נבצע פעולת shift right.
  - כלומר, נדחוף את כל המחרוזת מימין למקום ההכנסה ימינה, ובמקום הפנוי שנוצר נדחוף את התו החדש.

כך המ"ט M' שקול ל-M. □

הערה. במה מ"ט נבדלת ממחשב?

- למ"ט כמות זיכרון בלתי חסומה - סרט אינסופי.
  - עם זאת, באותה המידה ניתן לחבר למחשב התקני זיכרון ללא הגבלה, וכך להגיע לכמות זיכרון בלתי חסומה גם במחשב.
- מודל הזיכרון של מ"ט הוא סדרתי.
  - כדי להגיע לתא זיכרון מסוים, הראש צריך לעבור על פני כל התאים שבין המיקום הנוכחי שלו והתא אליו הוא רוצה להגיע.
  - לעומת זאת, זיכרון של מחשב מבוסס על גישה אקראית (RAM). כדי להגיע לתא זיכרון, פונים ישירות אליו.
- למעבד ישנם רגיסטרים שבהם ניתן לשמור מידע זמני, בעוד שמ"ט קיים הסרט בלבד.
  - המחשב שקול למודל k-סרטי, בו כל סרט מכיל רגיסטר. על כן, הדבר לא נבדל ממ"ט חד-סרטית.
- מעבד מסוגל לבצע פקודות אריתמטיות-לוגיות, בעוד שמ"ט יודעת רק לקרוא ולכתוב תו בודד בסרט.
  - ניתן לממש כל פעולה אריתמטית-לוגית באמצעות מ"ט, ע"י פירוק לתתי פעולות (ראינו דוגמאות פשוטות, כמו השוואה וחלוקה ב-2).

ההבדלים לעיל חסרי חשיבות מבחינת כוח החישוב.

בנוסף, נייצג RAM במ"ט באופן הבא:

- הזיכרון כולו יסומלץ ע"י סרט בודד, שתוכנו מהצורה הבא:

$$@A_1\#C_1\$A_2\#C_2\$A_3\#C_3\cdots$$

  - $@, \$, \#$ הם תווים מיוחדים.
    * A_i, C_i הם מספרים שלמים (מקודדים בינארית).
    * A_i מייצג כתובת.
    * C_i מייצג תוכן.
    * הזוג $A_i\#C_i$ פירושו בתא בכתובת A_i מאוחסן C_i.
  - כמו כך, נאפשר לשנות את ה-$\#$ ב-x, והסימון A_ixC_i פירושו: C_i מכיל זבל.
  - קריאה מהכתובת A_i תתבצע כך:
    * סורקים את הסרט משמאל לימין, ונחפש את המקום הראשון בו כתוב $A_i\#$ (כאשר יש $\$$ או $@$ לפני A_i).
    * מעתיקים את התוכן של מה שמופיע אחרי ה-$\#$ ועד ה-$\$$ הבא.
    * אם לא נמצא $A_i\#$, זה אומר שטרם נכתב לכתובת A_i תוכן, ונחזיר 0.
  - כתיבה לתא A_i תתבצע כך:
    * סורקים את הסרט משמאל לימין. אם מתגלה $A_i\#$, משנים את ה-$\#$ ל-x.
    * כשמגיעים לקצה הסרט כותבים $A_i\#C_i\$$, כאשר C_i הוא התוכן המיועד.

מנקודה זו והלאה נשתכנע כי מ"ט ומחשב שקולים.

3 מכונת טיורינג לא-דטרמיניסטית

הגדרה. מ"ט לא-דטרמיניסטית M היא שביעייה $M = \left(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject}\right)$, כאשר:

- Q - קבוצה סופית של מצבים.
- Σ - א"ב הקלט, $\Sigma \notin$רווח(טודו).
- Γ - א"ב הסרט, $\Sigma \cup \{ \} \subseteq \Gamma$.
- q_0 - מצב התחלתי.
- q_{accept} - מצב מקבל.
- q_{reject} - מצב דוחה ($q_{accept} \neq q_{reject}$).
- פונקציית המעברים $\delta : Q \times \Gamma \rightarrow P\left(Q \times \Gamma \times \{L, R\}\right)$:

$$\delta\left(q, a\right) = \left\{\left(q', b, L\right), \left(q'', a, R\right), \ldots\right\}$$

- במ"ט דטרמיניסטית היה מסלול חישוב יחיד, החל מהקונפיגורציה ההתחלתית.
  - בסופו הגענו לקונפיגורציה מקבלת/דוחה, או לולאה אינסופית.
- במ"ט לא-דטרמיניסטית קיימים מספר מסלולי חישוב, החל מהקונפיגורציה ההתחלתית ניתן לעבור למספר קונפיגורציות שונות.
  - ייתכן שחלק ממסלולי החישוב יסתיימו בקונפיגורציה מקבלת, חלק יסתיימו בדוחה, וחלק בלולאה אינסופית.

הגדרה. נגדיר את שפת המ"ט, $L\left(M\right)$, כך:

$$L\left(M\right) = \left\{w \in \Sigma^* \mid \text{קיים מסלול חישוב של } M \text{ על } w \text{ שמגיע לקונפיגורציה מקבלת}\right\}$$

משפט. כל שפה שמתקבלת ע"י מ"ט לא-דטרמיניסטית, מתקבלת גם ע"י מ"ט דטרמיניסטית.

הערה. קשיים להוכחת המשפט.

1. למכונה הלא-דטרמיניסטית מסלולי חישוב אינסופיים.
2. צריך לתכנת את המ"ט הדטרמיניסטית.

הערה. מנקודה זו, לאור התזה של צרץ' וטיורינג, ניתן לתאר מ"ט באמצעות פסאודו-קוד בלבד.

הוכחה. נוכיח את המשפט.

- נסתכל על עץ הקונפיגורציות של המ"ט הלא-דטרמיניסטי, ונבצע עליו חיפוש לרוחב.
- בהינתן מילה w, אם $w \in L(M)$ יש לעצור ולקבל.
- אחרת, מותר להיכנס ללולאה אינסופית.

<u>האלגוריתם:</u> טיול על עץ הקונפיגורציות, עם BFS (DFS לא יעבוד, מחשש למסלול אינסופי).

1. אתחל $i \leftarrow 1$.
2. סרוק את כל המסלולים מאורך i בעץ הקונפיגורציות.
   (א) אם באחד מהם יש קונפיגורציה מקבלת, עצור וקבל.
3. אחרת, $i \leftarrow i+1$ וחזור ל-2.

נשים לב כי אם $w \in L(M)$, קיים מסלול חישוב על w שמסתיים בקונפיגורציה מתקבלת - נסמן את אורך המסלול הקצר ביותר כזה ב-i^*.

- הלולאה תגיע ל-$i = i^*$.
  - לפני כן התבצע חישוב סופי בלבד בכל שלב, במספר סופי של שלבים.
  - כאשר $i = i^*$, נמצא את מסלול החישוב המקבל - נעצור ונקבל.

אם $w \notin L(M)$, לא קיים מסלול חישוב שמסתיים בקונפיגורציה מקבלת. נפריד למקרים:

- ניכנס לולאה אינסופית, וכך המילה לא תתקבל ע"י המכונה.
- כל המסלולים הם סופיים ומסתיימים בקונפיגורציה דוחה, וכך נסיים ללא קבלה ונדחה.

מכאן, סיימנו את ההוכחה. □

הגדרה. מ"ט M <u>מכריעה</u> שפה L אם לכל w:
$w \in L \Longleftarrow$ כל מסלולי החישוב מסתיימים, וקיים
מסלול שמסתיים במצב מקבל.
$w \notin L \Longleftarrow$ כל מסלולי החישוב של M על w מסתיימים במצב דוחה.

משפט. כל השפה המוכרעת ע"י מ"ט לא-דטרמיניניסטית מוכרעת ע"י מ"ט דטרמיניסטית.

הוכחה. באופן דומה להוכחה הקודמת, נסרוק את עץ הקונפיגורציות של המ"ט הלא-דטרמיניסיטית.
המילה תתקבל אמ"מ אם מצאנו מסלול שמסתיים במצב מקבל (יש לוודא שעוצרים בסוף).
כאן ניתן להשתמש גם ב-DFS, מאחר וכל המסלולים סופיים. □

הגדרה. שפה L תיקרא <u>ניתנת לקבלה</u> אם קיימת מ"ט שמקבלת אותה.

הגדרה. שפה L תיקרא <u>כריעה</u> אם קיימת מ"ט שמכריעה אותה.

משפט. כל שפה כריעה היא ניתנת לקבלה.

הוכחה. נובע באופן ישיר מההגדרות - כריעה היא מקרה פרטי של קבלה. □

איור 36: היררכיית המחלקות השונות

3.1 סגירויות

משפט. אם L שפה כריעה אז גם $\overline{L}$ כריעה.

הוכחה. מאחר ו-L כריעה, קיימת מ"ט שמכריעה אותה.
נהפוך במ"ט את המצבים q_{accept} ו-q_{reject}, ונקבל מ"ט שמכריעה את $\overline{L}$. □

הערה. ההוכחה נכשלת עבור שפות <u>ניתנות לקבלה</u>.
אם ב-L מילים שלא מתקבלות כתוצאה מלולאה אינסופית, גם לאחר החלפת המצבים ניכנס ללולאה אינסופית והמילים לא יתקבלו ב-$\overline{L}$.

משפט. אם L_1, L_2 שפות כריעות אז גם $L_1 \cap L_2$ כריעה.

הוכחה. נבנה מ"ט שמריצה את המכונה של L_1 על הקלט, ואחר כך את של L_2 על הקלט.
נקבל אמ"מ הקלט התקבל בשתיהן. □

משפט. אם L_1, L_2 שפות ניתנות לקבלה אז גם $L_1 \cap L_2$ ניתנת לקבלה.

הוכחה. נבנה מ"ט שמריצה את המכונה של L_1 על הקלט, ואחר כך את של L_2 על הקלט.
נקבל אמ"מ הקלט התקבל (ולכן בהכרח המכונה נעצרה) בשתיהן. □

משפט. אם L_1, L_2 שפות כריעות אז גם $L_1 \cup L_2$ כריעה.

הוכחה. נבנה מ"ט שמריצה את המכונה של L_1 על הקלט, ואחר כך את של L_2 על הקלט.
נדחה אמ"מ הקלט נדחה בשתיהן. □

משפט. אם L_1, L_2 שפות ניתנות לקבלה אז גם $L_1 \cup L_2$ ניתנת לקבלה.

הוכחה. נבנה מ"ט שמריצה את שתי המכונות של L_1 ו-L_2 במקביל, ע"י מ"ט דו-סרטית.
נקבל אם באחת מהמכונות הקלט התקבל. □

סגירות / פעולה	משלים	חיתוך	איחוד
שפות ניתנות לקבלה	X	$\checkmark$	$\checkmark$
שפות כריעות	$\checkmark$	$\checkmark$	$\checkmark$

טבלה 7: סגירויות של שפות ניתנות לקבלה וכריעות

3.2 מכונת טיורינג אוניברסלית

מ"ט היא שביעייה $M = \left(Q, \Sigma, \Gamma, \delta, q, q_{accept}, q_{reject}\right)$, וניתן לקודד את השביעייה ע"י מחרוזות מעל א"ב סופי $\Sigma = \{0,1\}$.

נקודד מ"ט M ע"י מחרוזת $\langle M\rangle$, ואת הקידוד של $\langle M\rangle$ ניתן בתור קלט למ"ט אחרת.

משפט. קיימת מ"ט U כך שעל כל קלט מהצורה $\langle M, w\rangle$, כאשר M מ"ט ו-w מחרוזת, U מסמלצת את פעולת M על w.

כלומר, אם M מקבלת את w אז U מקבלת את הקלט $\langle M, w\rangle$, אם M דוחה את w, אז U דוחה את $\langle M, w\rangle$, ואם M נכנסת ללולאה אינסופית על w אז U נכנסת ללולאה אינסופית על $\langle M, w\rangle$.

U תיקרא מ"ט אוניברסלית.

הערה. רעיון ההוכחה: ניתן לקודד את התיאור של M ולבצע מתוך הקידוד.

הערה. מ"ט יכולה בהינתן אס"ד, אסל"ד, דקדוק או א"מ ומילה, להכריע האם המילה מתקבלת.

נשים לב כי U אינה מכונה מכריעה: קיים קלט $\langle M, w\rangle$ כך ש-M לא עוצרת על w, ואז U לא תעצור על הקלט $\langle M, w\rangle$.

הגדרה. נגדיר את השפות $ACCEPT, HALT$ באופן הבא:

$$L\left(U\right) = ACCEPT = \{\langle M, w\rangle \mid w \in L\left(M\right) \text{-ו } \text{מילה } w \text{ ,מ"ט } M\}$$

$$HALT = \{\langle M, w\rangle \mid w \text{ עוצרת על } M\text{-ו } \text{מילה } w \text{ ,מ"ט } M\}$$

נשים לב כי $ACCEPT \subseteq HALT$.

3.3 בעיית העצירה $HALT$

קלט: מ"ט M ומחרוזת w.

פלט: האם M עוצרת על w (כלומר $\langle M, w\rangle \in HALT$?)

הערה. האם $HALT$ ניתנת לקבלה? כן - לסמלץ.

האם $HALT$ כריעה?

טענה. $HALT$ ניתנת לקבלה.

הוכחה. נבנה מ"ט T שמקבלת את $HALT$. בהינתן קלט $\langle M, w\rangle$:

- נבדוק שקידוד תקין, אחרת נדחה.
- אחר כך, נריץ את U על $\langle M, w\rangle$ בשינוי אחד:
  - כאשר U רוצה להיכנס ל-q_{reject}, T תיכנס ל-q_{accept}.

נקבל כל קלט שייעצר, בין אם נדחה או מתקבל במקור. לא נקבל כל קלט שנכנס ללולאה אינסופית. כך, T מקבלת את $HALT$. □

3.4 בעיית ההכרעה

הגדרה. נגדיר את השפה $NON-EMPTY$ כך:

$$NON-EMPTY = \{\langle M \rangle \mid L(M) \neq \emptyset\}$$

<u>קלט:</u> מ"ט M.

<u>פלט:</u> האם קיים קלט w כך ש-M מקבלת את w.

טענה. $NON-EMPTY$ ניתנת לקבלה.

דוגמה. נתחיל משפה קלה יותר:

$$L = \{\langle M \rangle \mid \exists w : |w| \leq 1000 \wedge w \in L(M)\}$$

נסתכל על האלגוריתם הבא:

- נבנה מכונה שתקבל את השפה $L(M)$.
- עבור $i = 1, \ldots, 1000$:
  - עבור על כל המילים w מאורך i.
    * הרץ את M על w.
    * אם M קיבלה את w, עצור וקבל.

האלגוריתם יכול לא לעצור. בעיה:
יכול להיות שלא נעצור על קלטים שהם כן בשפה.
למשל, ייתכן שיש מילה w מאורך 7 כך ש-$w \in L(M)$, אך יש מילה מאורך 6 שעליה M נכנסת ללולאה אינסופית, ולכן ניתקע ולא נתחיל להריץ את M על w.

<u>נשפר את האלגוריתם:</u> $i = 0$, עבור על כל המילים w כך ש-$|w| \leq 1000$.

- הרץ את M על w במשך i צעדים.
- אם M עצרה וקיבלה, נעצור ונקבל.
- אחרת, $i \leftarrow i + 1$.

הערה. טכניקה זו נקראת <u>הרצה מבוקרת:</u>

מריצים את M על w מס' מוגבל של צעדים בכל פעם.

<u>כלל זהב:</u> אל תריץ באופן לא מבוקר מ"ט שהרמת מהרחוב (כי עשויה להיתקע בלולאה אינסופית).

הוכחה. נוכיח כי המכונה מקבלת את L.

- אם $\langle M \rangle \in L$ אז האלגוריתם עוצר ומקבל:
  - קיימת מילה w כך ש-$|w| \leq 1000$ ו-M מקבלת את w (וכך בפרט עוצרת על w).

- נסמן ב-i^* את מספר הצעדים של M על w עד שעצרה.
- נשים לב שכל שלב של האלגוריתם בפני עצמו עוצר.
  * כך, אם w המילה הכי קצרה ש-M מקבלת אז נקדם את i בלולאה עד ש-$i = i^*$.
  * אז, עוברים על כל המילים מאורך לכל היותר 1000 (וביניהן w).
  * מסמלצים את M על w למשך $i = i^*$ צעדים, וכך w תתקבל ו-M תיעצר ותתקבל.
  * לבסוף, נקבל.

- אם $\langle M\rangle \notin L$, אז M לא מקבלת אף מילה מאורך לכל היותר 1000.
  - במקרה זה נכנסים ללולאה אינסופית, וכך המכונה לא מקבלת את $\langle M\rangle$.

כך, קיבלנו כי המכונה מקבלת את L. $\square$

כעת, נסתכל על $NON - EMPTY$.
נשנה את האלגוריתם:

- $i = 0$.
- עבור על כל המילים w כך ש-$|w| \leq i$.
  - הרץ את M על w במשך i צעדים.
  - אם M עצרה וקיבלה, נעצור ונקבל.
  - אחרת, $i \leftarrow i + 1$.

טענה. האלגוריתם מקבל את $NOT - EMPTY$.

הוכחה. יהי $\langle M\rangle$ קלט לאלגוריתם.

- אם $\langle M\rangle \notin NOT - EMPTY$ אז M לא מקבלת אף קלט.
  - האלגוריתם עוצר ומקבל אמ"מ יש קלט ש-M קיבלה, ולכן האלגוריתם נכנס ללולאה אינסופית.
- אם $\langle M\rangle \in NOT - EMPTY$ אז קיימת מילה w כך ש-$w \in L(M)$.
  - נסמן $a = |w|$, ואת מספר הצעדים של M על w ב-b.
  - נגדיר $i^* = \max\{a, b\}$.
  - נשים לב כי עבור כל i, האלגוריתם לא נכנס ללולאה אינסופית.
    * לכן, בהכרח נגיע במספר צעדים סופי ל-$i = i^*$.
    * בנקודת זמן זו, עוברים על כל המילים w מאורך לכל היותר i^* (בפרט על מילים מאורך a), ומריצים את M ל-$b \geq i^*$ צעדים.
    * כך, נראה כי M מקבלת את w ב-b בצעדים, וכך האלגוריתם יעצור ויקבל.

$\square$

משפט. אם $HALT$ כריעה אז $ACCEPT$ כריעה.

הוכחה. נניח כי $HALT$ כריעה.
אזי, קיימת מ"ט M_{HALT} שמכריעה אותה. נבנה מ"ט M_{ACCEPT} שמכריעה את $ACCEPT$.
בהינתן קלט $\langle M, w\rangle$:

- הרץ את M_{HALT} על $\langle M,w\rangle$.
  - אם ענתה לא, נדחה.
  - אחרת, הרץ את M על w והחזר בהתאם (החישוב בטוח יסתיים כי $HALT$ החזירה ש-M עוצרת על w).

$\square$

מסקנה. אם $ACCEPT$ לא כריעה אז $HALT$ לא כריעה.

נרצה טכניקה שתראה לנו ששפות לא כריעות.

3.5 שיטת הלכסון של קנטור

משפט. אין פונקציה $f:\mathbb{N}\to\mathbb{P}\left(\mathbb{N}\right)$ שהיא על.

הוכחה. תהי $f:\mathbb{N}\to\mathbb{P}\left(\mathbb{N}\right)$, ונראה כי קיימת $S\in P\left(\mathbb{N}\right)$ כך שלכל i מתקיים $f\left(i\right)\neq S$.

i \ $f\left(i\right)$	1	2	3	4	5	$\cdots$
1	$\checkmark$	X	$\checkmark$	$\checkmark$	X	
2	$\checkmark$	X	$\checkmark$	$\checkmark$	$\checkmark$	
3	$\checkmark$	X	$\checkmark$	X	$\checkmark$	
4	X	$\checkmark$	$\checkmark$	$\checkmark$	X	
5	$\checkmark$	X	$\checkmark$	X	X	
$\vdots$						$\ddots$

טבלה 8: ייצוג הפונקציה f.

אם $f\left(1\right)=\{1,3,4\}$, למשל, נסמן $\checkmark$ ב-1, 3 ו-4.

נסתכל על האלכסון המסומן בטבלה, ונבנה שתי קבוצות:

1. $D=\{i\mid i\in f\left(i\right)\}$ (כל ה-$\checkmark$-ים באלכסון).
2. $S=\overline{D}=\{i\mid i\notin f\left(i\right)\}$ (כל ה-X-ים באלכסון).

הקבוצה $S=\overline{D}$ שונה מכל השורות: לכל i מתקיים $f\left(i\right)\neq\overline{D}$, שתי הקבוצות אינן מסכימות על האיבר ה-i. $\square$

נרצה להשתמש בטיעון לכסון דומה כדי להראות דוגמא לשפות שאינן כריעות / אינן ניתנות לקבלה.

<u>שמורה:</u> כל מחרוזת היא קידוד של איזושהי מ"ט.

ניתן לעשות זאת ע"י כך שכל מחרוזת שאיננה מקודדת למ"ט, נחשוב עליה כמחרוזת שמקודדת מ"ט שמייד עוצרת ודוחה.

מ"ט \ קלט	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\cdots$
M_1	$\checkmark$	X	$\checkmark$	
M_2	$\checkmark$	X	$\checkmark$	
M_3	$\checkmark$	X	$\checkmark$	
$\vdots$				$\ddots$

טבלה 9: טבלת מ"ט/קלטים

במקום ה-$[i,j]$ נכתוב $\checkmark$ אם M_i מקבלת את הקלט $\langle M_j\rangle$, ו-X אחרת. השורה ה-i בשפה מגדירה את השפה $L(M_i)$.

נסתכל על האלכסון, ונגדיר שתי שפות באופן הבא:

1. $D=\{\langle M\rangle \mid \langle M\rangle\in L(M)\}$ (כל ה-$\checkmark$-ים באלכסון).
2. $\overline{D}=\{\langle M\rangle \mid \langle M\rangle\notin L(M)\}$ (כל ה-X-ים באלכסון).

משפט. $\overline{D}$ לא ניתנת לקבלה.

הוכחה. נניח בשלילה כי $\overline{D}$ ניתנת לקבלה. כלומר, קיימת מ"ט M שמקבלת אותה.
נסתכל על הקלט של M עם הקלט $\langle M\rangle$:

1. אם $\langle M\rangle\in L(M)$, אז $\langle M\rangle\in D$ וכך $\langle M\rangle\notin\overline{D}$.
   (א) בסתירה לכך ש-$L(M)=\overline{D}$ ו-$\langle M\rangle\in L(M)$ וגם $\langle M\rangle\notin\overline{D}=L(M)$).
2. אם $\langle M\rangle\notin L(M)$, אז $\langle M\rangle\notin D$ וכך $\langle M\rangle\in\overline{D}$.
   (א) סתירה, מאחר ו-$\overline{D}=L(M)$, $\langle M\rangle\in\overline{D}$ וגם $\langle M\rangle\notin L(M)$.

בכל מקרה, הגענו לסתירה וכך $\overline{D}$ לא ניתנת לקבלה. □

טענה. אם $ACCEPT$ כריעה אז $\overline{D}$ כריעה.

מסקנה. $ACCEPT$ לא כריעה.
אם $\overline{D}$ לא כריעה אז $ACCEPT$ לא כריעה.
$\overline{D}$ לא ניתנת לקבלה, ובפרט לא כריעה.

מסקנה. $HALT$ לא כריעה.
ראינו כי אם $ACCEPT$ לא כריעה אז $HALT$ לא כריעה.

הוכחה. נוכיח כי אם $ACCEPT$ כריעה אז $\overline{D}$ כריעה.
תהי M_{ACCEPT} מ"ט שמכריעה את $ACCEPT$.
נבנה מ"ט $M_{\overline{D}}$ שמכריעה את $\overline{D}$:

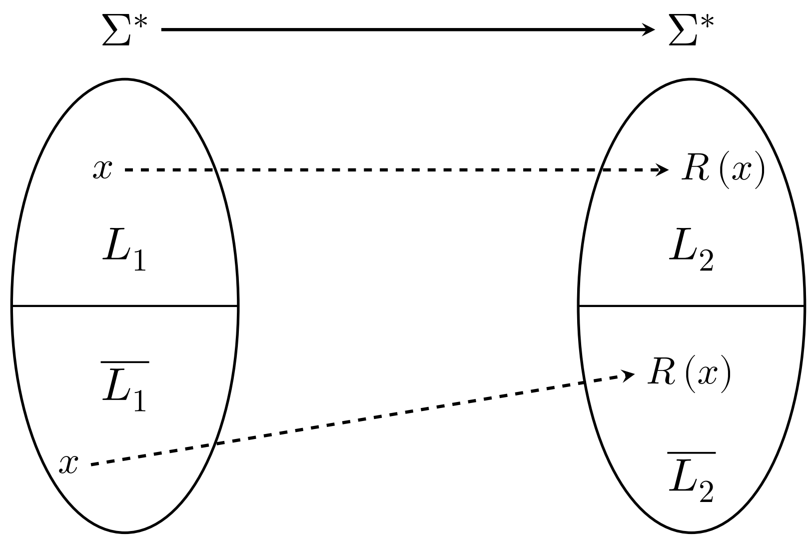
- בהינתן קלט $\langle M\rangle$, נריץ את M_{ACCEPT} על $\langle M,\langle M\rangle\rangle$.
- M_{ACCEPT} תעצור, ותענה האם M מקבלת את $\langle M\rangle$.
  - אם כן, נכריע ב-$\overline{D}$ את ההפך מהכרעת M_{ACCEPT}.

□

הגדרה. <u>רדוקציית מיפוי</u> R מ-A ל-B היא מ"ט שפועלת לכל קלט ומתקיים:

$$w \in A \Longleftrightarrow R(w) \in B$$

מסמנים זאת $A \leq_m B$.



איור 37: רדוקציית המיפוי $R : \Sigma^* \to \Sigma^*$
הפונקציה ממפה כל $x \in L_1$ ל-$R(x) \in L_2$, וכל $x \notin L_1$ ל-$R(x) \notin L_2$.

מסקנה. מסקנות מרדוקציית מיפוי - אם $A \leq_m B$ אז:

1. אם B כריעה אז A כריעה (מכאן, אם A לא כריעה אז B לא כריעה).
2. אם B ניתנת לקבלה אז A ניתנת לקבלה (מכאן, אם A לא ניתנת לקבלה אז B לא ניתנת לקבלה).

קלט	כריעה	ניתנת לקבלה
$ACCEPT = \{\langle M, w\rangle \mid w \in L(M)\}$	X	✓
$HALT = \{\langle M, w\rangle \mid$ עוצרת על w $M\}$	X	✓
$NOT-EMPTY = \{\langle M\rangle \mid L(M) \neq \emptyset\}$	X	✓
$EMPTY = \{\langle M\rangle \mid L(M) = \emptyset\}$	X	X
$EQUAL = \{\langle M_1, M_2\rangle \mid L(M_1) = L(M_2)\}$	X	X

טבלה 10: כריעות וניתנות לקבלה של שפות

טענה. $EMPTY \leq_m EQUAL$

הוכחה. נגדיר רדוקציית מיפוי R באופן הבא:

$$R\left(\langle M\rangle\right) = \langle M, M'\rangle$$

כך ש-M' דוחה כל קלט.

$$\langle M\rangle \in EMPTY \iff L\left(M\right) = \emptyset = L\left(M'\right) \iff \langle M, M'\rangle \in EQUAL$$

□

הערה. הפונקציה R אינה חח"ע ואינה על, R לא מריצה את M ומתייחסת אליה בתור מחרוזת בלבד.

טענה. $ACCEPT \leq_m NOT - EMPTY$

הוכחה. נגדיר רדוקציית מיפוי R באופן הבא: $R\left(x\right) = \langle M'\rangle$, כאשר M' מוגדרת ע"י:

- אם x מהצורה $\langle M, w\rangle$, עבור כל קלט הרץ את M על w והחזר כמו M.
  - אחרת, M' תדחה עבור כל קלט (כך, $\langle M'\rangle \notin NOT - EMPTY$)

נראה כי $x \in ACCEPT \iff R\left(x\right) \in NOT - EMPTY$:

- אם $x \in ACCEPT$, אזי הוא מהצורה $\langle M, w\rangle$ ו-$w \in L\left(M\right)$.
  - מכאן, M' תקבל על כל קלט ולכן $\langle M'\rangle \in NOT - EMPTY$.
- אם $x \notin ACCEPT$, אזי הוא לא מהצורה $\langle M, w\rangle$ או ש-$w \notin L\left(M\right)$.
  - בכל מקרה, M' דוחה על כל קלט ולכן $M' \notin NOT - EMPTY$.

מכאן, $x \in ACCEPT \iff R\left(x\right) \in NOT - EMPTY$ ולכן $ACCEPT \leq_m NOT - EMPTY$.

□

דוגמה. שפות מהצורה $L = \{\langle M\rangle \mid$ מ"ט ו-$L\left(M\right)$ מקיימת תכונה כלשהי $M\}$

$$L = \{\langle M\rangle \mid w \in L\left(M\right)\}$$

$$REGULAR = \{\langle M\rangle \mid M \text{ מ"ט ו-} L\left(M\right) \text{ רגולרית}\}$$

$$NOT - REGULAR = \{\langle M\rangle \mid M \text{ מ"ט ו-} L\left(M\right) \text{ לא רגולרית}\}$$

$$NOT - CFL = \{\langle M\rangle \mid M \text{ מ"ט ו-} L\left(M\right) \text{ לא ח"ה}\}$$

טענה. $REGULAR$ אינה כריעה.

הערה. $NOT-REGULAR=\overline{REGULAR}$.
מכאן, $REGULAR$ כריעה אמ"מ $NOT-REGULAR$ כריעה - מספיק להוכיח כי $NOT-REGULAR$ לא כריעה.

טענה. $ACCEPT\leq_{m}NOT-REGULAR$.

הוכחה. נבנה רדוקציית מיפוי מ-$ACCEPT$ ל-$NOT-REGULAR$.

- תחילה, $\emptyset\in REGULAR$ ו-$\{a^{n}b^{n}\mid n\geq0\}\in NOT-REGULAR$.
- בנוסף, קיימת מ"ט T כך ש-$L\left(T\right)=\{a^{n}b^{n}\mid n\geq0\}$ (מכאן $\langle T\rangle\in NOT-REGULAR$)
- נבנה רדוקציית מיפוי R כך ש-$R\left(\langle M,w\rangle\right)=\langle M'\rangle$, ו-$M'\left(x\right)$ מוגדר להיות:
  - הרץ את M על w.
    - אם M דוחה, דחה.
  - הרץ את T על x והחזר כמוה.
- אם $w\in L\left(M\right)$ אז $L\left(M'\right)=L\left(T\right)$.
  - לכל קלט x, M תעצור על w ולאחר מכן T תריץ את x ויוחזר כמוה.
- מכאן, אם $w\in L\left(M\right)$ אז $L\left(M'\right)\in NOT-REGULAR$.
- אם $w\notin L\left(M\right)$ אז $L\left(M'\right)=\emptyset$.
  - או ש-M דוחה את w, וכך M' תדחה ללא תלות ב-x.
  - או ש-M נכנסת ללולאה אינסופית על w, וכך M' תיכנס ללולאה אינסופית לא תלות ב-x.
  - בכל מקרה, אף מילה לא תתקבל.
- מכאן, אם $w\notin L\left(M\right)$ אז $L\left(M'\right)\in REGULAR$.

בסופו של דבר, קיבלנו כי $\langle M,w\rangle\in ACCEPT$ אמ"מ $\langle M'\rangle\in NOT-REGULAR$, וכך הרדוקציה עובדת ומתקיים $ACCEPT\leq_{m}NOT-REGULAR$. □

טענה. $ACCEPT\leq_{m}NOT-CFL$.

הוכחה. באופן זהה להוכחת $ACCEPT\leq_{m}NOT-REGULAR$, כאשר נבחר את T להיות מ"ט כך ש-$L\left(T\right)=\{a^{n}b^{n}c^{n}\mid n\geq0\}$ (או כל שפה ח"ה). □

4 שפות שלא ניתנות לקבלה

4.1 משפט רייס

שאלה: בהינתן $L = \{\langle M\rangle \mid M$ מ"ט ו-$L(M)$ מקיימת תנאי$\}$, עבור אילו תנאים L כריעה?

משפט: כמעט אף פעם.

בעיה: לא בדיוק.

דוגמה. מספר דוגמאות לשפות כריעות מסוג זה.

$$L = \{\langle M\rangle \mid M \text{ מ"ט ו-} L(M) \text{ היא שפה } = \Sigma^*\}$$

L כריעה - מ"ט שמקבלת כל קלט.

$$L = \{\langle M\rangle \mid M \text{ מ"ט ו-} L(M) \text{ היא שפה } = \emptyset\}$$

L כריעה - מ"ט שדוחה כל קלט.

$$L = \{\langle M\rangle \mid M \text{ מ"ט עם לפחות } 100 \text{ מצבים}\}$$

L כריעה - ניתן לקבוע באמצעות הקידוד של M בלבד, ללא תלות ב-$L(M)$.

הגדרה. שפה L היא טריוויאלית אם $L = \emptyset$ או $L = \Sigma^*$.

משפט. כל שפה $L \subseteq \Sigma^*$ שאינה טריוויאלית ומקיימת תכונה של שפות היא לא כריעה.

הגדרה. שפה L מקיימת תוכה של שפות אם לכל מ"ט M_1, M_2 כך ש-$L(M_1) = L(M_2)$ מתקיים:

$$\langle M_1\rangle \in L \iff \langle M_2\rangle \in L$$

כלומר, L לא מבדילה בין שתי מ"ט בעלות אותה השפה.

דוגמה. השפה $EVEN = \{\langle M\rangle \mid \exists w \in \Sigma^* : |w| \equiv 0 \mod 2 \land w \in L(M)\}$ לא כריעה.

1. $EVEN \subseteq \Sigma^*$.
2. $EVEN$ אינה טריוויאלית (מ"ט שדוחה כל קלט לא בשפה ומ"ט שמקבלת כל קלט בשפה).
3. $EVEN$ מקיימת תכונה של שפות: היא אינה מתייחסת למבנה המ"ט, ורק לשפה. כך, היא לא תבדיל בין שתי מ"ט שוות שפה.

הוכחה. הוכחת משפט רייס
תהי L שפה שמקיימת את שלושת תנאי המשפט.
נגדיר מ"ט M_{EMPTY} כך ש-$L\left(M_{EMPTY}\right)=\emptyset$. נפריד למקרים:

1. $\langle M_{EMPTY}\rangle\notin L$.
2. $\langle M_{EMPTY}\rangle\in L$.

אם $\langle M_{EMPTY}\rangle\notin L$, מאחר ו-$L$ אינה טריוויאלית אז $L\neq\emptyset$. לכן, קיימת מ"ט T כך ש-$\langle T\rangle\in L$.
נוכיח כי $ACCEPT\leq_m L$:
נגדיר רדוקציית מיפוי R כך ש-$R\left(\langle M,w\rangle\right)=\langle M'\rangle$, כ-$M'\left(x\right)$ מוגדרת ע"י:

- הרץ את M על w.
  - אם M דוחה את w, דחה.
- הרץ את T על x והחזר כמוה.

טענה. מספר טענות.

1. אם $w\in L\left(M\right)$ אז $L\left(M'\right)=L\left(T\right)$.
   (א) מאחר ו-L מקיימת תכונה של שפות אז היא לא מבדילה בין M' ו-T.
   (ב) מאחר ו-$\langle T\rangle\in L$ אז גם $\langle M'\rangle\in L$.
2. אם $w\notin L\left(M\right)$ אז $L\left(M'\right)=\emptyset=L\left(M_{EMPTY}\right)$.
   (א) מאחר ו-L מקיימת תכונה של שפות אז היא לא מבדילה בין M' ו-M_{EMPTY}.
   (ב) מאחר ו-$\langle M_{EMPTY}\rangle\notin L$ אז גם $\langle M'\rangle\notin L$.

לבסוף, בנינו רדוקציית מיפוי מ-$ACCEPT$ ל-L וכך L אינה כריעה. כעת, נטפל במקרה בו $\langle M_{EMPTY}\rangle\in L$:

- נסתכל על השפה $\overline{L}$.
  - $\overline{L}\subseteq\Sigma^*$.
  - $\overline{L}$ אינה טריוויאלית (כי L אינה טריוויאלית).
  - $\overline{L}$ מקיימת תכונה של שפות (סגורה תחת משלים).
  - $M_{EMPTY}\notin\overline{L}$.
- מכאן, $\overline{L}$ מקיימת את כל תנאי המקרה הקודם, וכך $\overline{L}$ אינה כריעה.
- מאחר ו-L כריעה אמ"מ $\overline{L}$ כריעה, קיבלנו כי L אינה כריעה.

$\square$

הערה. משפט רייס לא תמיד יחסוך שימוש ברדוקציית מיפוי.
עבור שפות שלא מקיימות את התכונה, למשל:

$$DECIDE=\{\langle M\rangle\mid M\text{ מ"ט שעוצרת על כל קלט}\}$$

ייתכנו שתי מ"ט שמקבלות את אותה השפה - אחת תעצור ותדחה קלט קלט ואחת תיכנס ללולאה אינסופית.

דוגמה. מספר שפות שאינן כריעות (לפי רייס) וניתנות לקבלה (ע"י הרצה מבוקרת)

$$EVEN = \{\langle M\rangle \mid \text{מ"ט שמקבלת מילה כלשהי מאורך זוגי } M\}$$

הראנו כי $EVEN$ לא כריעה, נראה אלגוריתם שמקבל אותה:

1. אתחל $i \leftarrow 0$.
2. עבור על כל המילים w כך ש-$|w| \leq i$:
   (א) אם w מאורך זוגי הרץ את M על w ל-i צעדים.
   (ב) אם M קיבלה אז קבל.
3. בצע $i \leftarrow i+1$ וחזור ל-1.

$$ZEROS = \{\langle M\rangle \mid \text{מ"ט שמקבלת מילה כלשהי שמתחילה ב-000 } M\}$$

ניתן להראות בקלות ש-$ZEROS$ לא כריעה ע"פ רייס. אלגוריתם שמקבל את השפה:

1. אתחל $i \leftarrow 0$.
2. עבור על כל המילים w כך ש-$|w| \leq i$:
   (א) אם w מתחיל ב-000 הרץ את M על w ל-i צעדים.
   (ב) אם M קיבלה אז קבל.
3. בצע $i \leftarrow i+1$ וחזור ל-1.

$$HAS-REGULAR = \{\langle M\rangle \mid \text{מ"ט שמקבלת ביטוי רגולרי כלשהו } M\}$$

ניתן להראות בקלות שהשפה לא כריעה ע"פ רייס. אלגוריתם שמקבל אותה:

1. אתחל $i \leftarrow 0$.
2. עבור על כל המילים w כך ש-$|w| \leq i$:
   (א) אם w הוא ביטוי רגולרי הרץ את M על w ל-i צעדים.
   (ב) אם M קיבלה אז קבל.
3. בצע $i \leftarrow i+1$ וחזור ל-1.

משפט. אם $L, \overline{L}$ ניתנות לקבלה אז $L, \overline{L}$ כריעות.

הוכחה. קיימות מ"ט M_1, M_2 שמקבלות את $L, \overline{L}$ בהתאמה. נתאר אלגוריתם שמכריע את L.
בהינתן קלט w, מתקיים כי $w \in L(M_1)$ או $w \in L(M_2)$, ואחת מהן לפחות תעצור. האלגוריתם:

1. אתחל $i \leftarrow 1$.
2. הרץ את M_1, M_2 על w למשך i צעדים.
3. אם אחת עצרה וקיבלה, החזר בהתאם (M_1 - קבל, M_2 - דחה).
4. בצע $i \leftarrow i+1$ וחזור ל-1.

□

מסקנה. אם L לא כריעה ו-$\overline{L}$ ניתנת לקבלה, אזי L לא ניתנת לקבלה.

הוכחה. אחרת, $L, \overline{L}$ ניתנות ולקבלה וכך שתיהן כריעות - סתירה. □

דוגמה. $NOT-ACCEPT$ אינה ניתנת לקבלה.
מאחר ו-$ACCEPT$ לא כריעה, גם $NOT-ACCEPT$ לא כריעה.
מכאן, מאחר ו-$ACCEPT$ ניתנת לקבלה, $NOT-ACCEPT$ אינה ניתנת לקבלה

מסקנה. שפות ניתנות לקבלה לא סגורות תחת משלים.

משפט. אם $L_1 \leq_m L_2$ ו-L_1 לא ניתנת לקבלה אז L_2 לא ניתנת לקבלה.

דוגמה. $EMPTY$ אינה ניתנת לקבלה.
הוכחנו כי $\overline{EMPTY} = NOT-EMPTY$ אינה כריעה וניתנת לקבלה.
מכאן, $EMPTY$ אינה ניתנת לקבלה.
בנוסף, ניתן להשתמש במשפט לעיל:

- $NOT-ACCEPT \leq_m EMPTY$, ניתן להראות ע"י רדוקציית מיפוי:
  - $R(\langle M, w\rangle) = M'$, כאשר M' מריצה את M על w ומחזירה כמוה.
- מכאן, מאחר ו-$NOT-ACCEPT$ לא ניתנת לקבלה אז $EMPTY$ לא ניתנת לקבלה.

הערה. באופן כללי, כאשר נרצה להוכיח אי ניתנות לקבלה, נשתמש באחת מהשיטות:

1. רדוקציה משפה שאינה ניתנת לקבלה.
2. הוכחת אי כריעות של השפה ונתינות לקבלה של השפה המשלימה.

דוגמה. $EQUAL$ לא ניתנת לקבלה.

- השיטה השנייה לא תעבוד, מאחר ו-$NOT-EQUAL$ גם לא ניתנת לקבלה.
- ראינו קודם לכן כי $EMPTY \leq_m EQUAL$.
- מכאן, מאחר ו-$EMPTY$ לא ניתנת לקבלה אז $EQUAL$ לא ניתנת לקבלה.

4.2 שיטת מסלולי החישוב

טענה. השפה $ALL = \{\langle M\rangle \mid L(M) = \Sigma^*$ מ"ט ו- $M\}$ אינה כריעה.

הוכחה. באמצעות משפט רייס.

- ALL אינה טריוויאלית - יש מ"ט ששפתן Σ^* ויש כאלו שלא.
- ALL מהצורה המאתימה ומקיימת תכונה של שפות.

לכן, השפה מקיימת את תנאי המשפט וכך ALL אינה כריעה. $\square$

משפט. ALL לא ניתנת לקבלה.

הוכחה. נראה באמצעות רדוקציה מ-$NOT-ACCEPT$ ל-ALL.
נגדיר רדוקציית מיפוי R להיות $R(\langle M,w\rangle) = \langle M'\rangle$, כאשר:

$$w \notin L(M) \Longleftrightarrow L(M') = \Sigma^*$$

השפה של M' תהיה כל מסלול שאינו מקבל. נסתכל על מסלולי החישוב של M כמילים:
נקודד קונפיגורציה (שלשה שמתארת את מצב הסרט) $C = (w_1, q, w_2)$ ע"י המילה w_1qw_2, מעל א"ב שכולל את $\Gamma \cup Q$.
מסלול חישוב הוא רצף קונפיגורציות $\{C_i\}_{i\geq 1}$, כך שלכל i מתקיים $C_i \vdash_M C_{i+1}$.
בנוסף, נגדיר שפה שתכיל את כל קידודי מסלולי החישוב הסופיים.
נקודד מסלול חישוב $\{C_i\}_{i=1}^t$ באמצעות המילה $C_1\#C_2\#\cdots\#C_t$. בהינתן מ"ט M ומילה w, נגדיר שפות $A_{M,w}, B_{M,w}$ באופן הבא:

$$A_{M,w} = \{\text{מסלולי חישוב של } M \text{ על } w \text{ שעוצרים ומקבלים}\}$$

$$B_{M,w} = \overline{A_{M,w}}$$

- אם $w \in L(M)$, אז $A_{M,w}$ מכילה מילה בודדת (M דטרמיניסטית) והשפה המשלימה $B_{M,w} \neq \Sigma^*$.
- אחרת, $A_{M,w} = \emptyset$ ומתקיים $B_{M,w} = \Sigma^*$.

מכאן, $B_{M,w}$ היא בדיוק השפה שנרצה להגיע אליה ברדוקציית המיפוי.

- השפה $A_{M,w}$ כריעה לכל M, w.
  - מכאן, גם $B_{M,w} = \overline{A_{M,w}}$ כריעה.
- בנוסף, בהינתן $\langle M,w\rangle$ קל לבנות קידוד של מ"ט M' שמכריעה את $B_{M,w}$.

לכן, רדוקציית המיפוי תהיה $R(\langle M,w\rangle) = M'$ כך ש-$L(M') = B_{M,w}$.

- אם $w \in L(M)$ אז $L(M') = B_{M,w} \neq \Sigma^*$, כלומר $\langle M'\rangle \notin ALL$.
- אחרת, $L(M') = \Sigma^*$ וכך $\langle M'\rangle \in ALL$.

$\square$

הערה. בשונה מהוכחות קודמות, בהוכחה זו אף מ"ט (או קידוד של אחת) לא מריצה את M על w.
בפרט, M' עוצרת על כל קלט.

הערה. נסמן ב-DFA אס"ד, ב-$NDFA$ אסל"ד, ב-CFG דח"ה וב-TM מ"ט.

שפה	כריעות	שיטה
$ACCEPT_{DFA} = \{\langle M, w\rangle \mid w \in L(M)$-ו אס"ד $M\}$	✓	מסמלצים
$EMPTY_{DFA} = \{\langle M\rangle \mid L(M) = \emptyset$-ו אס"ד $M\}$	✓	מחפשים מצב מקבל עם DFS
$ALL_{DFA} = \{\langle M\rangle \mid L(M) = \Sigma^*$-ו אס"ד $M\}$	✓	הופכים מצבים ובודקים האם ב-$EMPTY_{DFA}$
$ACCEPT_{NDFA} = \{\langle M, w\rangle \mid w \in L(M)$-ו אסל"ד $M\}$	✓	הופכים לאס"ד
$EMPTY_{NDFA} = \{\langle M\rangle \mid L(M) = \emptyset$-ו אסל"ד $M\}$	✓	הופכים לאס"ד
$ALL_{NDFA} = \{\langle M\rangle \mid L(M) = \Sigma^*$-ו אסל"ד $M\}$	✓	הופכים לאס"ד
$ACCEPT_{CFG} = \{\langle M, w\rangle \mid w \in L(M)$-ו דח"ה $M\}$	✓	בדיקה האם מילה ניתנת לגזירה, מופיע בנספחים
$EMPTY_{CFG} = \{\langle M\rangle \mid L(M) = \emptyset$-ו דח"ה $M\}$	✓	מופיע בנספחים
$ALL_{CFG} = \{\langle M\rangle \mid L(M) = \Sigma^*$-ו דח"ה $M\}$	X	מפתיע!

טבלה 11: מספר שפות על מודלים שאינם מ"ט וכריעותן

משפט. השפה ALL_{CFG} אינה ניתנת לקבלה.

הוכחה. נוכיח ברדוקציה מ-$NOT-ACCEPT$ ל-ALL_{CFG}.
נבנה רדוקציה $R(\langle M, w\rangle) = G$, כאשר:

$$w \notin L \iff L(G) = \Sigma^*$$

נוכיח באופן דומה להוכחה הקודמת, רק נרצה לבנות א"מ M' ששפתו $B_{M,w}$.
נקודד קונפיגורציה (שלשה שמתארת את מצב הסרט) $C = (w_1, q, w_2)$ ע"י המילה w_1qw_2, מעל א"ב שכולל את $\Gamma \cup Q$.
נקודד מסלול חישוב $\{C_i\}_{i=1}^{t}$ באמצעות המילה $C_1\#C_2^R\#C_3\#C_4^R\cdots\#C_t$.
נותר לבנות א"מ שמקבל את $B_{M,w}$. כל מילה ב-$B_{M,w}$ מקיימת את אחד מהבאים:

1. המילה מכילה קידוד לא תקין - L_1.
2. הקונפיגורציה C_1 לא מהצורה (ε, q_0, w) - L_2.
3. הקונפיגורציה C_t אינה מקבלת (המצב אינו q_{ACCEPT}) - L_3.
4. קיים i כך ש-$C_i \nvdash_M C_{i+1}$ - L_4.

$$\Longrightarrow B_{M,w} = L_1 \cup L_2 \cup L_3 \cup L_4$$

קל להוכיח ש-L_1, L_2 ו-L_3 ח"ה. נוכיח כי L_4 ח"ה:

- באופן אי-דטרמיניסטי, ננחש כי במעבר ה-i מתקיים $C_i \nvdash_M C_{i+1}$.
- נדחוף את C_i למחסנית (ויופיע בסדר הפוך) $C_i^R = w_2^R q w_1^R$.
  - אם היינו צריכים לבדוק $C_i \neq C_{i+1}$, היינו עושים זאת בקלות עם הוצאה מהמחסנית.

- באופן דומה, נבדוק האם $C_i \nvdash_M C_{i+1}$.
  - מאחר והקונפיגורציות עוקבות, הן שונות ב-3 תווים לכל היותר: סביב המצב q.
  - נבדוק שוויון בכל אזור אחר, וסביב האזור השונה נבדוק אם המעבר תקין.

מכאן, גם L_4 ח"ה וכך $B_{M,w}$ ח"ה (מסגירות תחת איחוד של שפות ח"ה). כלומר, קיים דח"ה G שגוזר את $B_{M,w}$, הרדוקציה עובדת ו-$NOT-ACCEPT \leq_m ALL_{CFG}$.
בסך הכל, קיבלנו כי ALL_{CFG} לא ניתנת לקבלה. $\square$

הערה. השפה $A_{M,w}$ אינה ח"ה, וניסיון להוכיח שהיא כן היה נכשל ב-L_4:
התנאי היה דורש לכל במקום קיים, וכך א"מ בלבד לא היה מסוגל לקבל את השפה.

מסקנה. לא כל בעיה של אס"ד או א"מ היא כריעה!

משפט. שפה אינסופית $L = \{\langle M\rangle \mid$ מקיימת תכונה מסויימת $L(M)\}$ שקיים אלגוריתם שבודק את קיום התכונה אינה קבילה.

דוגמה. השפה $L = \{\langle M\rangle \mid$ מקבלת כל ראשוני $M\}$
נראה כי $NOT-ACCEPT \leq_m L$ ע"י רדוקציה R:

$$R(\langle M,w\rangle) = \langle M'\rangle$$

- כאשר $M'(x)$:
  - הרץ את M על w למשך $|x|$ צעדים, ודחה אם היא קיבלה.
  - קבל אם x ראשוני ודחה אחרת.
- אם $\langle M,w\rangle \in NOT-ACCEPT$ אז $L(M')$ היא כל הראשוניים.
- אחרת, נסמן ב-k את מספר הצעדים לקבלת w:
  - קיים ראשוני שגדול מ-k, ועבורו נדחה (M תקבל את w)
  - מכאן, $\langle M'\rangle \neq L$

5 P ו-NP

בתחילת הקורס שאלנו את עצמנו את השאלות:

- מה מחשב יכול לבצע?
  - כריעות / ניתנות לקבלה של שפות
- מה מחשב יכול לבצע באופן יעיל?
  - למשל, שפה כריעה שלוקח זמן רב להכריע אותה

משאבי חישוב: זמן החישוב.

- המשאב המעניין ביותר

- בקורסים אחרים - זיכרון, זמן מקבילי, תקשורת

דוגמה. מה יעיל ומה לא יעיל? מספר פונקציות זמן ריצה $T(n)$:

$$30 \quad 10n^2 \quad 100000n^{100000} \quad 2^n \quad n!$$

<u>מטרה:</u>

- יעילות שלא תלויה במודל החישוב (למשל, שימוש במ"ט עם כמה סרטים)
- אם תכנית 1 יעילה, ותכנית 2 קוראת ל-1 ויעילה בשאר חלקיה, אז גם 2 יעילה

הגדרה. זמן ריצה פולינומי מוגדר ע"י פונקציית זמן $T(n) = \mathcal{O}(n^c)$, עבור $c \in \mathbb{R}$.

הערה. לעיתים c גדול מאוד ייראה (באופן מעשי, עבור קלטים שאינם גדולים) הרבה יותר גרוע מפונקציית זמן 2^n, אך עבור n גדול מספיק הדבר לא נכון.

הגדרה. עבור מ"ט דטרמיניסטית M וקלט x כך ש-M עוצרת על x, נגדיר את $Time(M,x)$ להיות מספר הצעדים של M על x.

עבור פונקציה $t:\mathbb{N}\to\mathbb{N}$, נאמר כי M רצה בזמן t אם:

$$\forall x \in \Sigma^* : Time(M,x) \leq t(|x|)$$

הגדרה. נגדיר את הקבוצות הבאות.

$$Time(t(n)) = \{L \mid L(M) = L \text{ וגם } \mathcal{O}(t(n)) \text{ קיימת מ"ט } M \text{ דטרמיניסטית שרצה בזמן}\}$$

$$P = \bigcup_{c=1}^{\infty} Time(n^c) = \{\{L \mid L(M) = L \text{ קיימת מ"ט } M \text{ דטרמיניסטית שרצה בזמן פולינומי וגם}\}\}$$

הערה. P היא קבוצת השפות שמחשבים יכולים להכריע באופן יעיל (זמן פולינומיאלי).

הגדרה. עבור מ"ט <u>לא דטרמיניסטית</u> M וקלט x, נגדיר את $NTime(M,x)$ להיות אורך מסלול החישוב הארוך ביותר של M על x.

בהינתן פונקציה $t:\mathbb{N}\to\mathbb{N}$, נאמר כי מ"ט M רצה בזמן t אם:

$$\forall x \in \Sigma^* : NTime(M,x) \leq t(|x|)$$

הגדרה. נגדיר את הקבוצות הבאות.

$$NTime(t(n)) = \{L \mid L(M) = L \text{ וגם } \mathcal{O}(t(n)) \text{ קיימת מ"ט } M \text{ לא דטרמיניסטית שרצה בזמן}\}$$

$$NP = \bigcup_{c=1}^{\infty} NTime(n^c) = \{\{L \mid L(M) = L \text{ קיימת מ"ט } M \text{ לא דטרמיניסטית שרצה בזמן פולינומי וגם}\}\}$$

הערה. NP היא קבוצת השפות שמחשבים אי דטרמיניסטיים יכולים להכריע באופן יעיל (זמן פולינומיאלי).

משפט. הגדרת P אינה תלויה במספר הסרטים של המ"ט.

הוכחה. ראינו כי מ"ט k-סרטית שקולה למ"ט חד-סרטית, וגם כי אם ה-k-סרטית רצה בזמן $t\left(n\right)$ אז החד-סרטית רצה בזמן $\mathcal{O}\left(t^{2}\left(n\right)\right)$.
מכאן, במעבר מ-k-לסרטית לחד-סרטית נישאר ב-P. □

מסקנה. הגדרת P על כל מחשב שאנו מכירים שקולה להגדרת P ע"י מ"ט.
כלומר, השפות שניתנות להכרעה בזמן פולינומי זהות בכל מודל חישוב שאנו מכירים כיום.
לכן, נוכל לפתח את התיאוריה עבור מ"ט והדבר יתאים לכל מחשב (פרט למחשב קוונטי).

דוגמה. דוגמא לשפה ב-P - ניתן להמיר כל בעיה לשפה:

$$PATH=\{\langle G,s,t\rangle \mid G=(V,E) \text{ גרף, } s,t\in V \text{ וקיים מסלול } s\rightsquigarrow t\}$$

ניתן לקודד את G ע"י מטריצת שכנויות, כלומר מחרוזת באורך n^{2}.

משפט. $PATH\in P$.

הוכחה. נבצע BFS או DFS בזמן לינארי (באורך הקלט), ולכן $\mathcal{O}\left(n^{2}\right)$ ופולינומי. □

הערה. $P\subseteq NP$, וכך גם $PATH\in NP$.

משפט. לכל שפה רגולרית L מתקיים $L\in P$.

הוכחה. נוכל לסמלץ אסל"ד באמצעות מ"ט, וכך נפתור בזמן לינארי באורך המילה. □

משפט. לכל שפה ח"ה L מתקיים $L\in P$.

הערה. אין לנו עדיין את הכלים להוכיח את המשפט.
האלגוריתם שבאמצעותו הוכחנו כי כל שפה ח"ה היא כריעה עובד בזמן אקספוננציאלי.

5.1 $HAM-PATH$

דוגמה. דוגמא לשפה ב-NP:

$$HAM-PATH=\{\langle G,s,t\rangle \mid G \text{ גרף מכוון ויש מסלול המילטוני מ-} s \text{ ל-} t\}$$

תזכורת - מסלול המילטוני הוא מסלול שמבקר בכל קודקוד בדיוק פעם אחת.

הערה. האם $HAM-PATH\in P$? לא יודעים.

טענה. $HAM-PATH\in NP$.

הוכחה. נבנה מ"ט לא דטרמיניסטית M שרצה בזמן פולינומי ומכריעה את $HAM-PATH$.
למ"ט לא דטרמיניסטית יש כוח רב לעומת הדטרמיניסיטית - נוכל לנחש פיתרון באופן דטרמיניסטי, ומשם נותר רק לוודא אותו.

- ננחש באופן לא דטרמיניסטי מסלול המילטוני מ-s ל-t.

- נבדוק שהוא באמת מסלול המילטוני, ונקבל בהתאם.

באופן פורמלי:

- נסמן את מספר הקודקודים בגרף ב-n, ונקודד קודקודים ע"י מספר $1, \cdots, n$, מיוצג בבסיס בינארי.
  - אורך של קודקוד הוא $\log n$
  - נקודד סדרת קודקודים ע"י שרשור שלהם, וכך נקודד סדרה של k קודקודים ב-$k \log n$ ביטים.
- נבנה מ"ט לא דטרמיניסטית תלת-סרטית, שמאותחלת עם הקלט בסרט הראשון.
  - הסרט השלישי מכיל 1 ואחריו $n \log n - 1$ 0-ים

נחלק את האלגוריתם לשני שלבים - שלב הניחוש ושלב הבדיקה.

- שלב הניחוש:
  - נכתוב בסרט השני $n \log n$ 1-ים
  - יהיה מצב שעבור 1 בסרט השני יאפשר:
    * כתיבת 0 בסרט השלישי ותזוזה ימינה עם שני הראשים.
    * כתיבת 1 בסרט השלישי ותזוזה ימינה עם שני הראשים.
- שלט הבדיקה: (דטרמיניסטי) התייחס לסרט השלישי כפרמוטציה של $[n]$, ובדוק האם הוא קידוד של מסלול המילטוני מ-s ל-t:
  - בדוק שהקודקוד הראשון הוא s
  - בדוק שהקודקוד האחרון הוא t
  - בדוק שכל שני קודקודים עוקבים מחוברים בקשת
  - בדוק שכל קודקוד מופיע פעם אחת בלבד
    קבל אם כל הבדיקות צלחו, ואחרת דחה.

מ"ט זו פשוט בודקת את כל האפשרויות - exhaustive search.
נוכיח כי המכונה מקבלת את $HAM - PATH$:

- אם $\langle G, s, t\rangle \in HAM - PATH$, אז קיים ב-$G$ מסלול המילטוני מ-s ל-t.
  - נסמן את קידוד המסלול ב-y ($|y| = n \log n$)
  - קיים מסלול חישוב בו נכתב y על הסרט השלישי במסלול, הבדיקה תעצר והקלט יתקבל
  - מכאן, $\langle G, s, t\rangle \in L\left(M\right)$
- אם המכונה קיבלה קלט $\langle G, s, t\rangle$, אזי קיים מסלול חישוב בו המכונה עצרה וקיבלה.
  - כלומר, נמצא כי קיים מסלול המילטוני בגרף.
  - לכן $\langle G, s, t\rangle \in HAM - PATH$

נוכיח כי המכונה רצה בזמן פולינומי:

- זמן הריצה של המכונה הוא אורך מסלול החישוב הארוך ביותר

- כל מסלול חישוב באורך סופי
- שלב הניחוש עולה $\mathcal{O}\left(n\log n\right)$
- שלב הבדיקה עולה $\mathcal{O}\left(n\times n^{2}\right)$ (באופן שאינו יעיל ביותר)

- בסך הכל, זמן הריצה הוא $\mathcal{O}\left(n^{3}\right)=\mathcal{O}\left(\left(n^{2}\right)^{1.5}\right)$
  - אורך הקלט בחזקת 1.5

בסך הכל, קיבלנו כי $HAM-PATH\in NP$. □

5.2 $CLIQUE$

דוגמה. $CLIQUE=\{\langle G,k\rangle\mid k$ קליקה בגודל G-וקיימת ב $k\in\mathbb{N}$,גרף $G\}$
קליקה - קבוצת קודקודים בה יש קשת בין כל זוג קודקודים.

טענה. $CLIQUE\in NP$

הוכחה. נבנה מ"ט לא דטרמיניסטית M שמכריעה את $CLIQUE$ בזמן פולינומי.
הרעיון: לבדוק כל קבוצה אפשרית בגודל k.
בהינתן גרף על n קודקודים וקבוצת קודקודים S בגודל k, נקודד אותה ע"י מחרוזת בינארית באורך $k\log n$ (באופן דומה ל-$HAM-PATH$)

- שלב הניחוש:
  - כתיבה של $k\log n$ 1-ים על הסרט השני.
  - במצב q המ"ט יכולה (עם לולאה עצמית ל-q):
    - * אם יש 1 בסרט השני, לכתוב $0/1$ בסרט השלישי ולזוז ימינה בשניהם.
    - * אם הגענו לסוף רצף ה-1-ים בסרט השני אז נגמר שלב הניחוש.
      נסמן ב-y את תוכן הסרט השלישי במצב זה.
- שלב הבדיקה (דטרמיניסטי): נבדוק שהקבוצה המקודדת בסרט השלישי היא קליקה.
- נרצה לחשב את $f\left(x,y\right)$:
  - בדוק ש-x מהצורה $\langle G,k\rangle$.
  - בדוק ש-y הוא קידוד של קליקה מגודל k ב-G (עוברים כל כל זוג ובודקים במטריצה האם הם שכנים).
- אם שתי הבדיקות מצליחות קבל, ואחרת דחה.

נוכיח כי $L\left(M\right)=CLIQUE$:

- אם קיימת קליקה בגרף, יש מסלול חישוב שינחש אותה וכך הקלט יתקבל.
- אם יש מסלול חישוב מקבל, אזי הוא ניחש ובדק שקיימת קליקה - וכך יש קליקה בגודל k ב-G.

בסך הכל, $w\in CLIQUE\iff w\in L\left(M\right)$ וכך $L\left(M\right)=CLIQUE$.
נוכיח כי M רצה בזמן פולינומי:

- שלב הניחוש עולה $n^2+k\log n=\mathcal{O}\left(n^2\right)$
- שלב הבדיקה עולה $k^2n^2=\mathcal{O}\left(n^4\right)$ (ניתן באופן יעיל יותר)
- כך, בסך הכל עלות האלגוריתם היא $\mathcal{O}\left(n^4\right)$
  - גודל הקלט הוא n^2, וכך מחשבים בזמן ריבועי כגודל הקלט

בסך הכל, קיבלנו כי $CLIQUE\in NP$.

□

הערה. דרך נוספת להסתכל על NP

- P - שפות שניתנות להכרעה בזמן פולינומי ע"י מ"ט דטרמיניסטית
- NP - שפות שניתנות לבדיקה בזמן פולינומי ע"י מ"ט דטרמיניסטית בהינתן "תמיכה"
  - למשל, קיבלנו קליקה ועלינו רק לבדוק שהיא מתאימה לדרישות

הגדרה. נאמר שלשפה L יש מוודא פולינומי אם קיים $c\in\mathbb{N}$ ומ"ט דטרמיניסטית M שרצה בזמן פולינומי כך ש:

$$\forall x\in\Sigma^*\exists y:|y|\leq|x|^c\wedge M\left(x,y\right)=1\iff x\in L$$

הסימון $M\left(x,y\right)=1$ משמעו M מקבלת את הקלט x,y

הערה. בהוכחה של $CLIQUE\in NP$ הראנו כי ל-$CLIQUE$ יש מוודא פולינומי עבור $c=1$. העזרה היא הפתרון, ועל המ"ט הדטרמיניסטית רק לוודא שהוא נכון.

משפט. $L\in NP$ אמ"מ ל-L קיים מוודא פולינומי

הוכחה. נוכיח את המשפט

- אם $L\in NP$, קיימת מ"ט א"ד פולינומית M שמקבלת את L.
  - מכאן, קיים c כך שלכל קלט x מסלול החישוב הארוך ביותר של M, הוא לכל היותר מאורך $|x|^c$.
  - נתייחס למסלול חישוב מקבל בתור "עד" y, שמקבל כקלט $x\in L$.
  - נגדיר מוודא פולינומי M' עבור השפה L.
    * בהינתן קלט x ומחרוזת y מאורך $|x|^c$ לכל היותר, הוא יתייחס ל-y כמסלול חישוב של M.
    * נריץ את את M על x במסלול החישוב y (ומכאן ההרצה דטרמיניסטית), והחזר כמוה.
    * זמן הריצה פולינומי, מאחר ו-y פולינומי באורכו.
  - אם $x\in L$, קיים מסלול חישוב מקבל y וכך קיים y שעבורו $M'\left(x,y\right)=1$.
  - אם $x\notin L$, לא קיים מסלול חישוב מקבל וכך לכל y מתקיים $M'\left(x,y\right)=0$.
  - מכאן, M' הוא מוודא פולינומי של L.

- אם ל-L יש מוודא פולינומי, קיים קבוע c ומ"ט דטרמיניסטית M שרצה בזמן פולינומי:

$$\forall x \in \Sigma^* : \exists y \left|y\right| \leq \left|x\right|^c \land M\left(x,y\right) = 1 \Longleftrightarrow x \in L$$

  - נבנה מ"ט א"ד פולינומית שמקבלת את L:
    * נחש מחרוזת y מאורך לכל היותר $\left|x\right|^c$.
    * הרץ $M\left(x,y\right)$ וקבל אמ"מ M קיבלה.
  - אם $x \in L$, קיים y כך ש-$M\left(x,y\right) = 1$, וכך מסלול חישוב שבו נוחש y יתקבל.
  - אם $x \notin L$, לכל מסלול חישוב מתקיים $M\left(x,y\right) = 0$ וכך המילה תידחה.

בסך הכל, הוכחנו כי לשפה L קיים מוודא פולינומי אמ"מ $L \in NP$. □

הערה. בשלב מאוחר יותר נראה כי יש שפות כריעות שלא ב-NP.

- בחזרה ל-$P = NP$

נניח ש-$P = NP$, אז אפשר לפתור כל דבר באופן אופטימלי:

- צריך לבנות גשר
  - מציאת תכנון y הוא מסובך
  - בדיקה של תכנון כלשהו קלה הרבה יותר
- באופן מעשי, נשמע מופרך!

הערה. יש בעיות (מסלול המילטוני, למשל) שהן קשות לפחות כמו כל שאר הבעיות ב-NP.
מכאן, אם נצליח להוכיח שהן ב-P אז $P = NP$.

5.3 $3-SAT$

תזכורת: אלגברה בוליאנית.

- משתנים: $x_1, x_2, ...$
- קשרים לוגיים: $\lor, \land, \neg$ (סדר פעולות: $\neg$, $\land$, ולבסוף $\lor$)
- ליטרל: x או $\neg x$
- פסוקית: $l_1 \lor l_2 \lor \cdots \lor l_t$ כאשר $l_1, ..., l_t$ ליטרלים
- נוסחא: $(x_1 \lor \neg x_2) \land (x_3 \lor \neg x_1 \lor x_2) \land x_1 \land \neg x_4$

נאמר כי נוסחא היא בצורת CNF אם היא גימום (מהמילה גם - $\land$) של פסוקית. בנוסף, נוסחא היא בצורת $k-CNF$ אם בכל פסוקית יש בדיוק k ליטרלים.

- השמה היא פונקציה α שמתאימה לכל משתנה x_i ערך אמת.
- בהינתן נוסחא φ והשמה α, נוכל להציב את α ב-φ לקבלת ערך אמת

הגדרה. נאמר כי נוסחא φ היא ספיקה אם קיימת השמה α שהצבתה ב-φ גורמת לערך T.

$$3-SAT=\{\varphi \mid 3-CNF \text{ נוסחא ספיקה בצורת } \varphi\}$$

טענה. $3-SAT\in NP$.

הוכחה. ננחש השמה מספקת, ונותר רק לבנות מוודא פולינומי.
נבנה מוודא פולינומי $M\left(x,y\right)$ ל-$3-SAT$:

- בדוק ש-x נוסחא בצורת $3-CNF$
- נסתכל ב-y (מתקיםי כי $\left|y\right|\le\left|x\right|^{1}=\left|x\right|$, כלומר $c=1$)
- נסמן $m=\left|x\right|$ ואת מספר המשתנים ב-n, ברור כי $n\le m$
- אם $\left|y\right|=n$ אז y השמה תקינה, נציב את y ב-x ונפלוט את ערך האמת.

□

<u>שאלה:</u> האם $3-SAT\in P$?

<u>תשובה:</u> לא יודעים! (כנ"ל עבור $CLIQUE$)

עם זאת, אנחנו יודעים את הדברים הבאים:

- $3-SAT\in P\Longleftrightarrow CLIQUE\in P$
- אם $P\neq NP$ אז $3-SAT\notin P$
  - כלומר, אם $3-SAT\in P$ אז $P=NP$

5.4 רדוקציה פולינומית

הגדרה. בהינתן שתי שפות L_1,L_2, נאמר שיש רדוקציה פולינומית מ-L_1 ל-L_2, ונסמן $L_1\le_p L_2$, אם קיימת מ"ט R שעל כל קלט x עוצרת ופולטת $R\left(x\right)$, כך ש:

$$x\in L_1\Longleftrightarrow R\left(x\right)\in L_2$$

בנוסף, R רצה בזמן פולינומי.

משפט. אם $L_1\le_p L_2$ אז $L_2\in P\Longleftarrow L_1\in P$

הוכחה. נגדיר מ"ט עובר L_1 להיות $M_1\left(x\right)=M_2\left(R\left(x\right)\right)$.
זמן הריצה של M_2 הוא $\mathcal{O}\left(n^{c_1}\right)$ עבור $c_1\in\mathbb{N}$.
זמן הריצה של R הוא $\mathcal{O}\left(n^{c_2}\right)$ עבור $c_2\in\mathbb{N}$.
אזי, זמן הריצה של M_1 הוא $\mathcal{O}\left(n^{\max\{c_1,c_2\}}\right)$.
□

דוגמה. $3-SAT\le_p CLIQUE$
מכאן, אם $CLIQUE\in P$ אז $3-SAT\in P$
למרות שהבעיות נראות שונות לחלוטין!

הוכחה. נבנה רדוקציה פולינומית R שמקבלת נוסחא φ בצורת $3-CNF$, ופולטת גרף G ומספר k.

נסמן את מספר המשתנים ב-φ ב-n, ואת מספר הפסוקיות ב-φ ב-m.

$$\varphi = c_1 \wedge \cdots \wedge c_m$$

$$c_i = l_1 \vee l_2 \vee l_3$$

נבנה גרף G בעל $3m$ קודקודים - קודקוד לכל מופע של ליטרל ב-φ.
הגרף יהיה מלא, מלבד:

- קשתות בין שני קודקודים מאותה פסוקית (=שורה)
- קשתות בין ליטרלי לשלילתו

בנוסף, נבחר את k להיות m.

1. תחילה, R רצה בזמן פולינומי
   - קל, בניית מטריצת שכנויות ב-$\mathcal{O}\left(m^2\right)$
2. אם φ ספיקה אז ב-G יש קליקה בגודל k
   - נסמן ב-α השמה שמספקת φ
   - φ בצורת $3-CNF$, ומכאן בכל פסוקית α מספקת לפחות ליטרל אחד
   - נבחר ליטרל אחד מסופק מכל פסוקית לקבוצה S
     - ב-S יש $m = k$ איברים
     - יש צלע בין שני כל קודקודים (כל אחד מפסוקית שונה, ולא יכול להיות ליטרל ושלילתו)
   - מכאן, S היא קליקה בגודל k ב-G.
3. אם ב-G קליקה מגודל k אז φ ספיקה
   - נסמן ב-S קליקה ב-G מגודל $k = m$
   - מהגדרת הגרף, S לא כוללת קודקודים מאותה השורה, ולא כוללת ליטרל ושלילתו
   - נגדיר השמה α שמתאימה לכל משתנה x_i ערך אמת:
     - אם x_i מופיע בחיוב ב-S, נגדיר $\alpha\left(x_i\right) = T$
     - אם x_i מופיע בשלילה ב-S, נגדיר $\alpha\left(x_i\right) = F$
     - אם x_i לא מופיע ב-S, נגדיר $\alpha\left(x_i\right) = T$ (בעיקרון dont-care, אך כל השמה לא תשנה)
   - α מספקת את φ:
     - מאחר ו-$|S| = m$, בכל פסוקית יש ליטרל שמופיע ב-S
     - הגדרנו את α בצורה שהליטרל יסופק, וכך הפסוקית מסתפקת

בסך הכל, בנינו רדוקציה פולינומית מ-$3-SAT$ ל-$CLIQUE$, וכך $3-SAT \leq_p CLIQUE$. □

הערה. הרדוקציה לא יודעת לבדוק האם φ ספיקה או האם קיימת קליקה, ורק יודעת <u>לתרגם</u> מנוסחא לגרף כך שקיימת קליקה אמ"מ הנוסחא ספיקה.

דוגמה. $VERTEX-COVER=\{\langle G,k\rangle\mid k$ גרף, G ו-$k\in\mathbb{N}$ יש כיסוי בצמתים מגודל G-וב $k\}$
$V'\subseteq V$ הוא כיסוי בצמתים של גרף $G=(V,E)$ אם:

$$\forall e=(u,v)\in V: u\in V\vee v\in V$$

טענה. $3-SAT\leq_p VERTEX-COVER$

הוכחה. נבנה רדוקציה פולינומית R מ-$3-SAT$ ל-$VERTEX-COVER$, שמקבלת נוסחא φ בצורת $3-CNF$ ופולטת גרף G ומספר k.
בהינתן נוסחא φ בצורת $3-CNF$, R תבנה גרף G ומספר $k=n+2m$.

- נחבר קשת בין כל ליטרל ושלילתו
- לכל פסוקית (l_1,l_2,l_3) נוסיף משולש:
  - 3 צמתים חדשים עבור l_1,l_2,l_3 שמחוברים לליטרלים המקוריים של הגרף.
  - - הקשתות $(l_1,l_2),(l_1,l_3),(l_2,l_3)$.
  - כיסוי בצמתים במשולש מכיל לפחות 2 צמתים.
- מכל משולש נבחר $2/3$ צמתים - הצומת שלא נבחר מייצג את הליטרל המסופק של הפסוקית.
  - הקשתות של המשולש יכוסו ע"י שני הצמתים האחרים במשולש
  - הקשת לליטרל המקורי תכוסה ע"י הליטרל המקורי בעצמו

1. R רצה בזמן פולינומי (בניית הגרף עולה $\mathcal{O}\left(m^2\right)$)
2. אם φ ספיקה אז ב-G כיסוי מגודל $k=n+2m$
   - תהי α השמה שמספקת את φ.
     - דרושים לפחות n קודקודים עבור כל השורות המקוריות
     - דרושים לפחות $2m$ קודקודים למשולשים, 2 מכל פסוקית
   - עבור כל השורות - קודקודי המשתנים, נבחר קודקוד אחד מכל צלע
     - אם $\alpha(x_i)=T$ נבחר ב-x_i, אחרת ב-$\neg x_i$
   - לכל פסוקית c קיים ליטרל (אחד לפחות) כך ש-α מספקת אותו
     - נבחר לכיסוי את שני הקודקודים האחרים במשולש
   - כל קשת שמחברת בין קודקוד משתנה ולעותק שלו בפסוקית תכוסה ע"י קודקוד המשתנה המקורי
3. אם ב-G כיסוי בגודל $k=n+2m$ אז φ ספיקה
   - יהי S כיסוי בגודל $k=n+2m$
   - S מכסה את כל צלעות המשתנים וצלעות המשולשים
     - מכאן, ב-S קודקוד בכל שורה ושני קודקודים בכל משולש

- נגדיר השמה α באופן הבא:

$$\alpha\left(x_i\right)=\begin{cases}T & x_i\in S\\ F & \neg x_i\in S\end{cases}$$

- α מספקת את S - תהי c פסוקית כלשהי:
  - נסתכל על הצלע בין ליטרל l, שלא נבחר במשולש, ובין קודקוד המשתנה שלו
  - הצלע לא מכוסה ע"י שני הקודקודים במשולש, וכך בהכרח מכוסה ע"י קודקוד המשתנה l
- מכאן, לפי הצורה בה הגדרנו את α, מתקיים כי $\alpha\left(l\right)=T$.
- בסך הכל, קיבלנו כי הליטרל מסתפק וכך הפסוקית מסתפקת.

$\square$

6 שפות NP שלמות

הגדרה. שפה L תיקרא NP-שלמה אם:

1. $L\in NP$ (קל להראות)
2. לכל שפה $L'\in NP$ מתקיים $L'\leq_p L$

משפט. אם שפה L היא NP-שלמה אז מתקיים:

$$NP=P\Longleftrightarrow L\in P$$

משפט Cook-Levin:

משפט. $3-SAT$ היא NP-שלמה.

מסקנה. מספר מסקנות ממשפט Cook-Levin

1. $NP-P\Longleftrightarrow 3-SAT\in P$.
2. מאחר ו-$3-SAT\leq_p HAM-PATH$, אז גם $HAM-PATH$ היא NP-שלמה.
3. מאחר ו-$3-SAT\leq_p CLIQUE$, אז גם $CLIQUE$ היא NP-שלמה.
4. מאחר ו-$3-SAT\leq_p VERTEX-COVER$, אז גם $VERTEX-COVER$ היא NP-שלמה.

הערה. כעת נוכל לשלב את העובדה ש-$3-SAT$ היא NP-שלמה עם הרדוקציות שעשינו קודם לכן.
למשל, מאחר וראינו כי $3-SAT \leq_p CLIQUE$, ניתן להסיק כעת שגם $CLIQUE$ כי NP-שלמה.

6.1 $K-COLOR$

הגדרה. גרף G הוא K-צביע אם ניתן לצבוע את קודקודיו ב-K צבעים כך שלכל קשת שני קודקודיה בצבעים שונים.

$$K-COLOR = \{\langle G\rangle \mid \text{צביע-}K \text{ גרף } G\}$$

- משולש (וכל מעגל באורך אי זוגי) דורש 3 צבעים
- קליקה דורשת שכל צומת יהיה בצבע אחר
- כל גרף מישורי הוא 4-צביע

הערה. למה $K-COLOR$ היא בעיה חשובה?
צריך לסדר מערכת שעות שבה לא יהיה סטודנט שרשום לשני קורסים שונים באותה השעה.

- קודקודים - קורסים
- קשת - סטודנט שרשום לשני הקורסים
- אם הגרף צביע ב-K צבעים אז יש מערכת בת K שעות שונות

משפט. $K-COLOR$ היא NP-שלמה.

משפט. $3-COLOR$ היא NP-שלמה.

הערה. קל להוכיח ש-$3-COLOR \in NP$: מוודא פולינומי של צביעה חוקית.

מסקנה. אם אפשר לבדוק האם גרף 3-צביע אז ניתן למצוא מסלול המילטוני, לספק נוסחאות ולמצוא כיסוי בצמתים בזמן פולינומי.

משפט. $2-COLOR \in P$.

הוכחה. גרף 2-צביע אמ"מ הוא דו-צדדי, ניתן לבדוק בפשטות באמצעות DFS. □

טענה. $3-SAT \leq_p 3-COLOR$.

הוכחה. נבנה רדוקציה פולינומית R מ-$3-SAT$ ל-$3-COLOR$.
נבנה גרף בשם Gadget, ונבצע אותו עם 3 צבעים: $\{T, F, A\}$.

גרף גרף איור איור קודקודים וזה

איור 38: גרף Gadget

- ניתן לצבוע <u>קודקודי קצה</u> ב-$\{T, F\}$
- ניתן לצבוע <u>קודקודי ביניים</u> ב-$\{F, A\}$

נבנה גרף G בהגדרה: נסמן את מספר המשתנים ב-φ ב-n, ואת מספר הפסוקיות ב-m.

- נתחיל ממשולש בסיסי: בו קודקוד אחד מסומן ב-A, שני ב-F והשלישי ב-T.
- לכל משתנה x_i נוסיף את הקודקודים $x_i, \neg x_i$, ואת הקשתות:

$$(x_i, \neg x_i), (x_i, A), (\neg x_i, A)$$

  - זהו רכיב ההשמה, בדומה להוכחה ב-VC.
- אם נצבע את הגרף ב-3 צבעים נקבל השמה, יש להראות שהיא מספקת לכל פסוקית $c = (l_1 \vee l_2 \vee l_3)$.
  - עבור הפסוקית $(x_1, \vee \neg x_2, \vee x_7)$, למשל, נוסיף עותק של Gadget בו קודקודי הליטרלים משמים כקודקודי הקצה
    * כלומר, נוסיף רק את הקודקודים שאינם קצה ונחברם בהתאם
  - בנוסף, נוסיף צלעות מקודקודי הביניים ל-T של המשולש הבסיסי (כך הם יהיו צבועים ב-A או F)
  - וזהו!
- קל לראות כי הרדוקציה פולינומית - הגרף ההתחלתי בגודל n, וכל פסוקית מוסיפה Gadget בגודל $\mathcal{O}(1)$.

יש להוכיח כי הגרף צביע אמ"מ הנוסחא ספיקה - רעיון ההוכחה:

- אם הגרף צביע (נסמן את הצבעים ב-$\{T, F, A\}$), כל ליטרל ושלילתו צבועים ב-T, F.
  - כך, לכל פסוקית קודקודי הקצה צבועים ב-T, F
  - קודקודי הביניים צבועים ב-F, A, כי הם מחוברים ל-T של המשולש הבסיסי.
  - מכאן, חייב להיות קודקוד קצה אחד עם T.
- אם הנוסחא ספיקה, נצבע את רכיב ההשמה ב-T, F בהתאם להשמה.
  - נצבע את המשולש הבסיסי בהתאם.
  - בכל Gadget קודקודי הקצה כבר נצבעו, וקודקודי הביניים הם A מלבד לקודקוד אחד, שמחובר לקודקוד כזה שצבעו T.
    * חייב להיות אחד כזה - בכל פסוקית יש ליטרל מסופק
    * נצבע קודקוד ביניים כזה ב-F
  - את שאר קודקודי ה-Gadget תמיד אפשר לצבוע (כי קודקודי הביניים מכילים גם F וגם A).

בסך הכל, בנינו רדוקציה פולינומית מ-$3-SAT$ ל-$3-COLOR$, וכך $3-SAT \leq_p 3-COLOR$ ו-$3-COLOR$ היא NP-שלמה. $\square$

משפט. כל שפה $L \in NP$ היא כריעה.

הוכחה. ל-L קיים מוודא פולינומי V, ומתקיים שלכל x:

$$x \in L \Longleftrightarrow \exists y : |y| \leq |x|^c, v(x,y) = 1$$

נבנה מ"ט דטרמיניסטית שמכריעה את השפה. בהינתן קלט x, נעבור על כל ה-y-ים האפשריים מאורך לכל היותר $|x|^c$, ונריץ לכל אחד מהם את $v(x,y)$.
אם באחת ההרצות $v(x,y) = 1$ קבל, ואחרת דחה. □

<u>בעיה:</u> זמן אקספוננציאלי.

6.2 סיבוכיות זמן אקספוננציאלית

$$P = \bigcup_{c=1}^{\infty} TIME(n^c)$$

$$NP = \bigcup_{c=1}^{\infty} NTIME(n^c)$$

$$EXP = \bigcup_{c=1}^{\infty} TIME\left(2^{n^c}\right) = \{L(M) = L \text{ -ו } t(n) \leq 2^{n^c} \text{ קיים קבוע } c \text{ ומ"ט דטרמיניסטית } M \text{ שרצה בזמן}\}$$

$$\Longrightarrow P \subseteq NP \subseteq EXP$$

<u>שאלה:</u> האם $P \neq NP$? לא יודעים.

<u>שאלה:</u> האם $NP \neq EXP$? לא יודעים.

<u>יודעים:</u> לא ייתכן ש-$P = NP$ וגם $NP = EXP$, מאחר ו-$P \neq EXP$.

משפט. $P \neq EXP$.

משפט. לכל פונקציית זמן "תקינה" $t(n)$ קיימת שפה L שמקיימת:

1. $L \in TIME\left(t^3(n)\right)$
2. $L \notin TIME(t(n))$

הערה. $t(n)$ היא תקינה אם קיימת מ"ט שיכולה לחשב את $t(n)$.
אם נציב $t(n) = 2^n$ נקבל שפה ב-EXP אך לא ב-P.

<u>רעיון:</u> נבנה מ"ט A שרצה בזמן $t^3(n)$ ו-$L = L(A)$.

- בהינתן קלט, A תתייחס אליו כקידוד של מ"ט M ותריץ את M על $\langle M \rangle$ למשך $t(n)$ צעדים (בזמן $t^2(n)$):
  - אם M קיבלה את $\langle M \rangle$, A תדחה.
  - אם M דחתה את $\langle M \rangle$, A תקבל.

- אם M לא עצרה, A תפלוט פלט כלשהו (dont-care)

<u>טכניקה:</u> לכסון.

- נניח שקיימת מ"ט M שרצה בזמן $t(n)$ ומכריעה את $L(A)$.
- מה יקרה כשנריץ את A על הקלט M?
  - M תריץ את $\langle M\rangle$ למשך $t(n)$ צעדים, ובוודאות תעצור.
  - עם זאת, A מחזירה הפוך ממה ש-M מחזירה על $\langle M\rangle$, למרות שהן אמורות לעשות את אותו הדבר (שתי מ"ט שמכריעות את אותה השפה)
- הגענו לסתירה!

הערה. במקרה זה, יכלנו להסתדר גם עם $t^2(n)$ זמן ולא $t^3(n)$.

- $TIME(t(n))$ מוגדרת ע"י $\mathcal{O}$.
- כך, ייתכן שהמכונה תרוץ בזמן $c\cdot t(n)$ וכך נרוץ t^2 צעדים במקום t: סה"כ $t^3(n)$ זמן.

חלק III
נספחים

1 תרגילים

1.1 אוסף טענות

הוכח או הפרך כל אחת מהטענות.

1. לכל L, $L \subseteq L^2 \iff \varepsilon \in L$.
2. לכל L_1, L_2: $L_2 = (L_1 \cup L_2) \setminus L_1$.
3. L^* אינסופית לכל L.
4. לכל L, $(L \setminus \varepsilon)^* = L^* \setminus \{\varepsilon\}$.
5. תהי שפה L מעל א"ב לא ריק. אזי:
   (א) $(L^*)^* = L^*$.
   (ב) $L^* \circ L^* = L^*$.

1.2 אוטומט סופי דטרמיניסטי

1. בנה אס"ד לשפות הבאות מעל $\Sigma = \{0,1\}$.

   (א) כל המילים שמתחילות ב-1 וגם שמסתיימות ב-0.

   (ב) $L = \{\varepsilon\}$.

   (ג) מילים שלא מכילות את הרצף 110.

2. יהי א"ב $\Sigma = \{0,1\}$.

   (א) בנה אס"ד עבור השפה שמכילה את כל המילים שמכילות גם 0 וגם 1.

   (ב) כתבו במפורש את δ של האוטומט (טבלה).

3. נתונה השפה $L = \left\{w \in \{0,1\}^* \mid 0011 \in w\right\}$.

   (א) תכננו אס"ד שמקבל את L.

   (ב) תכננו אס"ד שמקבל את $\overline{L}$.

4. תכננו אלגוריתם שמקבל אס"ד $M = (Q, q_0, \Sigma, \delta, F)$, ומכריע:

   (א) האם $L(M) = \emptyset$?

   (ב) האם $L(M) = \Sigma^*$?

5. יהי $\Sigma = \{a, \ldots, z\}$, ותהי $L_1 = \{w \mid \exists \sigma \in \Sigma : \#_\sigma \in w \geq 2\}$.

   (א) האם L_1 סופית?

   (ב) בנו אס"ד שמקבל את L_1 (הדרכה: קל יותר להגדיר פורמלית).

   (ג) בנו אס"ד שמקבל את $L_2 = \{w \mid \forall \sigma \in \Sigma : \#_\sigma \in w \geq 2\}$.

   (ד) בנו אס"ד שמקבל את $L_3 = \{w \mid \forall \sigma \in \Sigma : \#_\sigma \in w = 2\}$.

   (ה) האם L_3 סופית? ואם כן - כמה מילים יש בה?

6. יהיו $L_1 = \{w \mid \#_1 \in w \equiv 1 \mod 3\}, L_2 = \{w \mid |w| = 0 \mod 2\}$.

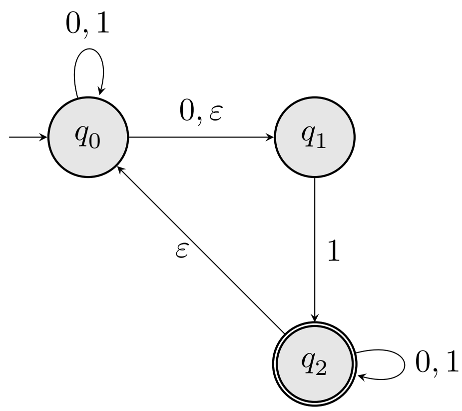
   (א) בנה אס"דים המקבלים את L_1, L_2.

   (ב) בנה את אוטומט החיתוך המקבל את $L_1 \cap L_2$.

1.3 אוטומט סופי לא דטרמיניסטי

1. בנו אסל"ד שמקבל את השפה $L = \{w \mid$ התו השני של w הוא $1\}$.

2. בהינתן האסל"ד הבא, בנה אס"ד שמקבל את אותה השפה.



3. לכל שפה L, נגדיר את השפה $e(L) = \{w \circ (|w| \mod 2) \mid w \in L\}$. הוכיחו כי L רגולרית $\Longleftarrow$ $e(L)$ רגולרית.

4. בהינתן שפה L מעל א"ב סופי, נגדיר שפה L' שמתקבלת מ-L באופן הבא:

$$L' = \{u \mid \exists w \in L : \text{מתקבלת מ-} w \text{ ע"י מחיקת לכל היותר 100 אותיות } u\}$$

אם L רגולרית, האם L' רגולרית?

5. יהי $\Sigma = \{a, b, c\}$ א"ב, ותהי $L = \{w \mid$ התו האחרון ב-w מופיע לפחות פעמיים ב-$w\}$. בנו אסל"ד שמקבל את L.

1.4 ביטויים רגולריים

1. תהי $L = \{w \mid \#_1 \in w = 3\}$.
   (א) כתבו ביטוי רגולרי לשפה L.
   (ב) כתבו ביטוי רגולרי לשפה $\overline{L}$.

2. נתון הביטוי הרגולרי $\left(0 \circ (0 \cup 1)^* \circ 0\right) \cup \left(1 \circ (0 \cup 1)^* \circ 1\right)$. מהי השפה?

3. כתבו ביטוי רגולרי לשפה $L = \{w \mid$ לא מכילה 101 $w\}$.

4. כתבו ביטוי רגולרי עבור השפה $\{w \mid$ לא מכילה את הצירוף 101 $w\}$ מעל $\{0,1\}^*$.

1.5 למת הניפוח ומשפט מייהיל נרוד

הוכיחו כי השפות הבאות אינן רגולריות.

1. $L = \left\{w \in \{0,1\}^* \mid \#_0 = \#_1\right\}$.
2. $L = \left\{w \in \{a,b,c\}^* \mid \#_a + \#_b = \#_c\right\}$.
3. $L = \left\{0^{n^3} \mid n \in \mathbb{N}\right\}$.
4. $L = \left\{w \in \{a,\dots,z\} \mid \text{קיימת אות שמופיעה לפחות } \frac{|w|}{2} \text{ פעמים}\right\}$.
5. $L = \left\{ww \mid w \in \{0,1\}^*\right\}$.
6. $L = \{w \mid \text{אינה שרשור של שני פלינדרומים } w\}$.

1.6 אוטומט מחסנית ודח"ה

1. תכננו אוטומטי מחסנית <u>בעלי מצב אחד</u> לשפות הבאות:
   (א) $L = \left\{w \in \{0,1\}^* \mid \#_0 = \#_1\right\}$.
   (ב) $L = \left\{w \in \{0,1\}^* \mid \#_1 \in u \leq \#_0 \in uw \text{ של } u \text{ לכל רישא } \right\}$.
2. תכננו א"מ ודח"ה עבור השפה $L = \{a^n b^m c^{n-m} \mid n \geq m \geq 0\}$.
3. נתונה השפה $L = \left\{w \in \{0,1\}^* \mid \#_0(w) = \#_1(w)\right\}$. האם השפה ח"ה? אם כן - בנו לה דח"ה.
4. הוכח/הפרך כי לכל א"מ M קיים א"מ M' שעבורו א"ב המחסנית הוא בעל 2 תווים בלבד, כך ש-$L(M) = L(M')$.
5. תכננו א"מ לשפות הבאות.
   (א) $L = \{a^n b^m c^{n+m} \mid n,m \geq 0\}$.
   (ב) $L = \left\{a^n b^m c^{\frac{n+m}{2}} \mid n+m \equiv 0 \mod 2\right\}$.
   (ג) $L = \{a^n b^m c^k \mid k \in \{n,m\}\}$.
   (ד) $L = \left\{w \in \{0,1\}^* \mid \#_0 = 2 \cdot \#_1\right\}$.
6. יהי $\Sigma = \{0,\dots,9\}$. עבור $w \in \Sigma^*$ נגדיר $e(w)$ להיות המילה המתקבלת מ-w ע"י השמטת התווים בעלי ערכים אי זוגיים. (למשל, $e(007342982) = 004282$). עבור שפה L מעל Σ, נגדיר $e(L) = \{e(w) \mid w \in L\}$, הוכח כי אם L ח"ה אז $e(L)$ ח"ה.
7. הוכיחו/הפריכו את הטענות הבאות.
   (א) אם L_1, L_2 שפות ח"ה אז $L_1 \circ L_2$ ח"ה.
   (ב) אם L ח"ה אז $\text{Prefix}(L)$, שפת הרישות של מילים ב-L, ח"ה.
   (ג) אפ L ח"ה ואינה רגולרית אז $\text{Prefix}(L)$ אינה רגולרית.
8. תכננו דח"ה לשפות הבאות.

(א) $L=\{a^nb^mc^{n+m}\mid n,m\geq 0\}$.

(ב) $L=\left\{a^nb^mc^{(n+m)/2}\mid n+m\equiv 0 \mod 2\right\}$.

(ג) $L=\left\{a^nb^mc^k\mid k\in\{n,m\}\right\}$.

(ד) שפת המילים המאוזנות.

1.7 שפות ח"ה

הוכיחו כי השפות הבאות אינן ח"ה.

1. $L=\left\{a^nb^nc^i\mid i\leq n\right\}$.
2. $L=\{0^t\mid$ ראשוני $t\}$.
3. $L=\left\{0^i1^{i^2}\mid i\geq 0\right\}$.
4. $L=\left\{s\circ s^R\circ s\mid s\in\{0,1\}^*\right\}$.

1.8 מכונת טיורינג

1. בנו מ"ט לשפה $\{0^n1^n\mid n\geq 1\}$.
2. בנו מ"ט לשפה $\{w\mid \#_0(w)=\#_1(w)\}$.

1.9 רדוקציות

1. הוכח כי אם $L,\overline{L}$ ניתנות לקבלה, אז שתיהן כריעות.
2. תהי $L=\left\{\langle M\rangle\mid \exists w: w,w^R\in L(M)\right\}$.

   (א) האם L ניתנת לקבלה?

   (ב) האם L כריעה?
3. תהי $L=\{\langle M\rangle\mid$ לא מקבלת אף תכנית C $M\}$.

   (א) האם L ניתנת לקבלה?

   (ב) האם L כריעה?
4. תהי $EMPTY_{CFG}=\{G\mid L(G)=\emptyset$ הוא דח"ה שמקיים $G\}$.

   (א) האם השפה ניתנת לקבלה?

   (ב) האם השפה כריעה?
5. האם קיימת מ"ט T שמקבלת $\langle M\rangle$ ופולטת את כל קידודי מ"ט $\langle M_1\rangle,\langle M_2\rangle,\ldots$ כך ש-

   $$L(M_1)=L(M_2)=\cdots=L(M)$$
6. עבור כל שפה מהבאות, קבעו מהי המחלקה הקטנה ביותר אליה היא שייכת:

(א) $L_1 = \left\{x\#x^R z x^R \mid z \in \{0,1\}^*\right\}$

(ב) $L_2 = \{a^n b^m \mid 2n \le 3m \vee 2m \le 3n\}$

(ג) $L_3 = \{\langle M, q\rangle \mid$ M מ"ט ו-q מיותר ב-$M\}$

(ד) $L_4 = \{\langle M\rangle \mid \langle M\rangle \in L(M)\}$

1.10 P לעומת NP

1. הוכח סגירות של איחוד עבור שפות ב-NP.
2. נגדיר $IND-SET = \{(G,k) \mid$ ב-G קבוצה ב"ת בגודל $k\}$. הוכח:

   (א) $CLIQUE \le_p IND-SET$

   (ב) $IND-SET \in NP$
3. נגדיר $DHPATH = \{\langle G\rangle \mid$ G מכוון ויש בו מסלול המילטוני$\}$. הראו כי:

   (א) $DHPATH \in NP$

   (ב) $HAM-PATH \le_p DHPATH$

   (ג) $DHPATH \le_p HAM-PATH$
4. הוכיחו כי $4-SAT \le_p 3-SAT$ ו-$3-SAT \le_p 4-SAT$

2 פתרונות

2.1 אוסף טענות

1. נוכיח את הטענה: כיוון אחד כל מילה לשרשר עם ε, כיוון שני מילה קצרה ביותר

2. נפריך את הטענה:

$$L_1 = \{a, b\}, L_2 = \{b, c\}$$

$$(L_1 \cup L_2) \setminus L_1 = \{c\} \neq \{b, c\} = L_2$$

3. נפריך את הטענה: עבור $L = \emptyset, \{\varepsilon\}$ מתקיים $L^* = \{\varepsilon\}$ וסופית.

4. נפריך את הטענה: הטענה שגויה לכל L! $\varepsilon \in (L \setminus \varepsilon)^*$ ו-$\varepsilon \notin L^* \setminus \{\varepsilon\}$, ולכן לא מתקיים השוויון.

5. נוכיח את שתי הטענות.

(א) נוכיח באמצעות הכלה דו כיוונית:

i. $L^* = (L^*)^1 \subseteq (L^*)^*$.

ii. תהי $w \in (L^*)^*$. אזי $\exists k \mid w = a_1 \circ \ldots \circ a_k, a_i \in L^* \Longrightarrow \exists l_1, \ldots, l_k \mid a_i = x_{l_1} \circ \ldots \circ x_{l_k}, x_i \in L$. כלומר, $w \in L^*$ ו-$(L^*)^* \subseteq L^*$.

iii. וכך, $L^* = (L^*)^*$.

(ב) נוכיח באמצעות הכלה דו כיוונית:

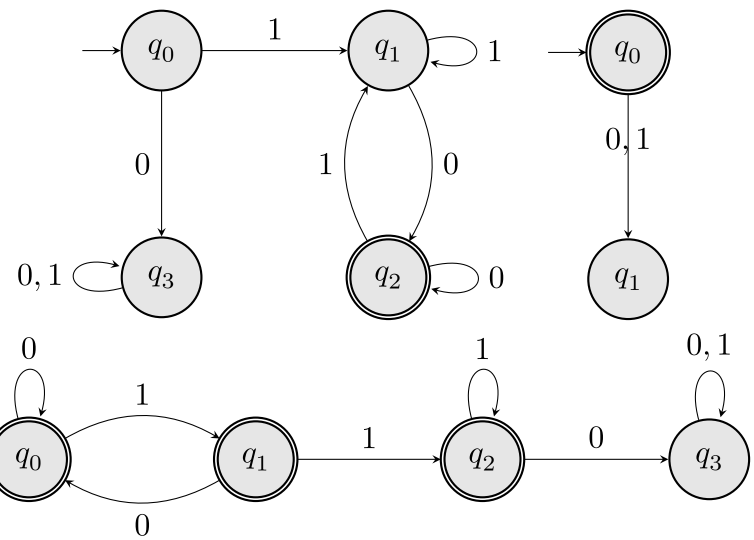
i. יהי $x \in L^*$, אזי $x = x \circ \varepsilon \subseteq L^* \circ L^*$. כלומר $L^* \subseteq L^* \circ L^*$.

ii. יהי $x \in L^* \circ L^*$. בפרט, $x = x_1 \circ x_2$ כך ש-$x_1, x_2 \in L^*$. מאחר ו-$x_1 \in L^{m_1}, x_2 \in L^{m_2}$, נקבל כי $x \in L^{m_1+m_2} \subseteq L^*$.

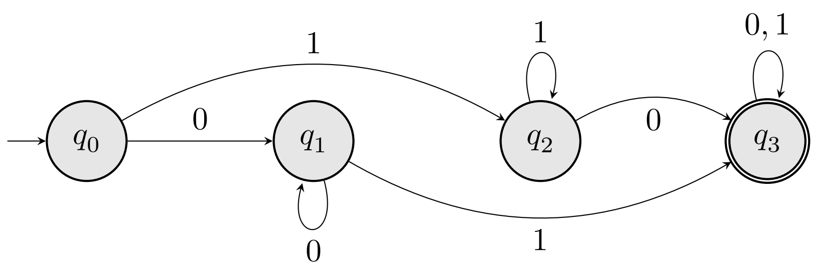
iii. וכך, $L^* \circ L^* = L^*$.

2.2 אוטומט סופי דטרמיניסטי

1. נבנה אס"ד לשפה (א - שמאל למעלה, ב - ימין למעלה, ג - למטה).



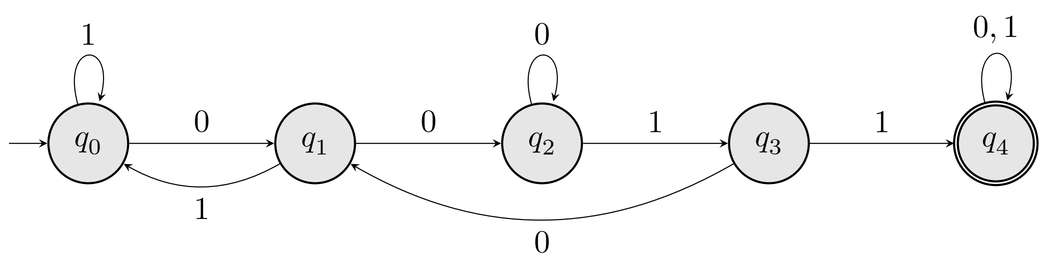
2. השפה היא $L = \left\{ w \in \{0,1\}^* \mid 0 \in w \wedge 1 \in w \right\}$.

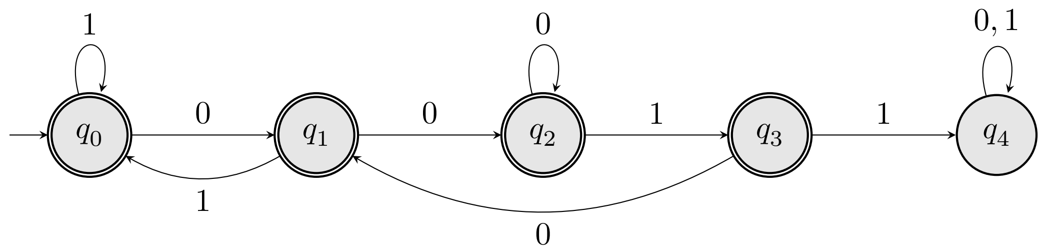


δ	0	1
q_0	q_1	q_2
q_1	q_1	q_3
q_2	q_3	q_2
q_3	q_3	q_3

טבלה 12: פונקציית המעברים של האוטומט

3. מלמעלה למטה: אס"ד שמקבל את L, אס"ד שמקבל את $\overline{L}$.





4. ניעזר באלגוריתם dfs.

   (א) נריץ dfs על האוטומט מ-q_0, ונחזיר האם F ישיגה.

   (ב) נהפוך את כל המצבים ונריץ על האוטומט הנוכחי את האלגוריתם מסעיף א'.

5. נפתור.

   (א) השפה אינה סופית - מכילה את $\{a^n \mid n \geq 2\}$, שאינה סופית בעצמה.

   (ב) נגדיר כל מצב באמצעות מחרוזת באורך 26, בה כל תו הוא 0, 1 או 2 (מייצג 2 ומעלה). כל תו במחרוזת מייצג את מספר ההופעות הנוכחי של האות באותו האינדקס. המצבים המקבלים הם כל אלו שיש בהם לפחות 2 אחד.

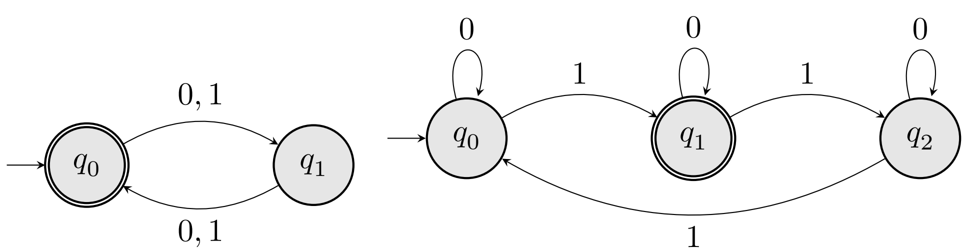
   (ג) בדומה לסעיף הקודם, אך כאן המצב המקבל היחיד הוא $q_{22\ldots2}$.

   (ד) באופן דומה לסעיף הקודם, אך כעת ברגע שנוספת אות שלה כבר שני מופעים, נעבור למצב בור.

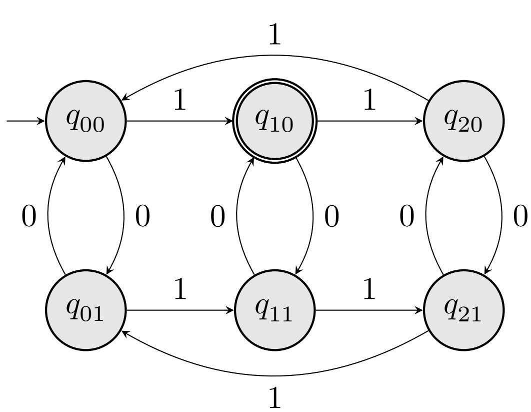
   (ה) כן - המחרוזת באורך 52 ומכילה 2 אותיות מכל סוג. לכן, יש בה $3^{26}/\binom{52}{2,2,\ldots,2} = \frac{3^{26}\cdot 52!}{2^{26}}$ מצבים.

6. נפתור.

   (א) מימין לשמאל: אס"דים המקבלים את השפות L_1 ו-L_2 בהתאמה.

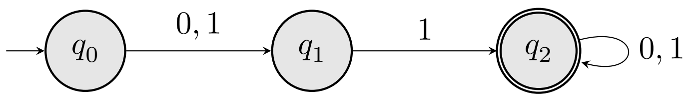


   (ב) נבנה אוטומט מכפלה.

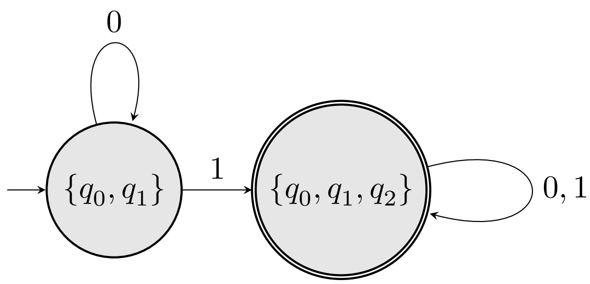


2.3 אוטומט סופי לא דטרמיניסטי

1. נבנה את האסל"ד.



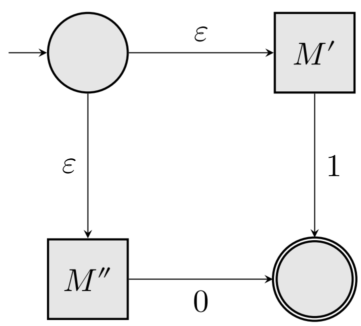
2. שרטוט האס"ד המתקבל (לאחר הצמצום).



3. נוכיח כי אם L רגולרית אז $e\left(L\right)$ רגולרית. מאחר ו-L רגולרית נוכל להגדיר שני אוטומטי-עזר M', M'':

$$L\left(M'\right) = L \cap \{w \mid |w| \equiv 1 \mod 2\}$$
$$L\left(M''\right) = L \cap \{w \mid |w| \equiv 0 \mod 2\}$$

מכאן, נבנה את האוטומט הבא.



אוטומט זה מקבל את $e\left(L\right)$, ולכן $e\left(L\right)$ רגולרית.

4. כן - נבנה אסל"ד ל-L' באופן הבא:
ניקח 101 עותקים של האס"ד של L, כך שהמצב ההתחלתי יהיה זה של עותק מספר 0.
לכל מעבר מהצורה $q_i \to q_j$, בכל עותק של האס"ד, נוסיף מעבר ε מ-q_i בעומק נוכחי ו-q_j בעומק הבא.
המצבים המקבלים יהיו המצבים המקבלים בכל העותקים.

2.4 ביטויים רגולריים

1. נפתור.

   (א) $0^* \circ 1 \circ 0^* \circ 1 \circ 0^* \circ 1 \circ 0^*$.

   (ב) ביטוי אחד (וארוך):

$$(0^*) \cup (0^* \circ 1 \circ 0^*) \cup (0^* \circ 1 \circ 0^* \circ 1 \circ 0^*)$$
$$\cup \left((0 \cup 1)^* \circ 1 \circ (0 \cup 1)^* \circ 1 \circ (0 \cup 1)^* \circ 1 \circ (0 \cup 1)^* \circ 1 \circ (0 \cup 1)^*\right)$$

2. אוסף כל המילים שמתחילות ונגמרות באותה האות ואורכן לפחות 2.

3. $0^* \circ \left(1^* \circ (000)^*\right)^* \circ 1^* \circ 0^*$.

4. $0^* \circ (1^* \circ 00 \circ 0^* \circ 1^*)^* \circ 1^* \circ 0^*$.

2.5 למת הניפוח ומשפט מייהיל נרוד

1. $L = \left\{w \in \{0,1\}^* \mid \#_0 = \#_1\right\}$: נובע מסגירות של שפות רגולריות:
   - נניח בשלילה כי L רגולרית.
   - אזי, השפה $L' = L(a^* b^*)$ היא רגולרית.
   - נגדיר $L'' = L' \cap L = \{a^n b^n \mid n \geq 0\}$.
     - מצד אחד, נובע מסגירות של שפות רגולריות תחת חיתוך ש-L'' רגולרית.
     - מצד שני, $L'' = \{a^n b^n \mid n \geq 0\}$ ואינה רגולרית.
   - הגענו לסתירה ולכן L אינה רגולרית.

2. $L = \left\{w \in \{a,b,c\}^* \mid \#_a + \#_b = \#_c\right\}$.

   (א) נסתכל על המילה $a^{n_0} b^{n_0} c^{2n_0}$. נשים לב כי y לא מכיל אף c, וכאשר ננפח נוסיף a, bים בלבד - התנאי לא יישמר ולכן השפה אינה רגולרית.

   (ב) נסתכל על קבוצת המילים $\{a^i\}_{i=1}^{\infty}$. בין כל a^i, a^j נראה כי הם אינם שקולים עם $z = c^i$. קיבלנו כי $\#_L = \infty$ ולכן L אינה רגולרית.

3. $L = \left\{0^{n^3} \mid n \in \mathbb{N}\right\}$.

   (א) נסתכל על המילה $0^{n_0^3}$. עבור $|y| = n_0$ נקבל שאורך המילה xy^2z הוא $n_0^3 + n_0$. עם זאת, המילה הקצרה ביותר שארוכה מ-n_0^3 היא באורך $(n_0+1)^3 = n_0^3 + 6n_0 + 1 > n_0^3 + n_0$, וכך $xy^2z \notin L$ ו-L אינה רגולרית.

   (ב) כל סדרה אינסופית תעבוד! נבחר סדרה של מילים השונות זו מזו - לכל מילה יש השלמה יחידה ל-x^3 הקרוב ביותר אליה, ולכן כל שתי מילים לא שקולות ו-L אינה רגולרית.

4. $L = \left\{ w \in \{a, \ldots, z\} \mid \right.$ קיימת אות שמופיעה לפחות $\frac{|w|}{2}$ פעמים$\left.\right\}$.

(א) נסתכל על המילה $a^{n_0}b^{n_0}c^{2n_0}$. נשים לב כי y לא מכיל אף c, וכאשר ננפח נוסיף a, bים בלבד - התנאי לא יישמר ולכן השפה אינה רגולרית.

(ב) נסתכל על קבוצת המילים $\left\{a^ib^ic^i\right\}_{i=1}^{\infty}$: הן נבדלות אחת מהשנייה ע"י c^i.

5. $L = \left\{ ww \mid w \in \{0,1\}^* \right\}$.

(א) נסתכל על המילה $a^{n_0}ba^{n_0}b$. ניפוח יוסיף aים בלבד לחלק הראשון, וכך המילה לא תהיה מהצורה $w \circ w$ - בסתירה ללמת הניפוח, ולכן L אינה רגולרית.

(ב) נסתכל על קבוצת המילים $\left\{a^ib^i\right\}_{i=1}^{\infty}$, כל זוג מילים נבדל ב-$a^ib^i$.

6. $L = \{w \mid$ w אינה שרשור של שני פלינדרומים$\}$.

טענה. את המילה 0^m10^n11 עבור $n > m \geq 1$ לא ניתן לחלק לשני פלינדרומים.

הוכחה. נסתכל על מקום החלוקה של המילה לשני הפלינדרומים i.

- אם $i \leq m+1$ החלק השני אינו פלינדרום.
- אם החלוקה היא בתוך 0^n לא ייתכן ששני החלקים פלינדרומים.
- אם החלוקה אחרי 0^n החלק הראשון אינו פלינדרום.

$\square$

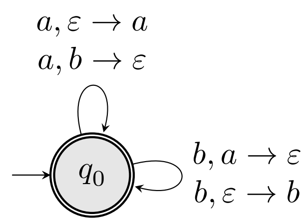
(א) כעת נשתמש בלמת הניפוח עבור המילה $0^{n_0+1}10^{n_0+1}11$. נרצה "לשאוב" חלק מהאפסים ולהגיע לייצוג $xy^0z = xz$. כך, המילה תהיה 0^m10^n11 כך ש-$n > m \geq 1$, שלא ניתנת לפירוק לשני פלינדרומים. לכן $xy^0z \notin L$ בסתירה ללמת הניפוח, ו-L אינה רגולרית.

(ב) נסתכל על סדרת המילים $\left\{0^i\right\}_{i=1}^{\infty}$. שתי מילים שונות $0^i, 0^j$ נבדלות ב-$10^{\max\{i,j\}}11$ (בה"כ $i > j$, ואז 0^i10^i11 ניתנת לחלוקה ו-0^j10^i11 לא ניתנת לחלוקה). נובע מטענת העזר שכל זוג מילים לא שקולות, וכך $\#_L = \infty$ ו-L אינה רגולרית.

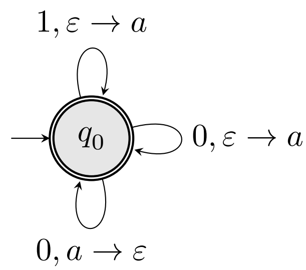
2.6 אוטומט מחסנית ודח"ה

1. אל הפתרון!

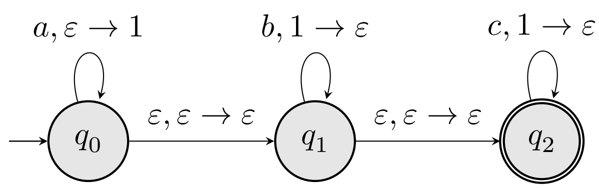
(א) $L=\left\{w\in\{0,1\}^* \mid \#_0=\#_1\right\}$.



(ב) $L=\left\{w\in\{0,1\}^* \mid\right.$ לכל רישא u של w $\left. \#_1\in u\le\#_0\in u\right\}$.



2. $L=\{a^nb^mc^{n-m}\mid n\ge m\ge 0\}$



דח"ה: $\begin{array}{l} S\to aSc\mid A \\ A\to aAb\mid\varepsilon \end{array}$.

3. השפה חסרת הקשר, דח"ה: $S\to 0S1S\mid 1S0S\mid\varepsilon$. נוכיח באמצעות הכלה דו כיוונית:

הוכחה. נוכיח כי $L(G)=L$.

תחילה, נוכיח באינדוקציה כי $L(G)\subseteq L$, כלומר כי אחרי n צעדי גזירה $\#_0=\#_1$.

- לאחר צעד גזירה בודד נקבל $\varepsilon,0S1S,1S0S$.
- אחרי צעד גזירה יכולים לקרות השינויים הבאים:
  - $S\to\varepsilon$, $\#_0$ ו-$\#_1$ לא השתנו.
  - $S\to 0S1S\mid 1S0S$, $\#_0$ ו-$\#_1$ גדלו ב-1.
- בסך הכל, קיבלנו כי האינווריאנטה נשמרת בכל צעד, וכך כל מילה שנגזרת מהדקדוק מקיימת $\#_0=\#_1$.

כעת, נוכיח כי $L\subseteq L(G)$ - באינדוקציה על מילים ב-L (מאורך זוגי בהכרח).

- בסיס האינדוקציה:

- עבור $|w| = 0$, מאחר ו-$S \to \varepsilon$ גם $\varepsilon \in L(G)$.
- עבור $|w| = 2$, המילים היחידות ב-L הן $01, 10$, שנגזרות ע"י הדקדוק ($S \Rightarrow 0S1S \Rightarrow 01$, $S \Rightarrow 1S0S \Rightarrow 10$).

- צעד האינדוקציה: נניח כי כל מילה באורך n נגזרת ע"י G, ונוכיח עבור $n+2$.
  - תהי מילה w מאורך $n+2$, נניח בה"כ כי היא מתחילה ב-0.
  - נגדיר w_j בתור $\#_0 - \#_1$ ברישא מאורך j של w, ויהי $1 \le i \le |w|$ הקטן ביותר כך ש-$w_i = 0$.
  - אם w מתחילה ב-0 אזי $w[i] = 1$ (אחרת $w_{i-1} = -1$ ו-$w_1 = 1$, וכך היה $i' < i$ שמקיים $w_{i'} = 0$).
  - מכאן, $w = 0 \circ w[2 \ldots i-1] \circ 1 \circ w[i+1 \ldots n+2]$.
  - לפי הנחת האינדוקציה $w[w \ldots i-1]$ ו-$w[i+1 \ldots n+2]$ נגזרות ע"י הדקדוק, מאחר ו-$S \to 0S1S$ גם w נגזרת ע"י הדקדוק.

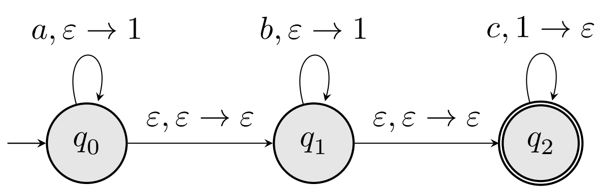
ולכן, $w \in L(G) \implies w \in L$ וכך $L \subseteq L(G)$. בסך הכל, קיבלנו כי $\boxed{L = L(G)}$. □

4. הטענה נכונה.

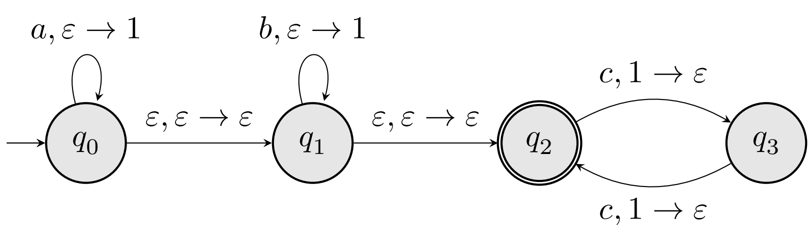
הוכחה. נייצג את א"ב המחסנית ע"י קוד חסר רישות (שרירותי) מעל $\{0,1\}$.
כל פעולת מחסנית תוחלף בשרוך פעולות עם מילות הקוד. □

5. אל הפתרון!

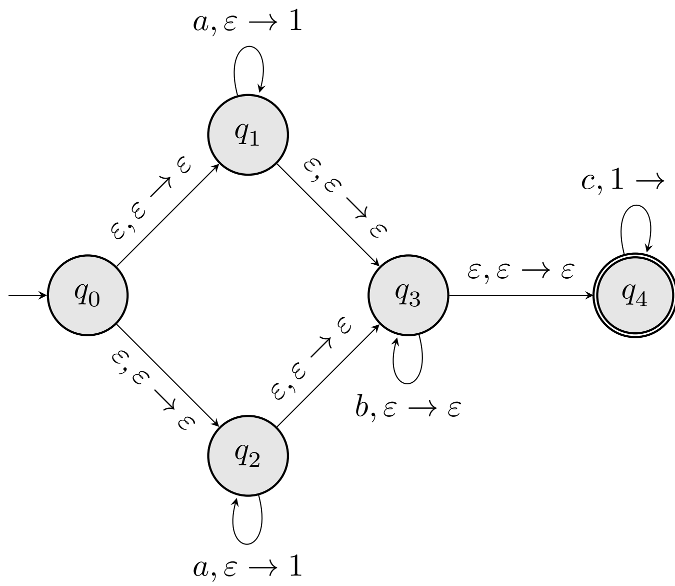
(א) $L = \{a^n b^m c^{n+m} \mid n, m \ge 0\}$.



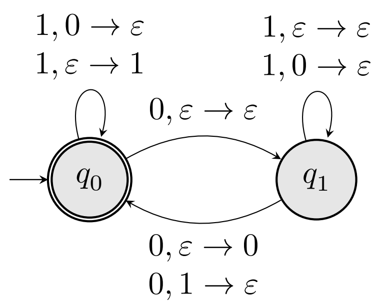
(ב) $L = \left\{a^n b^m c^{(n+m)/2} \mid n + m \equiv 0 \mod 0\right\}$.



(ג) $L = \left\{a^n b^m c^k \mid k \in \{n, m\}\right\}$.



(ד) $L = \left\{ w \in \{0,1\}^* \mid \#_0 = 2 \cdot \#_1 \right\}$.



6. ניתן לפתור בפשטות באמצעות דח"ה.

הוכחה. מאחר ו-L ח"ה, קיים א"מ M שמקבל אותה. נחליף את כל המעברים של האותיות האי זוגיות במעברי ε, ולא נשנה את המחסנית.

כלומר, נגדיר א"מ M_e של $e(L)$ באופן הבא:

כל מעבר $\sigma, \gamma_1 \to \gamma_2$, כאשר σ אי זוגי - הפוך למעבר $\varepsilon, \gamma_1 \to \gamma_2$. כדי להבדיל מעברים אלה ממעברי ε שכבר ב-L, נקרא להם מעברי ε אי זוגיים.

למה זה עובד? הכל טוב. $\square$

7. נפתור.

(א) <u>הטענה נכונה</u> - נחבר את סוף הא"מ של L_1 לתחילת הא"מ של L_2 (יש לבדוק שהגענו למחסנית ריקה במעבר בין שני האוטומטים).
בנוסף, באמצעות דח"ה - אם S_1, S_2 הם הנונטרמינלים ההתחלתיים של L_1, L_2 בהתאמה, הנונטרמינל של $L_1 \circ L_2$ יהיה S, ונוסיף את החוק $S \to S_1 S_2$.

(ב) <u>הטענה נכונה</u> - ל-L קיים א"מ M. נבנה א"מ מחסנית N שיכיל שני עותקים של M - אחד זהה ל-M (נסמן ב-A), והשני בו כל מעברי $\sigma, \gamma_1 \to \gamma_2$ יוחלפו ב-$\varepsilon, \gamma_1 \to \gamma_2$ (נסמן ב-B). בנוסף, מכל מצב ב-Q_A נעביר מעבר $\varepsilon, \varepsilon \to \varepsilon$ למצב המקביל ב-Q_B.

הוכחה. נוכיח כי $L(N) = \text{Prefix}(L)$.

תהי $w \in \text{Prefix}(L)$. אזי, קיים u כך ש-$w \circ u \in L$ וקיים לו מסלול מקבל ב-A, אותו נסמן ב-$(q_0, s_0), \dots, (q_n, s_n)$.

כעת, נבחר k כך שהתו האחרון של w נקרא במעבר $(q_{k-1}, s_{k-1}) \to (q_k, s_k)$.

נשים לב כי המסלול $(q_0^A, s_0^A), \dots, (q_{k-1}^A, s_{k-1}^A), (q_k^B, s_k^B), \dots, (q_n^B, s_n^B)$ הוא מסלול חישוב של w ב-N.

מאחר ו-q_n הוא מצב מקבל, w מסתיימת במצב מקבל ולכן $w \in L(N)$.

תהי $w \in L(N)$. נניח כי w מתקבלת עם מסלול חישוב $(q_0^A, s_0^A), \dots, (q_n^B, s_n^B)$. אזי, קיים k עבורו $(q_k^A, s_k^A) \to (q_{k+1}^B, s_{k+1}^B)$, ושני המצבים הם עותקים של אותו המצב ב-M.

לפי הבנייה, לאחר המעבר הנ"ל לא נשארים תווים לקרוא. נשים לב כי מסלול החישוב ב-M: $(q_0, s_0), \dots, (q_k, s_k), (q_{k+2}, s_{k+2}), \dots, (q_n, s_n)$ מתקבל - הורדנו מעבר ε. החלק $(q_0, s_0), \dots, (q_k, s_k)$ מתאים ל-w, ולכן קיימת u שמקיימת $wu \in L$, כנדרש. $\square$

(ג) <u>הטענה אינה נכונה</u> - עבור $L = \{w \mid \#_0(w) = \#_1(w)\}$. השפה ח"ה ואינה רגולרית, ו-$\text{Prefix}(L) = \{0,1\}^*$, ולכן רגולרית.

8. נפתור.

(א) $L = \{a^n b^m c^{n+m} \mid n, m \geq 0\}$.

$$S \to aSc \mid B$$
$$B \to bBc \mid \varepsilon$$

הוכחה. נוכיח כי הדח"ה גוזר את השפה. $\square$

- בהינתן מילה $a^n b^m c^{n+m}$, נפעיל את הכלל $S \to aSc$ n פעמים.
  - לאחר מכן, נפעיל $S \to B$ ואז $B \to bBc$ m פעמים. לבסוף, נשתמש ב-$B \to \varepsilon$ ונקבל את המילה.
- ניתן לראות באינדוקציה כי $\#_a + \#_b = \#_c$ אחרי כל צעדי גזירה.
  - ברור כי כל ה-a-ים מופיעים לפני כל ה-b-ים, ושני אלו לפני כל ה-c-ים.

(ב) $L = \left\{a^n b^m c^{(n+m)/2} \mid n + m \equiv 0 \mod 2\right\}$.

$$A \to aaAc \mid abBc \mid B$$
$$B \to bbBc \mid \varepsilon$$

(ג) $L = \left\{a^n b^m c^k \mid k \in \{n, m\}\right\}$.

$$S \to A \mid A'$$
$$A \to aAc \mid B$$
$$B \to bB \mid \varepsilon$$
$$A' \to aA' \mid B'$$
$$B' \to bB'c \mid \varepsilon$$

(ד) שפת המילים המאוזנות (אפשר גם: $S \to S0S1S \mid S1S0S \mid \varepsilon$).

$$S \to SS \mid 0S1 \mid 1S0 \mid \varepsilon$$

2.7 שפות ח"ה

1. נפתור.

(א) $L = \{a^n b^n c^i \mid i \leq n\}$.

הוכחה. נניח בשלילה כי L רגולרית.

נשתמש בלמת הניפוח, ונסתכל על המילה $a^{n_0} b^{n_0} c^{n_0}$ שאורכה $3n_0 \geq n_0$.

אזי, קיימים u, v, x, y, z כך ש-$w = uvxyz$ ומתקיימים תנאי הלמה.

ננפח עם $k = 0$. vxy בטוח מכיל a-ים ו-b-ים בלבד! לא ניתן שיכיל את שלושת התווים, ואם מוסיפים a-ים חייב להוסיף b-ים.

על כן, עבור $k = 0$ מספר ה-a-ים וה-b-ים יקטן, וכך $\#_a < n_0 = \#_c$ והמילה לא בשפה, בסתירה ללמת הניפוח.

בסך הכל, קיבלנו כי L אינה ח"ה. $\square$

(ב) $L = \{0^t \mid t \text{ ראשוני}\}$.

הוכחה. נניח בשלילה כי L רגולרית.

נשתמש בלמת הניפוח, ונסתכל על המילה 0^p עבור $p > n_0$.

אזי, קיימים u, v, x, y, z כך ש-$w = uvxyz$ ומתקיימים תנאי הלמה.

ננפח עם $k = p + 1$. נקבל כי אורך המילה הוא $p + p \times |vy|$, שאינו ראשוני - כלומר $uv^{p+1}xy^{p+1}z \notin L$ בסתירה ללמת הניפוח, וכך L אינה ח"ה. $\square$

(ג) $L = \left\{0^i 1^{i^2} \mid i \geq 0\right\}$.

הוכחה. נניח בשלילה כי L רגולרית.

נשתמש בלמת הניפוח, ונסתכל על המילה $0^{n_0} 1^{n_0^2}$ שאורכה $n_0 + n_0^2 \geq n_0$.

אזי, קיימים u, v, x, y, z כך ש-$w = uvxyz$ ומתקיימים תנאי הלמה.

ננפח עם $k = 2$. יתווספו n_0 תווים לכל היותר, וכך מספר ה-1-ים לא יכול להיות הריבוע השלם הבא.

כך, המילה לא בשפה, בסתירה ללמת הניפוח - ו-L אינה ח"ה. $\square$

(ד) $L = \left\{s \circ s^R \circ s \mid s \in \{0,1\}^*\right\}$.

הוכחה. נניח בשלילה כי L רגולרית.

נשתמש בלמת הניפוח, ונסתכל על המילה $0^{n_0} 1 1 0^{n_0} 0^{n_0} 1$ שאורכה $3n_0 + 3 \geq n_0$.

אזי, קיימים u, v, x, y, z כך ש-$w = uvxyz$ ומתקיימים תנאי הלמה.

ננפח עם $k = 2$. vxy מכיל אפסים מה-s השמאלי ומ-s^R, או מ-s^R ומ-s.

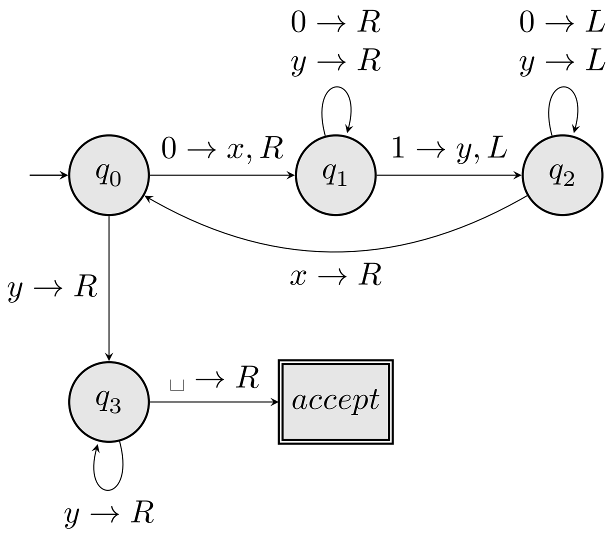
כך, בניפוח מספר ה-0-ים בחלק אחד של s יגדל ובשני לא.

על כן, עבור $k = 2$ המילה לא בשפה, בסתירה ללמת הניפוח.

בסך הכל, קיבלנו כי L אינה ח"ה. $\square$

2.8 מכונת טיורינג

1. $L=\{0^n1^n \mid n\geq 1\}$.



2. $L=\{w \mid \#_0(w)=\#_1(w)\}$. אלגוריתם:

(א) סמן את התו הראשון ב-$\$$.

i. אם הוא 0, צעד ימינה עד שנראה 1, סמן ב-x.

ii. אם הוא 1, צעד ימינה עד שנראה 0, סמן ב-x.

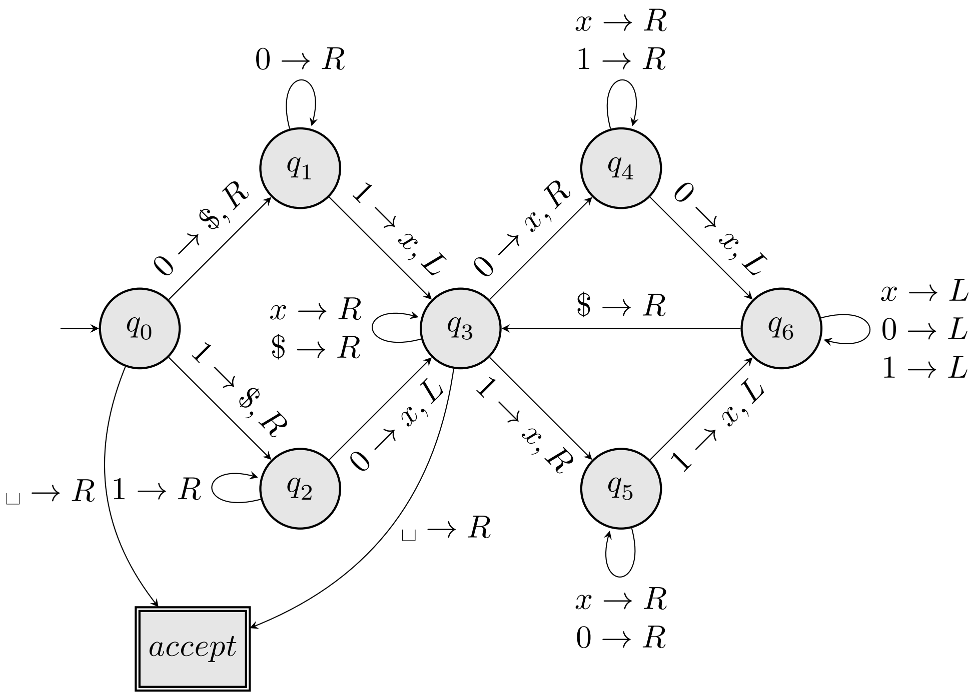
(ב) חזור עד $\$$, ואז לך ימינה עד שתראה תו שאינו x, אם הוא $0/1$ סמן אותו ב-x.

i. אם ראית 0, המשך ימינה עד 1, סמנו ב-x וחזור ל-(ב).

ii. אם ראית 1, המשך ימינה עד 0, סמנו ב-x וחזור ל-(ב).

iii. אם ראית רווח, עבור למצב מקבל.

הערה. ניתן לפתור בקלות באמצעות מ"ט 2-סרטית.



2.9 רדוקציות

1. הוכח כי אם $L, \overline{L}$ ניתנות לקבלה, אז שתיהן כריעות.

הוכחה. נבנה בה"כ מכונה שמכריעה את L.

בהינתן M, M' מ"ט שמקבלות את $L, \overline{L}$ בהתאמה, תהי M'' מ"ט שמכריעה את L:

הרץ בהרצה מבוקרת את M, M' במקביל על הקלט

ענה כן אם L קיבלה, ולא אם $\overline{L}$ קיבלה (אחת מהן בוודאות קיבלה את הקלט). $\square$

2. תהי $L = \{\langle M\rangle \mid \exists w : w, w^R \in L(M)\}$.

(א) L ניתנת לקבלה. נבנה מ"ט M' שמקבלת את L:

i. הרץ בהרצה מבוקרת (בדומה ל-$NON{-}EMPTY$) את הקלט M על כל המילים $w \in \Sigma^*$.

ii. אם בשלב מסוים נמצא w, w^R ש-M קיבלה, קבל.

(ב) L אינה כריעה. נוכיח ע"י רדוקציה מ-$ACCEPT$.

$$R(\langle M, w\rangle) = M'$$

- כאשר $M'(x)$ מוגדרת באופן הבא:
  - באופן בלתי תלוי בקלט, הרץ את M על w ועשה כמוה.
- זו רדוקצית מיפוי מ-$ACCEPT$ ל-L.
  - מילה תתקבל אמ"מ היא ב-$ACCEPT$.
- מכאן, L כריעה אמ"מ $ACCEPT$ כריעה ולכן L לא כריעה.

3. תהי $L = \{\langle M \rangle \mid$ C לא מקבלת אף תכנית $M\}$.

- L לא ניתנת לקבלה (וכך גם לא כריעה) - נראה ע"י רדוקציה מ-$NOT-ACCEPT$.

$$R\left(\langle M,w\rangle\right)=M'$$

- $M'(x)$ מוגדרת ע"י: הרץ את M על w ופעל כמוה.
  - אם דבר לא מתקבל, גם אף תכנית C לא תתקבל.
  - אם כל דבר מתקבל, קיימת תכנית C שתתקבל.
- מכאן, זו רדוקציית מיפוי וכך L לא ניתנת לקבלה ($NOT-ACCEPT$ לא ניתנת לקבלה).

4. תהי $EMPTY_{CFG} = \{G \mid L(G) = \emptyset$ הוא דח"ה שמקיים $G\}$.

השפה כריעה. אלגוריתם:

- יהיו שתי רשימות l_1, l_2.
  - l_1 תאותחל עם כל הנונטרמינלים, ואת l_2 לכל הטרמינלים.
- העבר מ-l_1 ל-l_2 את כל הנונטרמינלים A שקיים עבורם כלל גזירה מהצורה Aלסדרת איברים ב-l_2.
- נעצור כשאף איבר מ-l_1 לא עבר ל-l_2.
  - אם S ב-l_1 השפה ריקה.
  - אחרת, $L(G) \neq \emptyset$.

5. תחילה, קיימים קלטים שעליהם T לא תעצור לעולם, וניתן להראות כך שלא קיימת T כזאת.

הוכחה. נניח בשלילה כי קיימת מ"ט כזו T. אזי, $EQUAL$ כריעה:

בהינתן $EQUAL\left(\langle M_1\rangle, \langle M_2\rangle\right)$, נבדוק האם $\langle M_2\rangle \in T\left(\langle M_1\rangle\right)$. □

6. נפתור!

(א) $L_1 = \left\{x\#x^R z x^R \mid z \in \{0,1\}^*\right\}$ היא כריעה.

i. נניח בשלילה ש-L_1 ח"ה, אזי השפה L' שמוגדרת להיות החיתוך של L_1 והשפה השקולה ל-10*1#10*110*1.

ii. מסגירות של שפות רגולריות עם חיתוך, גם L' ח"ה.

$$L' = \{10^n 1\#10^n 110^n 1 \mid n \geq 0\}$$

iii. נשתמש בלמת הניפוח לשפות ח"ה:

א'. נפרק מילה $w = 10^{n_0}1\#10^{n_0}110^{n_0}1$ ל-$w = uvxyz$ ע"פ הלמה.

ב'. ב-u, y יש 0-ים בלבד.

ג'. מאחר ו-$|vxy| \leq n_0$, לא ייתכן ש-v, y מכילים 0-ים משלושת הרצפים.

ד'. כך, המילה $uv^2xy^2z \notin L'$ - לא שלושת רצפי ה-0-ים יתנפחו והמילה לא תהיה תקינה.

iv. מכאן, L' ח"ה. הגענו לסתירה וכך L_1 אינה ח"ה.

v. ברור שהשפה כריעה - ניתן לתאר בקלות אלגוריתם שמכריע אותה.

(ב) $L_2 = \{a^n b^m \mid 2n \leq 3m \vee 2m \leq 3n\}$ היא רגולרית.

$$2n > 3m \Longrightarrow \frac{4n}{3} > 2m \Longrightarrow 3n > 2m$$

מכאן, $2n \leq 3m \vee 2m \leq 3n \equiv T$, וכך השפה L_2 היא פשוט a^*b^* - רגולרית.

(ג) $L_3 = \{\langle M, q\rangle \mid M$ מ"ט ו-q מיותר ב-$M\}$ אינה קבילה.
ניתן להראות אלגוריתם שמקבל את $\overline{L_3}$ ואת אי כריעותה של L_3, וכך להוכיח ש-L_3 אינה קבילה.

הוכחה. נוכיח כי L_3 לא קבילה.
נראה כי $NOT-ACCEPT \leq_m L_3$ ע"י רדוקציית מיפוי R:

$$R(\langle M, w\rangle) = \langle M', q\rangle$$

- כאשר $M'(x)$:
  - הרץ את M על w.
  - אם קיבלה עבור ל-q וקבל, אחרת דחה.
- קל לראות שהרדוקציה עובדת, וכך $NOT-ACCEPT \leq_m L_3$.

מכאן, מאחר ו-$NOT-ACCEPT$ אינה קבילה גם L_3 אינה קבילה. □

(ד) $L_4 = \{\langle M\rangle \mid \langle M\rangle \in L(M)\}$ קבילה.

הוכחה. נוכיח כי L_4 קבילה.
נראה כי $ACCEPT \leq_m L_4$:

$$R(\langle M, w\rangle) = \langle M'\rangle$$

כך ש-M' מריצה את M על w ומחזירה כמוה לכל קלט.

$$\langle M, w\rangle \in ACCEPT \Longrightarrow L(M') = \Sigma^* \Longrightarrow \langle M'\rangle \in L(M') \Longrightarrow \langle M'\rangle \in L_4$$

$$\langle M, w\rangle \notin ACCEPT \implies L(M') = \emptyset \implies \langle M'\rangle \notin L(M') \implies \langle M'\rangle \notin L_4$$

מכאן, L_4 אינה כריעה. בנוסף, נוכיח כי $L_4 \leq_m ACCEPT$:

$$R(\langle M\rangle) = \langle M, \langle M\rangle\rangle$$

$$\langle M\rangle \in L_4 \implies \langle M\rangle \in L(M) \implies \langle M, \langle M\rangle\rangle \in ACCEPT$$

$$\langle M\rangle \notin L_4 \implies \langle M\rangle \notin L(M) \implies \langle M, \langle M\rangle\rangle \notin ACCEPT$$

כך L_4 קבילה. כלומר - L_4 קבילה ואינה כריעה. □

2.10 P לעומת NP

1. נוכיח כי אם $L_1, L_2 \in NP$ אז $L_1 \cup L_2 \in NP$.

הוכחה. מאחר ו-$L_1, L_2 \in NP$ קיימים מוודאים פולינומיים V_1, V_2 עבורן בהתאמה. נבנה מוודא פולינומי $V(x,y)$:

- הרץ את $V_1(x,y)$, וקבל אם קיבל.
- הרץ את $V_2(x,y)$, וקבל אם קיבל.
- דחה.

□

2. נגדיר $IND-SET = \{\langle G, k\rangle \mid k \text{ קבוצה ב"ת בגודל } G\text{-ב}\}$. הוכח:

(א) רדוקציה פולינומית שתמיר מ-$\langle G, k\rangle$ ל-$\langle \overline{G}, k\rangle$.

(ב) קל לבנות מוודא פולינומי שבודק בהינתן קבוצה האם היא ב"ת ובגודל k.

3. נגדיר $DHPATH = \{\langle G\rangle \mid G \text{ מכוון ויש בו מסלול המילטוני}\}$. הראו כי:

(א) בהינתן מסלול, קל לבדוק בזמן פולינומי שהוא מסלול המילטוני תקין (נעבור על המסלול, כל קודקוד צריך להופיע בדיוק פעם אחת).

(ב) $HAM-PATH \leq_p DHPATH$ נגדיר רדוקציה $R(\langle G, s, t\rangle) = \langle G'\rangle$, בו דרגת היציאה של t ודרגת הכניסה של s הן 0.

i. אם קיים ב-G מסלול המילטוני $s \rightsquigarrow t$, אנחנו לא צריכים את הקשתות שנכנסות ל-s או יוצאות מ-t, וכך גם ב-G' מסלול המילטוני מ-s ל-t.

ii. אם ב-G' מסלול המילטוני כלשהו הוא בוודאות $s \rightsquigarrow t$: לא ייתכן ש-s אינו הראשון וש-t אינו האחרון.

(ג) $DHPATH \leq_p HAM-PATH$. נגדיר רדוקציה $R(\langle G\rangle) = \langle G', s, t\rangle$, שמוסיפה צמתים s ו-t שיחוברו לכל הקודקודים.

i. אם קיים ב-G מסלול המילטוני $u \rightsquigarrow v$, קיים מסלול המילטוני $s \rightsquigarrow t$ ב-G': $s \to u \rightsquigarrow v \to t$.

ii. אם ב-G' מסלול המילטוני $s \to u \rightsquigarrow v \to t$. נסתכל על הגרף G, שלא מכיל את s, t: המסלול $u \rightsquigarrow v$ הוא המילטוני.

4. וואלה נוכיח.

(א) נחזיר נוסחא $4-CNF$, בה כל פסוקית מהקלט משורשרת עם שכפול של המשתנה הראשון:

$$l_1 \vee l_2 \vee l_3 \to l_1 \vee l_2 \vee l_3 \vee l_1$$

הנוסחאות שקולות וכך הרדוקציה עובדת - הנוסחא המקורית ספיקה אמ"מ החדשה ספיקה.

(ב) נפלוט נוסחא $3-CNF$, בה עבור הפסוקית ה-i, $l_1 \vee l_2 \vee l_3 \vee l_4$, נגדיר משתנה α_i ואת הפסוקיות:

$$(l_1 \vee l_2 \vee \alpha_i) \wedge (l_3 \vee l_4 \vee \neg\alpha_i)$$

בהינתן השמה מספקת ϕ ל-$SAT-4$, נגדיר השמה ϕ' ל-$SAT-3$ באופן הבא:

$$\alpha_i \equiv T \Longleftrightarrow l_1 \equiv l_2 \equiv F$$

i. אם ϕ ספיקה אזי הפסוקית ה-i היא T.
   א'. אם $l_1 \equiv l_2 \equiv F$ אז $\alpha_i \equiv T$ וכך הפסוקית $l_1 \vee l_2 \vee \alpha_i \equiv T$ וגם $l_3 \vee l_4 \equiv T \Longrightarrow l_3 \vee l_4 \vee \neg\alpha_i \equiv T$.
   ב'. אחרת, $l_1 \vee l_2 \equiv T \Longrightarrow l_1 \vee l_2 \vee \alpha_i \equiv T$. בנוסף, $\alpha_i \equiv F$ וכך $l_3 \vee l_4 \vee \neg\alpha_i \equiv T$.

ii. אחרת, קיימת פסוקית i עבורה $l_1 \vee l_2 \vee l_3 \vee l_4 \equiv F$.
   א'. מכאן, $l_1 \vee l_2 \equiv F$ או $l_3 \vee l_4 \equiv F$.
   ב'. כך, ללא תלות ב-α_i אחת מתתי הפסוקיות יהיו F, וכך גם ϕ' לא ספיקה.