הרצאה 1

1 מבוא – מודלים חישוביים

במהלך ההרצאה יהיו הרבה מושגים חדשים ולא מובנים, שיוסברו בהמשך.

באופן לא פורמלי: מודל חישובי הוא מערכת המורכבת מקריאה של קלט, קבוצת מצבי בקרה, והתקדמות כלשהי.
לדוגמה – מערכת עקיבה סינכרונית קוראת קלט, מכילה מצבי זיכרון, ומתקדמת ביניהם בהתאם לקלט.
"אוטומט" הוא מודל חישובי שבנוי על אותו עיקרון.

המודל החישובי נותן לחקור "שפות", וזה מאפשר לנו לחקור מחשבים. מחשב הוא מודל חישובי מורכב. בקורס "חישוביות" נגדיר מערכת פשוטה יותר ששקולה בכוחה למחשב. בעזרת מערכות שקולות נוכל לחקור את גבולות המחשב. הקורס "אוטומטים" מהווה מבוא לחישוביות, ובמהלכו נגדיר ארבעה מודלים בעלי מורכבות עולה, עד המודל החמישי שיוגדר בחישוביות – מכונת טיורינג.

אוטומטים בקורס:
1) אוטומט סופי דטרמיניסטי.
2) אוטומט סופי לא דטרמיניסטי.
3) אוטומט סופי לא דטרמיניסטי, עם מסעי ϵ.
4) אוטומט מחסנית, עם שני מודי קבלה. (בסוף הקורס).

כוח סביר: מודל בעל כוח אינסופי, יוכל לעשות הכול. המודלים שיוגדרו בקורס יהיו תחת הגבלות מסוימות. יש קטגוריות של מודלים, לפי ההגבלות של כל אחד. הגבלות רלוונטיות:
1. יכולת לרשום משהו לזיכרון.
2. זיכרון סופי או אינסופי.
3. ביצוע כמה צעדים אפשריים על אותו קלט.
4. התקדמות בלי קלט.

1.1 הגדרות

א"ב הקלט: Σ – קבוצה סופית לא ריקה של תווים. $\Sigma_1 = \{a, b, c\}, \quad \Sigma_2 = \{0,1\}$.

מילה: סדרה סופית של תווים מתוך הא"ב הנתון. $w_1 = \{acccba\}, w_2 = \{1010011001\}$. מילה ריקה מסומנת ϵ.

אורך של מילה הוא מספר התווים. $|\epsilon| = 0, |abacb| = 5$.

שפה: קבוצה של מילים. $L_1 = \{w_1, w_2\}, L_2 = \{100, 011, 1\}$. שפה ריקה מסומנת ϕ. [אין מילים בפי :( ]

יכולה להיות סופית או אינסופית: $L = \{w \in \{a,b\}^* : w \ starts \ with \ a\}$.

> שימו לב: $\phi \neq \epsilon$. קודם כל, הם לא מאותו סוג. דבר שני, המילה הריקה היא עדיין מילה.
> שפה יכולה להכיל את המילה הריקה, אבל לא בהכרח. (לדוגמה, השפה L מהדוגמה הקודמת).

1.2 פעולות על מילים

שרשור: תהיינה w_1, w_2 מילים מעל א"ב Σ. אזי $w_1 \cdot w_2$ היא "הדבקה" של w_2 בסוף w_1.

באופן כללי, $w_1 \cdot w_2 = w_2 \cdot w_1$ אמ"מ $w_1 = w_2$. (שרשור הוא לא חילופי).

גם אם $w_1, w_2 \in L$, לא בהכרח $w_1 \cdot w_2 \in L$. לדוגמה אם L היא מילים באורך מסוים.

שרשור עם המילה הריקה לא משפיע.

שרשור הוא פעולה קיבוצית (קומוטטיבית): $w_1 \cdot (w_2 \cdot w_3) = (w_1 \cdot w_2) \cdot w_3 = w_1 \cdot w_2 \cdot w_3$.

חזקה: עבור מילה w, ומספר טבעי n. אז: $w^n = \underbrace{w \cdots w}_{n\ times}$.

אפשר גם להגדיר רקורסיבית:

$$w^0 = \epsilon, \qquad \forall i \geq 1, w^i = w \cdot w^{i-1}$$

היפוך: תהי $w = a_1 a_2 \dots a_k$ מחרוזת המורכבת מ-k תווים (לאו דווקא שונים). אזי $w^R = a_k a_{k-1} \dots a_2 a_1$.

1.3 פעולות על שפות

מכיוון ששפה היא קבוצה של מילים, ניתן להפעיל עליה את כל הפעולות שקיימות על קבוצות. איחוד, חיתוך, הפרש, הפרש סימטרי, משלים. בנוסף יש פעולות חדשות: שרשור, חזקה, איטרציה.

תהיינה L, L_1, L_2 שפות. אזי:

איחוד: $L_1 \cup L_2$ מכיל את כל המילים שמופיעות בלפחות אחת השפות.

חיתוך: $L_1 \cap L_2$ מכיל את כל המילים שמופיעות בשתי הקבוצות.

הפרש: $L_1 \setminus L_2 = \{w : w \in L_1 \wedge w \notin L_2\}$, כל המילים שמופיעות ב-$L_1$ ולא ב-L_2 .

הפרש סימטרי: $L_1 \Delta L_2 = (L_1 \setminus L_2) \cup (L_2 \setminus L_1) = \{\{w : w \in L_1 \wedge w \notin L_2\} \cup \{w : w \notin L_1 \wedge w \in L_2\}\}$.

כל המילים שמופיעות רק באחת השפות. (XOR)

שרשור: $L_1 \cdot L_2 = \{uv : u \in L_1, v \in L_2\}$ (לכל בחירה של מילה מכל קבוצה, שרשור של שתי המילים).
מתקיים: $|L_1 \cdot L_2| \leq |L_1| \cdot |L_2|$.

אם משרשרים עם שפה שקיים בה ϵ, יש לזה השפעה: כל המילים מהשפה השנייה יופיעו כמו שהן (כי הן משורשרות עם אפסילון).

$$L_1 = \{ab, c\}, L_2 = \{\epsilon, d\}, \qquad L_1 \cdot L_2 = \{ab\epsilon, abd, c\epsilon, cd\} = \{ab, abd, c, cd\}$$

שרשור עם שפה ריקה: $L \cdot \phi = \phi = \phi \cdot L$, כי לפי הגדרה: $L \cdot \phi = \{uv : u \in L, v \in \phi\}$, אבל אין $v \in \phi$.

חזקה: $L^n = \underbrace{L \cdots L}_{n\ times}$.

נשים לב להבדל בין הפעולה על מילה לפעולה על שפה: בחזקה על מילה, מבצעים את השרשור עם אותה מילה כל פעם. בחזקה של קבוצה, אפשר כל פעם לבחור מילה אחרת. אז, לדוגמה, אם יש שפה עם 4 מילים (נניח שאין מילים זהות) – בקבוצה L^n יהיו 4^n מילים אפשריות. גודל הקבוצה שמקבלים חסום ב m^n (כאשר $|L| = m$).

איטרציה: $L^* = \bigcup_{i=1}^{\infty} L^i = L^0 \cup L^1 \cup L^2 \cup \dots$. לדוגמה, אם ניקח $L = \Sigma$ (שפה שהיא רק האותיות של הא"ב). אז:
$L^* = \Sigma^0 \cup \Sigma^1 \cup \dots = \Sigma^*$, כל המילים בכל אורך שאפשר לייצר מהא"ב הזה. (תזכורת – מילה מוגדרת בתור רצף סופי בלבד).

אם $\Sigma = \{0,1\}$, אז Σ^* היא שפת כל המילים הבנאריות. באופן כללי: $\Sigma^0 = \epsilon$ אז המילה הריקה מוכלת באיטרציה של כל שפה.

טענה: $|\Sigma^*| \approx \aleph_0$. (אינסופית בת מניה).

הוכחה: נסדר את הא"ב בסדר כלשהו, ונרכיב מילים בצורה לקסיקוגרפית. המילה הראשונה היא המילה הריקה, ואז כל התווים עצמם, ואז כל המילים של שני תווים, וכו'. מכיוון שאנחנו מסדרים בסדר לקסיקוגרפי, אפשר לבצע התאמה חח"ע ועל לטבעיים.

(אם כל תו מייצג ספרה, זה בעצם בסיס ספירה. בכל בסיס ספירה יש לכל מספר טבעי בדיוק ייצוג אחד).

משלים: $\overline{L} = \Sigma^* \setminus L$. כל המילים האפשריות תחת הא"ב, שלא נמצאות בקבוצה. אבחנה: $L_1 \setminus L_2 = L_1 \cap \overline{L_2}$.
כל מה שנמצא בראשון ולא בשני, זה כמו כל מה שנמצא בראשון וגם נמצא במשלים של השני.

תרגיל ממבחן: צ"ל: $L_1 \subseteq L_1 \cdot L_2 \Leftrightarrow \epsilon \in L_2$. (נתון $L_1 \neq \phi$. אם זה לא נתון, יש הפרכה).

כיוון ראשון $\Rightarrow$ מתקיים לפי הגדרת שרשור: $L_1 = L_1 \cdot \{\epsilon\} \subseteq L_1 \cdot L_2$

כיוון שני $\Leftarrow$: נניח ש $\epsilon \notin L_2$, ונראה ש $L_1 \nsubseteq L_1 \cdot L_2$. יהיו u, v המילים הקצרות ביותר ב- L_1, L_2 בהתאמה. אזי המילה הקצרה ביותר ב- $L_1 \cdot L_2$ היא uv. מכיוון ש $v \neq \epsilon$, מתקיים $|v| \geq 1$. לכן $|u| \leq |uv|$, כלומר $u \notin L_1 \cdot L_2$.

2 אוטומט סופי

אוטומט סופי הוא מודל חישובי שבו יש אוסף סופי של מצבים עם כללי מעבר ממצב אחד לשני. שימושים:

1. פיתוח קומפיילרים,
2. תכנון מערכות ספרתיות,
3. אימות תוכנה וחומרה.

הגדרה לא פורמלית: מודל מתמטי מופשט שמתאר מערכת שמגיבה לקלטים סופיים. המודל מורכב מ:

1. סרט,
2. ראש קורא,
3. בקרה מרכזית,
4. פלט.

תכונות חשובות:

1. בקרה בגודל סופי.
2. החישוב תמיד מתחיל באותו מצב.
3. קלט באורך סופי, אך לא ידוע מראש ולא חסום.
4. מעבר יחיד על הקלט.

אופן פעולה:

1. בכל שלב האוטומט נמצא באחד המצבים.
2. בקריאת אות קלט, הראש מוזז אות אחת ימינה והאוטומט עובר למצב חדש.
3. זהות המצב נקבעת לפי המצב הקודם ואות הקלט שנקראה.
4. תוצאת החישוב מוגדרת ע"י המצב שבו נמצא האוטומט בסיום קריאת הקלט.

ההצגה הפשוטה ביותר של אוטומט היא גרף: צומת = מצב, קשת = מעבר בין מצבים. תיוג על הקשתות מציין סיבה למעבר.

באוטומט סופי שני סוגי מצבים: מצבים **מקבלים** ומצבים **לא מקבלים**.

תוצאת החישוב מוגדרת ע"י סוג המצב שאליו הגיע האוטומט בסיום קריאת הקלט:

- נאמר שהאוטומט **מקבל** קלט w אם הוא מסיים את קריאת w במצב מקבל.
- אחרת, נאמר שהוא **דוחה** את w.

תגובתו של אוטומט לקלט הינה פונקציה של המצב הנוכחי ושל האות.

הפונקציה הזו קובעת מצב יחיד שאליו האוטומט הדטרמיניסטי יעבור.

3 אוטומט סופי דטרמיניסטי

3.1 הגדרה

לאוטומט סופי דטרמיניסטי יש 5 מרכיבים:

1. א"ב – כל אותיות הקלט האפשריות עבור האוטומט. מספר האותיות בא"ב חייב להיות סופי וגדול ממש מ-0.
2. מצבים – כל המצבים שבהם האוטומט יכול להימצא. מספר המצבים חייב להיות סופי וגדול ממש מ-0.
3. מצב התחלתי – המצב שממנו מתחיל האוטומט את מסלול החישוב על כל מילת קלט.
4. קבוצת מצבים מקבלים – תת-קבוצה של קבוצת המצבים, המכילה 0 מצבים או יותר.
5. פונקציית מעברים – לכל זוג של מצב ואות, פונקציה זו מתאימה מצב (אחד ויחיד) שאליו עובר האוטומט כאשר במצב זה נקראת אות זו.

פורמלית: אוטומט סופי דטרמיניסטי A נתון על ידי חמישייה סדורה:

$$A = (\Sigma, Q, q_0, F, \delta)$$

1. Σ – א"ב קלט.
2. Q – קבוצה סופית לא ריקה של מצבים.
3. $q_0 \in Q$ – מצב התחלתי.
4. $F \subseteq Q$ – קבוצת מצבים מקבלים.
5. $\delta : Q \times \Sigma \to Q$ – פונקציית מעברים.

פונקציית המעבר: מקבלת שני ארגומנטים: מצב ואות.
$\delta(q, a)$ – המצב שאליו האוטומט עובר כשהוא במצב q וקרא את האות a מהקלט.

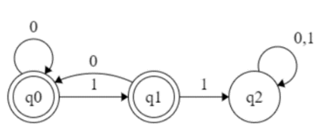
הצגה על ידי גרף:

- צומת = מצב.
- קשתות מייצגות את פונקציית המעבר. קשת pq מתויגת על ידי כל האותיות שיש להן מעבר מ- p ל- q.
- קשת מתויגת "התחלה" מצביעה למצב ההתחלתי.
- מצבים מקבלים מסומנים על ידי עיגול כפול. (או כוכבית, אם זה טבלה).
- כמו גרף, לפעמים נוח לייצג אותו בטבלה.

דוגמה 1: אוטומט המקבל את כל המחרוזות מעל $\{0,1\}$ שאין בהן 11.

מצבים מקבלים מסומנים ב- *, מצב התחלתי בחץ.
בדוגמה הזו, q_2 הוא **בור.**

Q	0	1
$\rightarrow. * q_0$	q_0	q_1
$* q_1$	q_0	q_2
q_2	q_2	q_2



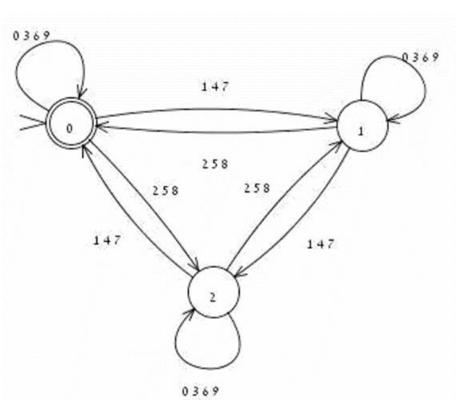
דוגמה 2: אוטומט המקבל את כל המחרוזות המסתיימות ב- "abc".

Q	a	b	c	$else$
$\rightarrow.\ q_0$	q_1	q_0	q_0	q_0
q_1	q_1	q_2	q_0	q_0
q_2	q_0	q_0	q_3	q_0
$*q_3$	q_0	q_0	q_0	q_0

דוגמה 3:

האם מספר

מתחלק ב-3:



3.2 הרחבת פונקציית המעברים למחרוזות

עד כה קראנו בעזרת δ אות בודדת כל פעם. נרצה להרחיב את הגדרת δ לקריאת מחרוזות. אינטואיציה: $\hat{\delta}$ מורחבת מחושבת ממצב q ומחרוזות $a_1 a_2 \dots a_n$ ע"י כך שנעקוב אחרי מסלול המתחיל ב-q ועובר דרך קשתות עם תיוגים $a_1, a_2, \dots, a_n$.

נגדיר באופן רקורסיבי על אורך המחרוזת:

בסיס: $\hat{\delta}(q, \epsilon) = q$. $\hat{\delta}(q, a) = \delta(q, a)$.

לכל wa (כאשר w היא מחרוזת, a היא אות): $\hat{\delta}(q, wa) = \delta\big(\hat{\delta}(q, w), a\big)$.

דוגמה:

$$\hat{\delta}(q_0, 011) = \delta\big(\hat{\delta}(q_0, 01), 1\big) = \delta\Big(\delta\big(\hat{\delta}(q_0, 0), 1\big), 1\Big)$$

$$= \delta\left(\underbrace{\delta\Big(\underbrace{\delta(q_0, 0)}_{q_0}, 1\Big)}_{q_1}, 1\right) = \delta(\delta(q_0, 1), 1)$$

$$= \delta(q_1, 1) = q_2$$

Q	0	1
$\rightarrow. * q_0$	q_0	q_1
$* q_1$	q_0	q_2
q_2	q_2	q_2

באופן כללי מתקיים:

$$\hat{\delta}(q, w_1 w_2) = \hat{\delta}\big(\hat{\delta}(q, w_1), w_2\big)$$

ההוכחה באינדוקציה על $|w_2|$.

3.3 שפה של אוטומט דטרמיניסטי

אם $A = (\Sigma, Q, q_0, F, \delta)$ הוא אוטומט, אזי $L(A)$ היא השפה שהאוטומט מקבל. כלומר, אוסף כל המחרוזות שהאוטומט מקבל – כלומר, אוסף כל המחרוזות שמהוות מסלול מהמצב ההתחלתי למצב מקבל.

פורמלית: $L(A)$ היא אוסף כל המחרוזות w כך ש: $\hat{\delta}(q_0, w)$ שייך ל- F.

$$L(A) = \{w \in \Sigma^* : \hat{\delta}(q_0, w) \in F\}$$

לכל מצב $q \in Q$, **שפת המצב** $L_A(q)$ היא השפה הבאה:

$$L_A(q) = \{w \in \Sigma^* : \hat{\delta}(q_0, w) = q\}$$

כל המחרוזות שאם נתחיל מ- q_0 ונקרא את המחרוזת, נגיע למצב הזה.

מתקיים:

$$L(A) = \bigcup_{q \in F} L_A(q)$$

שפת האוטומט היא איחוד כל המצבים המקבלים.

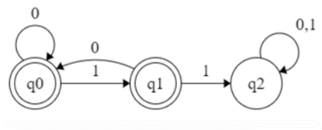
הרצאה 2

1 נכונות בנייה של אוטומט

הוכחה שהשפה של אוטומט היא השפה הנתונה:

בדוגמה מהרצאה 1 – אוטומט המקבל את כל המחרוזות הבינאריות שאין בהן 11:

Q	0	1
$* \, q_0 \rightarrow.$	q_0	q_1
$* \, q_1$	q_0	q_2
q_2	q_2	q_2



השפה שמקבל האוטומט נקראת $L(A)$.

השפה "המחרוזות הבינאריות שלא מכילות 11", נקרא לה L.

כדי להוכיח ש $L = L(A)$ (ניזכר ששפה היא קבוצה), נוכיח הכלה דו-כיוונית.

כיוון ראשון: כדי להוכיח ש $L(A) \subseteq L$, נוכיח שאם w מתקבל ע"י האוטומט אזי הוא לא מכיל שני ביטים 1 עוקבים.

ננסח טענה חזקה יותר, שמאפיינת לא רק את שפת האוטומט, אלא מצבים נוספים: "אפיון לשפת כל מצב מקבל". נאפיין את כל המצבים שחץ יחיד מוביל מהם למצב מקבל. זה בעצם המבנה של הרקורסיה – המצב הזה הוא ה"הנחה", והחץ האחרון הוא צעד האינדוקציה. זה יותר חזק מלאפיין את שפת האוטומט כי אנחנו נאפיין את כל אחד מהמצבים בפני עצמו.

הטענה:

- אם $\hat{\delta}(q_0, w) = q_0$, אז ב-$w$ אין 11, והיא מסתיימת ב-0, או $w = \epsilon$.
- אם $\hat{\delta}(q_0, w) = q_1$, אז ב-$w$ אין 11, והיא מסתיימת ב-1 בודד.

ההוכחה באינדוקציה: בסיס: $w = \epsilon$:

מתקיים: $\delta(q_0, \epsilon) \in F$ (מצב מקבל) ואכן $\epsilon \in L$ (כי $11 \notin \epsilon$).

נניח שהטענות החזקות מתקיימות עבור $|w| \leq n$, ונשקול w כך ש $|w| = n + 1$.

נסמן $w = x\sigma$ כאשר $|\sigma| = 1$, $|x| = n$, ונחלק לפי מקרים:

מקרה א: אם $\hat{\delta}(q_0, x\sigma) = q_0$: לפי בניית האוטומט, $\hat{\delta}(q_0, x) \in \{q_0, q_1\}$, כי אין עוד חץ שמגיע ל- q_0 בצעד בודד.

לכן, x אינה מכילה 11 ולכל היותר נגמרת ב-1 בודד. הצעד האחרון חייב להיות $\delta(q_0, 0) = \delta(q_1, 0) = q_0 \in F$ (כי אין דרך אחרת להגיע ל-q_0), ובסה"כ $x\sigma$ לא מכילה 11, ומסתיימת ב-0. והוכחנו את הטענה החזקה של שפת מצב זה.

מקרה ב: אם $\hat{\delta}(q_0, x\sigma) = q_1$: לפי בניית האוטומט, $\hat{\delta}(q_0, x) = q_0$ (כי רק מ-q_0 יש חץ ל-q_1).

לכן, x אינה מכילה 11 ולא נגמרת ב-1. הצעד האחרון חייב להיות $\delta(q_0, 1) = q_1 \in F$, ובסה"כ $x\sigma$ לא מכילה 11, ומסתיימת ב-1. והוכחנו את הטענה החזקה של שפת מצב זה.

בסה"כ, הוכחנו ש $L(A) \subseteq L$.

כיוון שני: כדי להוכיח ש $L \subseteq L(A)$, נוכיח בשלילה: נראה שאם w לא מתקבלת ע"י האוטומט, אזי w מכילה 11:

הדרך היחידה ש-w לא מתקבלת, היא אם האוטומט מגיע ל-q_2.

הדרך היחידה להגיע ל-q_2, $(\hat{\delta}(q_0, w) = q_2)$ היא אם $w = x1y$, כאשר x מסתיים ב-q_1, ו-y הוא הסיפא של w אחרי שהאוטומט מגיע ל-q_2 בפעם הראשונה. (פחות פורמלית – x מסתיים ב-q_1, ואז יש 1, ואז y יכול להיות כל דבר).

אם $\hat{\delta}(q_0, x) = q_1$, בהכרח $x = z1$ עבור z כלשהו.

לכן, $w = z11y$, כלומר w מכיל 11.

2 שפות רגולריות

נתעניין בשאלות הבאות:

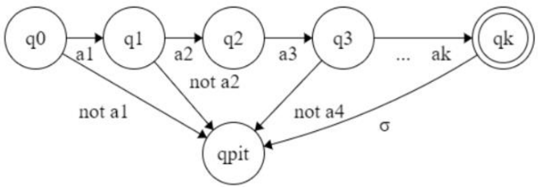
- האם קיימת שפה שאס"ד אינו מסוגל לזהות?
- אילו תכונות יש לקבוצת השפות שאותן יכול אס"ד לקבל?
- האם קיימים מודלים נוספים המקבלים את אותה קבוצת שפות?

הגדרה: שפה L היא **רגולרית** אם היא מתקבלת ע"י אס"ד.

דוגמאות לשפות רגולריות:

1. השפה הריקה ϕ.
2. השפה $\{\epsilon\}$.
3. לכל $a \in \Sigma$, השפה $\{a\}$.
4. לכל מילה $w \in \Sigma^*$, השפה $\{w\}$. (כזכור, כל w היא סופית).
5. השפה: $L = \{w \in \Sigma^* : \ |w| \equiv 1 \ (mod\ 4)\}$.

שפה 4: תהי $w = a_1 a_2 \ldots a_3$, ותהי $L = \{w\}$. כיצד נבנה אס"ד לשפה?



רעיון הבנייה: נרצה לראות את רצף התווים $a_1 a_2 \ldots a_3$ לפי הסדר. אם הצלחנו – עוברים למצב מקבל. אחרת, מצב בור. אם ראינו את הרצף הנכון, ואחריו תו נוסף, זה גם הולך לבור. סקיצה:

הבעיה היא שאי אפשר לצייר אוטומט כזה, כי האורך לא ידוע. נגדיר את האוטומט $A = (Q, \Sigma, q_0, \delta, F)$ פורמלית:

$$Q = \{q_0, q_1, \ldots q_k, q_{pit}\}, \qquad F = \{q_k\}$$

$$\forall i \in [k] : \delta(q_{i-1}, a_i) = q_i$$

$$\forall i \in [k], \forall \sigma \in (\Sigma \setminus \{a_i\}) : \delta(q_{i-1}, \sigma) = q_{pit}$$

$$\forall \sigma \in \Sigma : \delta(q_k, \sigma) = q_{pit} = \delta(q_{pit}, \sigma)$$

כל אות במילה (לפי הסדר) היא מצב. רק האות האחרונה היא מצב מקבל. לכל אות במילה, אם נמצאים באות וקוראים את האות הבאה במילה, עוברים למצב הבא. אם קוראים כל אות אחרת, עוברים לבור. מ-q_k, אם נקרא כל אות נעבור לבור.

שפה 5: נבנה אוטומט שמקבל את השפה:

	σ
$* \to q_0$	q_1
q_1	q_2
q_2	q_3
q_3	q_0

2.1 שפות לא רגולריות

בהכרח קיימות שפות לא רגולריות, כי:

אוסף האוטומטים מעל א"ב Σ הוא בן-מניה. לעומת זאת, אוסף השפות מעל Σ אינו בן-מניה (כי $|P(\Sigma^*)| = 2^{|\Sigma^*|}$, לפי משפט קנטור).

לאס"ד יש כמות סופית של מצבי בקרה, אבל הקלט הוא מאורך לא מוגבל. כלומר הוא לא יכול לספור. ולכן:

דוגמה לשפה לא רגולרית: $L = \{0^n 1^n : n \geq 1\}$. כלומר: $01, 0011, 000111 \in L$.

הוכחה: נב"ש שקיים אוטומט A כך ש $L(A) = L$.

נתבונן בסדרת המצבים ... $q_0, q_1,$ כך ש $q_i = \hat{\delta}(q_0, 0^i)$. (המצב שנגיע אליו אחרי i אפסים).

כיוון שמספר המצבים סופי ואורך הסדרה לא, קיימים $i < j$ כך ש $q_i = q_j$, כלומר $\hat{\delta}(q_0, 0^i) = \hat{\delta}(q_0, 0^j)$. (שובך היונים).

ע"פ תכונת שרשור נקבל:

$$\hat{\delta}(q_0, 0^i 1^i) = \hat{\delta}(\hat{\delta}(q_0, 0^i), 1^i) = \hat{\delta}(\hat{\delta}(q_0, 0^j), 1^i) = \hat{\delta}(q_0, 0^j 1^i)$$

אבל, $\hat{\delta}(q_0, 0^i 1^i) \in F$ ואילו $\hat{\delta}(q_0, 0^j 1^i) \notin F$, בסתירה לשוויון.

3 סגירות שפות רגולריות תחת פעולות בוליאניות (תכונות סגור)

ניקח תכונות שונות של שפות רגולריות, ונבדוק אם תוצאת הפעולה תהיה בהכרח שפה רגולרית. התכונות שנבדוק: הכלה, משלים, חיתוך, איחוד, חיסור. יהיו L, L_1, L_2:

3.1 הכלה

נניח ש $L_1 \subseteq L_2$. אם L_2 שפה רגולרית, לא בהכרח גם L_1 תהיה רגולרית. דוגמה נגדית:

$$L_2 = \{w \in \{a,b\}^*\}$$

$$L_1 = \{a^n b^n : n \geq 1\}$$

וכבר הראינו בחלק הקודם שזו לא שפה רגולרית.

הכלה היא לא תכונת סגור.

3.2 משלים

השפה המשלימה של L היא כל המילים שאינן ב-L: $\bar{L} = \Sigma^* \setminus L$. אם L רגולרית, גם $\bar{L}$ בהכרח רגולרית. הוכחה:

נבנה אוטומט המקבל את השפה המשלימה. קבוצת המצבים, פונקציית המעברים, והא"ב יהיו זהים. מצבים מקבלים יהיו לא-מקבלים, ולהיפך. תיאור פורמלי:

בהינתן אוטומט $A = (Q, \Sigma, q_0, \delta, F)$ עבור שפה L, נבנה אוטומט עבור השפה $\bar{L}$: $\bar{A} = (Q, \Sigma, q_0, \delta, Q \setminus F)$. (אוטומט שמקבל את כל המצבים ש A לא מקבל). ונוכיח ש $\bar{L} = L(\bar{A})$:

$$w \in \bar{L} \Leftrightarrow w \notin L = L(A) \Leftrightarrow \hat{\delta}(q_0, w) \notin F \Leftrightarrow \hat{\delta}(q_0, w) \in Q \setminus F \Leftrightarrow w \in L(\bar{A})$$

אם ניקח אוטומט קיים ונהפוך כל מצב מקבל ל-לא מקבל ולהיפך, נקבל אוטומט משלים.

3.3 חיתוך

שפת החיתוך $L = L_1 \cap L_2$ מתקבלת את כל המילים שנמצאות גם ב-L_1 וגם ב-L_2. אם L_1, L_2 רגולריות, בהכרח גם L רגולרית. נוכיח ע"י בניית האוטומט המקבל את שפת החיתוך. הוא נקרא **אוטומט מכפלה**:

נניח ש L_1, L_2 שפות רגולריות מעל Σ, ויהיו: $A_1 = (Q_1, \Sigma, q_{01}, F_1, \delta_1),\ A_2 = (Q_2, \Sigma, q_{02}, F_2, \delta_2)$ אוטומטים המקבלים את L_1, L_2, בהתאמה. כדי להראות ש $L = L_1 \cap L_2$ רגולרית, נבנה אוטומט A כך ש $L(A) = L$:

- נבנה אוטומט אשר יחקה בו זמנית את הפעולות של A_1, A_2.
- המצבים של האוטומט החדש יהיו **זוגות** (q_1, q_2) אשר מייצגים את המצב שבו היה כל אחד מהאוטומטים לאחר קריאת המילה (או חלק ממנה).
- אוטומט שמצביו הם זוגות של שני אוטומטים אחרים נקרא אוטומט מכפלה – מלשון **מכפלה קרטזית** בין המצבים.
- בעת קריאת האות a, האוטומט יעבור ממצב (q_1, q_2) למצב $(\delta_1(q_1, a), \delta_2(q_2, a))$.

תיאור פורמלי של אוטומט המכפלה: $A=(Q,\Sigma,q_0,\delta,F)$ כאשר:

$$Q=Q_1\times Q_2,\qquad F=F_1\times F_2$$

$$q_0=(q_{01},q_{02})$$

$$\forall q_1\in Q_1,q_2\in Q_2:\delta\big((q_1,q_2),a\big)=\big(\delta_1(q_1,a),\delta_2(q_2,a)\big)$$

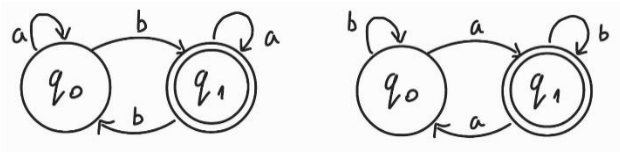
(נשים לב ש- δ מוגדר עבור תו בודד).

דוגמה: נבנה אס"ד לשפה הבאה:

$$L=\{w\in\{a,b\}^*:\#_a(w)\equiv 0(mod\ 2)\wedge\ \#_b(w)\equiv 1(mod\ 2)\}$$

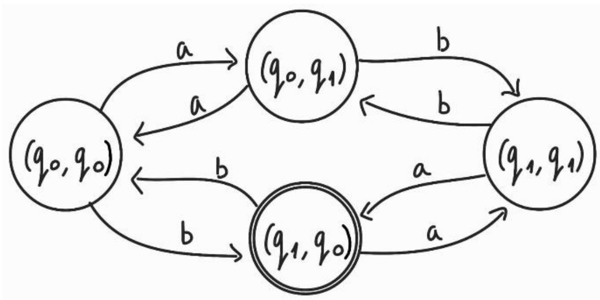
(שפת כל המילים מעל a,b שיש בהן מספר זוגי של a ומספר אי-זוגי של b).

האוטומטים A_1,A_2:



האוטומט A_2 מקבל מספר זוגי של a, ו-A_1 מקבל מספר אי-זוגי של b.

נבנה את אוטומט המכפלה $A_1\times A_2$: נאפיין קודם את המצבים, ונוסיף את פונקציית המעברים:



צ"ל ש $L(A)=L_1\cap L_2$. נשתמש בטענת עזר:

$$\forall w\in\Sigma^*,\forall(q_1,q_2)\in Q:\ \hat{\delta}\big((q_1,q_2),w\big)=(\hat{\delta}_1(q_1,w),\hat{\delta}_2(q_2,w))$$

כלומר, הבנייה עובדת גם למילים שלמות ולא רק תו בודד. הוכחה באינדוקציה על $|w|$:

בסיס: $|w|=0$, כלומר $|w|=\epsilon$. מהגדרת פונקציית המעברים נקבל:

$$\hat{\delta}\big((q_1,q_2),\epsilon\big)=(q_1,q_2)=\big(\delta_1(q_1,\epsilon),\delta_2(q_2,\epsilon)\big)$$

צעד: נניח שהטענה מתקיימת לכל $|u|<n$, ותהי $w=ua$ כך ש $|w|=n$. כלומר $|u|=n-1$. מתקיים:

$$\hat{\delta}\big((q_1,q_2),w\big)=^{א}\hat{\delta}\big((q_1,q_2),ua\big)=^{ב}\delta\big(\hat{\delta}\big((q_1,q_2),u\big),a\big)=^{ג}\delta\big(\hat{\delta}\big((q_1,q_2),u\big),a\big)=^{ד}$$
$$=\delta\Big(\big(\hat{\delta}_1(q_1,u),\hat{\delta}_2(q_2,u)\big),a\Big)=^{ה}\Big(\delta_1\big(\hat{\delta}_1(q_1,u),a\big),\delta_2\big(\hat{\delta}_2(q_2,u),a\big)\Big)=^{ו}$$
$$=\big(\delta_1(q_1,w),\delta_2(q_2,w)\big)$$

א – הצבה, ב – הגדרת $\hat{\delta}$, ג – הגדרת δ, ד – הנחת האינדוקציה, ה – הגדרת $\hat{\delta}$ עבור שני אוטומטים.

כעת, נוכיח ש $L(A)=L_1\cap L_2$:

$$w\in L(A)\Leftrightarrow\delta(q_0,w)\in F=F_1\times F_2$$
$$\Leftrightarrow\big[\hat{\delta}(q_0,w)=\hat{\delta}\big((q_{01},q_{02}),w\big)=\big(\delta_1(q_{01},w),\delta_2(q_{02},w)\big)\big]\in F_1\times F_2$$
$$\Leftrightarrow\big[\hat{\delta}_1(q_{01},w)\in F_1\ \wedge\ \hat{\delta}_2(q_{02},w)\in F_2\big]\Leftrightarrow w\in L_1\wedge w\in L_2\Leftrightarrow w\in L_1\cap L_2$$

3.4 איחוד

שפת האיחוד $L = L_1 \cup L_2$ מתקבלת את כל המילים שנמצאות גם ב-L_1 או ב-L_2. אם L_1, L_2 רגולריות, בהכרח גם L רגולרית.
ניתן להוכיח ע"י בניית האוטומט המקבל את שפת האיחוד, עם כלים שנלמד בהמשך (אסל"ד עם מסעי ϵ). ניתן להוכיח גם ע"י שימוש בסגירות לחיתוך ומשלים, עם חוקי דה-מורגן:

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

נבנה משלים לכל שפה, מכפלה, ואז משלים.

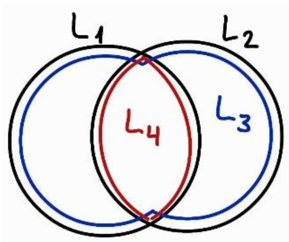
3.5 חיסור

שפת החיסור $L = L_1 \setminus L_2$ מתקבלת את כל המילים שנמצאות גם ב-L_1 ולא ב-L_2. אם L_1, L_2 רגולריות, בהכרח גם L רגולרית.
הוכחה: מתקיים:

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

שפות רגולריות סגורות לפעולות משלים וחיתוך, אז צד ימין רגולרי. ולכן גם צד שמאל רגולרי, כנדרש.

תרגיל 1: יהיו L_1, L_2, L_3, L_4 כך שמתקיים: $L_1 \cup L_2 = L_3,\ \ L_1 \cap L_2 = L_4$. נתון ש L_2, L_3, L_4 רגולריות.
נוכיח ש L_1 רגולרית: מתקיים: $L_1 = (L_3 \setminus L_2) \cup L_4$. מכיוון שאיחוד וחיסור הן תכונות סגור, גם L_1 רגולרית.



תרגיל 2: יהיו L_1, L_2, L_3 כך שמתקיים: $L_3 = L_2 \cup L_1$. נתון ש L_2, L_3 רגולריות. אז L_1 לא בהכרח רגולרית:

$$L_2 = \Sigma^*, \qquad L_1 = \{a^n b^n : n \geq 1\}$$

אז $L_1 \cup L_2 = L_2$. כלומר הנתונים מתקיימים, אבל L_1 לא רגולרית.

מצד שני, לא מובטח ש L_1 **אינה** רגולרית: אם נגדיר: $L_1 = L_2 = L_3 = \{a\}$, הנתונים מתקיימים ו- L_1 רגולרית.

תרגיל 3: יהיו L_1, L_2, L_3 כך שמתקיים: $L_3 = L_2 \cap L_1$. נתון ש L_2, L_3 רגולריות. אז L_1 לא בהכרח רגולרית:

$$L_2 = L_3 = \phi, \qquad L_1 = \{a^n b^n : n \geq 1\}$$

הרצאה 3

1 אוטומט סופי לא דטרמיניסטי

אוטומט סופי לא דטרמיניסטי – אסל"ד. (NFA – Nondeterministic finite automaton).

1.1 אי-דטרמיניזם

אסל"ד יכול להמשיך לכמה מצבים, או להיתקע. אס"ד נתקע אם הוא מגיע לבור. אסל"ד נתקע אם הגענו למקום שאין בו חץ יוצא שמתאים לאות הבאה בקלט.

מתחילים ממצב התחלתי נתון. מעבר ממצב ואות נתונה יכול להיות לקבוצת מצבים.

לכל מילה ייתכנו מספר חישובים אפשריים, או אף חישוב אפשרי.

האוטומט מקבל אם קיימת סדרת מעברים המובילה למצב מקבל.

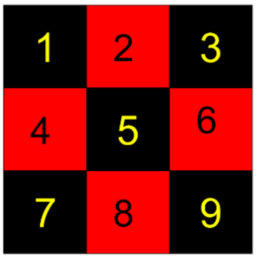
באופן אינטואיטיבי: אסל"ד תמיד "מנחש נכון" מהו המסלול שיוביל למצב מקבל, אם קיים כזה.

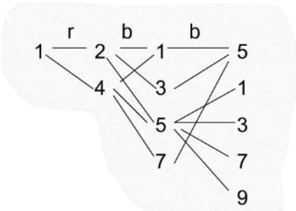
דוגמה 1: מעברים על לוח משבצות: קבוצת המצבים = קבוצת הריבועים על הלוח.

קלט: r = מעבר לריבוע שכן אדום. b = מעבר לריבוע שכן שחור.

מצב התחלתי ומצב מקבל הם ריבועים בפינות מנוגדות.

יש מסלול שמוביל ל9, ולכן rbb מתקבלת.

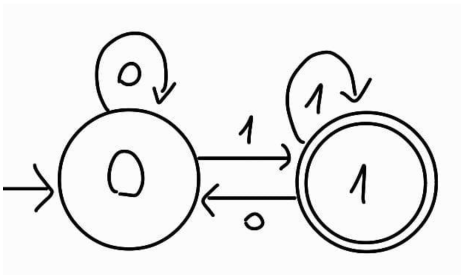




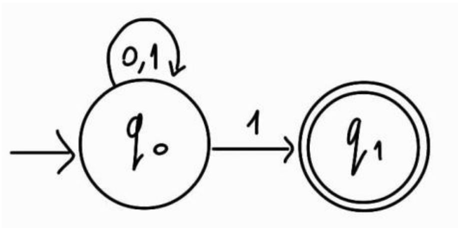
	r	b
→ 1	2,4	5
2	4,6	1,3,5
3	2,6	5
4	2,8	1,5,7
5	2,4,6,8	1,3,7,9
6	2,8	3,5,9
7	4,8	5
8	4,6	5,7,9
*9	6,8	5

דוגמה 2: נבנה אוטומט דטרמיניסטי לשפה:

$L = \{w1 : w \in \{0,1\}^*\}$:



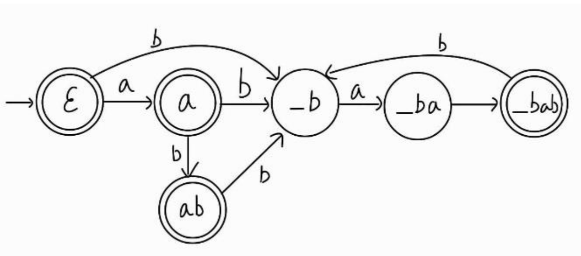
נבנה אסל"ד לאותה שפה:



דוגמה 3: נבנה אסל"ד לשפה: $L = \{\epsilon, a, ab\} \cdot \{bab\}^*$:

הסבר: החצי השמאלי זה כל המילים $\{\epsilon, a, ab\}$. כל אחת מהן מתקבלת. מכל אחת מהן, אפשר לעבור למצב ההתחלתי של צד ימין.

האוטומט לא דטרמיניסטי כי מהמילה "a" האות 'b' שולחת לשני כיוונים, ובגלל שאין שני חצים מכל מצב. אם אנחנו במצב "ab" ומקבלים 'a', ההתנהגות לא מוגדרת.



1.2 הגדרה פורמלית של אסל"ד – NFA

אסל"ד N הוא חמישייה $N = (Q, \Sigma, q_0, \delta, F)$, כאשר:

- Q היא קבוצת מצבים,
- Σ היא הא"ב הנתון,
- δ היא פונקציית מעברים,
- q_0 הוא המצב ההתחלתי,
- $F \subseteq Q$ היא קבוצת מצבים מקבלים.

פונקציית המעברים δ: $\delta(q,a)$ היא **קבוצת** מצבים. $\delta : (Q \times \Sigma) \to S \subseteq Q$.

הרחבת ההגדרה למחרוזות – הגדרה רקורסיבית:

בסיס: $\hat{\delta}(q,\epsilon) = \{q\}$.

צעד רקורסיבי: $\hat{\delta}(q,wa) = \bigcup_{p\in\hat{\delta}(q,w)} \delta(p,a)$. (אם אנחנו במצב q, המצבים שאפשר להגיע אליהם ע"י המילה wa הם כל המצבים שאפשר להגיע אליהם ע"י w, ואז a).

הרחבה לקבוצת מצבים: $\hat{\delta}(P,w) = \bigcup_{q\in P} \hat{\delta}(q,w)$. המצבים שאפשר להגיע אליהם מקבוצת מצבים, זה פשוט איחוד של כל מה שאפשר להגיע אליהם מכל אחד מהמצבים).

1.3 שפה של אסל"ד

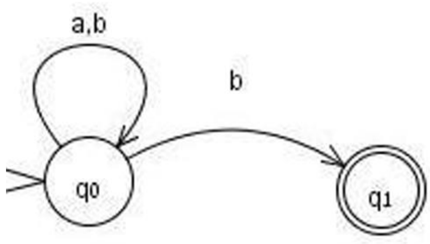
מחרוזת w מתקבלת אם $\hat{\delta}(q_0,w)$ מכיל לפחות מצב מקבל אחד. כלומר, קיים מסלול חישוב על w המכיל ב-q_0 ומסתיים במצב מקבל.

השפה של האוטומט היא קבוצת המחרוזות שהוא מקבל. עבור אסל"ד $N = (Q, \Sigma, q_0, \delta, F)$, השפה המתקבלת היא:

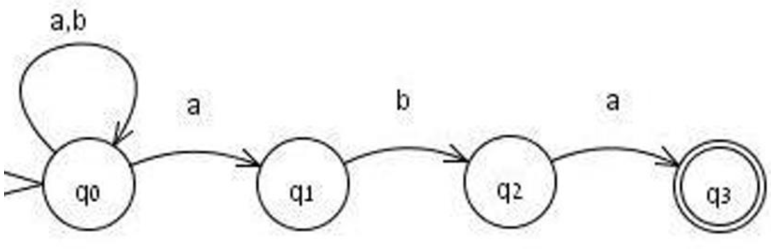
$$L(N) = \{w \in \Sigma^* : \ \hat{\delta}(q_0,w) \cap F \neq \phi\}$$

דוגמה 4:

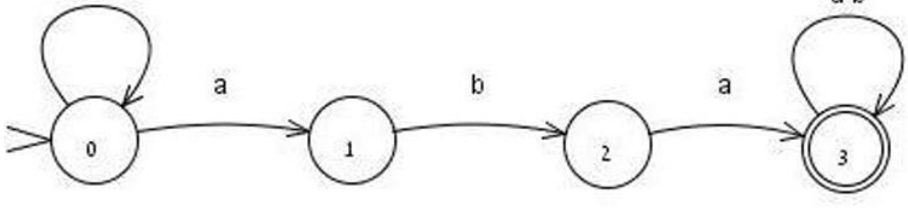
שפת המחרוזות המסתיימות ב-b:



שפת המחרוזות המסתיימות ב-aba:

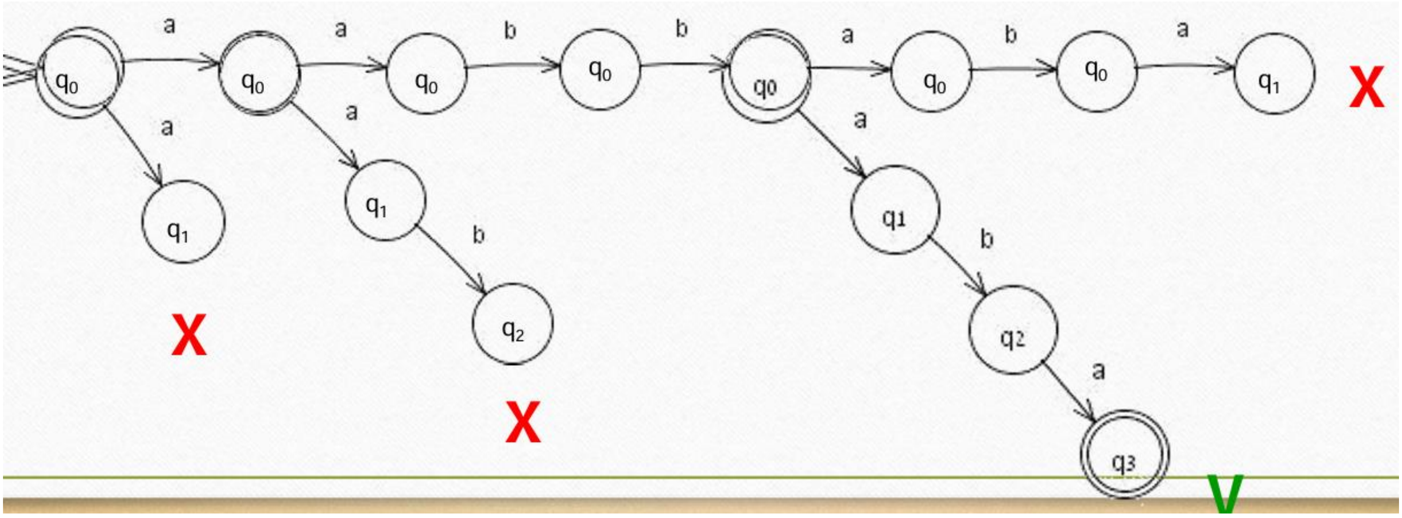


שפת המחרוזות המכילות aba:



בכל הדוגמאות, האוטומט "ינחש" מתי הוא מגיע למיקום המתאים במחרוזת כדי להתקדם.

דוגמה למסלולי חישוב על המילה: $aabbaba$: אם נעשה בחירה "לא נכונה", לא נגיע למצב מקבל. נבחר את המסלול שמוביל למצב מקבל:



2 שקילות של אס"ד ואסל"ד

מה כוח החישוב של אסל"ד ביחס לאס"ד? כלומר, האם קיימות שפות שמוד אחד מקבל והשני לא?

הגדרה: נאמר ששני אוטומטים הם **שקולים** אם הם מקבלים את אותה השפה.

טענה – שני המודלים שקולים. נוכיח בהכלה דו-כיוונית. בהינתן DFA נבנה NFA שקול, ולהיפך.

כיוון ראשון: בהינתן DFA, נבנה NFA שקול באופן הבא: אם $\delta_D(q,a) = p$, אז $\delta_N(q,a) = \{p\}$.

ההבדל הוא פשוט בסינטקס – מעבר לקבוצת מצבים במקום מצב יחיד. מבחינה לוגית (וציורית) הם אותו דבר.

2.1 אוטומט חזקה

כיוון שני: יהי אסל"ד $N = (Q, \Sigma, q_0, \delta_N, F_N)$. נבנה את האס"ד הבא:

- קבוצת המצבים היא $\mathcal{P}(Q)$ – קבוצת החזקה על Q, בלי $\emptyset$.
- הא"ב הוא Σ.
- המצב ההתחלתי היא $\{q_0\}$ (קבוצה שמכילה רק את q_0).
- קבוצת המצבים המקבלים היא F_D: כל תתי הקבוצות של Q, שמכילות איבר מ-F_N.
- פונקציית המעבר: $\delta_D(\{q_1, \dots q_k\}, a)$ היא קבוצה המתקבלת מאיחוד על כל $i = 1 \dots k$ של $\delta_N(q_i, a)$.

כלומר, אם קראנו אות, איך נדע לאן אנחנו עוברים? נעבור על כל אחד מהמצבים בקבוצת המצבים הנוכחית, ולכל אחד נבדוק לאן האות הזאת שולחת אותנו מהמצב הזה.

אוטומט כזה נקרא **אוטומט חזקה.**

2.2 בניית אוטומט חזקה

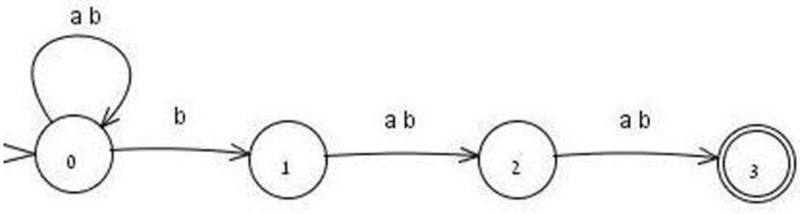
לא בהכרח צריך לבנות את כל מצבי אוטומט החזקה. ניתן לבנות רק את המצבים הניתנים להשגה: מתחילים במצב ההתחלתי, מתקדמים דרך המצבים שהגענו אליהם. עוצרים כשאין מצבים חדשים הניתנים להשגה.

דוגמה 6: נבנה אס"ד מתוך האסל"ד שראינו בדוגמה 1:

		r	b
→	1	2,4	5
	2	4,6	1,3,5
	3	2,6	5
	4	2,8	1,5,7
	5	2,4,6,8	1,3,7,9
	6	2,8	3,5,9
	7	4,8	5
	8	4,6	5,7,9
*	9	6,8	5

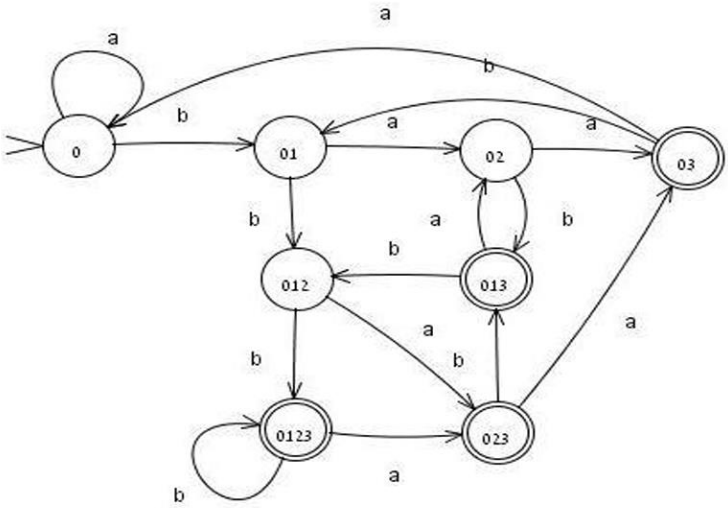
		r	b
→	{1}	{2,4}	{5}
	{2,4}	{2,4,6,8}	{1,3,5,7}
	{5}	{2,4,6,8}	{1,3,7,9}
	{2,4,6,8}	{2,4,6,8}	{1,3,5,7,9}
	{1,3,5,7}	{2,4,6,8}	{1,3,5,7,9}
*	{1,3,7,9}	{2,4,6,8}	{5}
*	{1,3,5,7,9}	{2,4,6,8}	{1,3,5,7,9}

דוגמה 7: שפת כל המילים מעל $\{a,b\}$ שבהן האות השלישית מהסוף היא b. האסל"ד הוא:

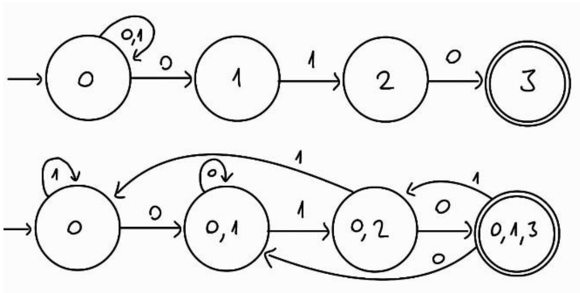


נבנה אס"ד:

	a	b
{0}	{0}	{0,1}
{0,1}	{0,2}	{0,1,2}
{0,2}	{0,3}	{0,1,3}
{0,1,2}	{0,2,3}	{0,1,2,3}
{0,3}	{0}	{0,1}
{0,1,3}	{0,2}	{0,1,2}
{0,2,3}	{0,3}	{0,1,3}
{0,1,2,3}	{0,2,3}	{0,1,2,3}



דוגמה 8: מחרוזות בינאריות שמסתיימות ב 010: ניקח את האסל"ד ונבנה אס"ד:



2.3 הוכחת שקילות

נוכיח באינדוקציה על $|w|$ שמתקיים: $\hat{\delta}_N(q_0,w) = \hat{\delta}_N(\{q_0\},w)$.

כלומר: כל מילה, באסל"ד היא מובילה לקבוצת מצבים. צריך להוכיח שבאס"ד, המצב (היחיד) שהמילה הזו מובילה אליו, הוא המצב שקרוי על שם אותה קבוצת מצבים.

בסיס: עבור $w=\epsilon$, $\hat{\delta}_D(\{q_0\},\epsilon) = \{q_0\} = \hat{\delta}_N(q_0,\epsilon)$.

צעד: נניח שהטענה נכונה ל-x, ותהי $w = xa$.

$$\hat{\delta}_D(\{q_0\},w) = \hat{\delta}_D(\{q_0\},xa) =^{\text{א}} \delta_D\big(\hat{\delta}_D(\{q_0\},x),a\big) =^{\text{ב}} \delta_D\big(\hat{\delta}_N(q_0,x),a\big) =^{\text{ג}} \delta_N\big(\hat{\delta}_N(q_0,x),a\big) = \hat{\delta}_N(q_0,w)$$

א – נשים לב שזה מעבר מקריאת מילה לקריאת אות. קבוצת המצבים שאפשר להגיע אליהם מ-q_0 ע"י קריאת xa, זהה לקבוצת המצבים שאשפר להגיע אליהם מ"קבוצת המצבים שאפשר להגיע אליהם מ-q_0 על ידי קריאת x" ע"י קריאת a.

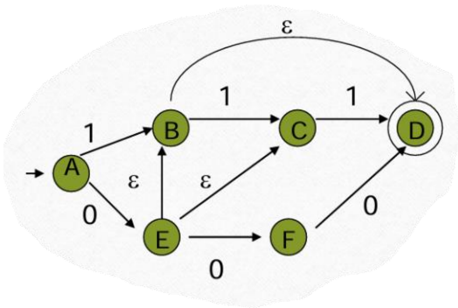
ב, ג – מהגדרת פונקציית המעברים בבנייה. ההבדל הוא בסינטקס בלבד.

כעת, נקבל:

$$w \in L(D) \Leftrightarrow^{\text{א}} \hat{\delta}_D(\{q_0\},w) \in F_D \Leftrightarrow^{\text{ב}} \hat{\delta}_D(\{q_0\},w) \cap F_N \neq \emptyset \Leftrightarrow^{\text{ג}} \hat{\delta}_N(q_0,w) \cap F_N \neq \emptyset \Leftrightarrow^{\text{ד}} w \in L(N)$$

א – הגדרה, ב – לוגיקה, ג – לפי ההוכחה לעיל, ד – הגדרה.

3 מסעי אפסילון



מסעי אפסילון מאפשרים מעבר ממצב למצב בעזרת קלט ϵ.

$\delta: Q \times (\Sigma \cup \{\epsilon\}) \to P(Q)$. לדוגמה:

אז באוטומט הזה, $\delta(A,0) = \{E,B,C,D\}$.

3.1 סגור של קבוצת מצבים

סגור אפסילון של מצב q הוא קבוצת המצבים שניתן להגיע אליהם מ-q ע"י מסעי אפסילון בלבד. נסמן: $CL^{\epsilon}(q)$.

בדוגמה: $CL^{\epsilon}(A) = \{A\}, CL^{\epsilon}(E) = \{B,C,D,E\}$

הסגור של קבוצת מצבים P הוא איחוד כל הסגורים של האיברים ב-P: $CL^{\epsilon}(P) = \bigcup_{q\in P} CL^{\epsilon}(q)$.

בדוגמה: $CL^{\epsilon}(\{A,E,F\}) = CL^{\epsilon}(A) \cup CL^{\epsilon}(E) \cup CL^{\epsilon}(F) = \{B,C,D,E\}$.

3.2 הרחבה למילים של פונקציית המעברים

בסיס: $\delta'(q,\epsilon) = CL^{\epsilon}(q)$

שלב האינדוקציה: $\delta'(q,xa) = \bigcup_{p\in\hat{\delta}'(q,x)} CL^{\epsilon}\big(\delta(p,a)\big)$.

כלומר: לכל אחד מהמצבים שאפשר להגיע אליהם ע"י קריאת x, נבדוק לאן אפשר להגיע משם ע"י קריאת a, ונבדוק מה הסגור אפסילון של כל אחד מהם.

אינטואיציה: $\hat{\delta}'(q,w)$ היא קבוצת המצבים שאליהם ניתן להגיע מ-q ע"י קריאת w ושימוש אפשרי במסעי אפסילון בכל שלב.
בדוגמה: $\delta'(A,\epsilon) = CL^{\epsilon}(A) = \{A\}$,

$$\delta'(A,0) = \bigcup_{p\in\delta'(A,\epsilon)} CL^{\epsilon}\big(\delta(p,0)\big) = CL^{\epsilon}\big(\delta(A,0)\big) = CL^{\epsilon}(\{E\}) = \{B,C,D,E\}$$

$$\delta'(A,01) = \bigcup_{p\in\delta'(A,0)} CL^{\epsilon}\big(\delta(p,1)\big) = \bigcup_{p\in\{B,C,D,E\}} CL^{\epsilon}\big(\delta(p,1)\big) = CL^{\epsilon}(C) \cup CL^{\epsilon}(D) = \{C,D\}$$

3.3 שפה של $\epsilon - NFA$

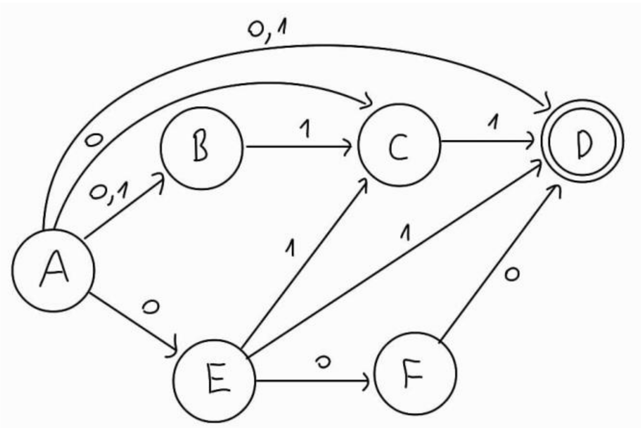
שפה של אוטומט לא דטרמיניסטי עם מסעי אפסילון היא קבוצת כל המחרוזות w שעבורן $\hat{\delta}'(q_0,w)$ מכילה מצב מקבל.

שקילות בין אל"ד עם מסעי אפסילון לבין אל"ד בלי מסעי אפסילון:

כיוון ראשון: כל אל"ד בלי מסעי אפסילון הוא גם אל"ד עם מסעי אפסילון – באופן טריוויאלי.

כיוון שני: מכל $\epsilon - NFA$, אפשר לבנות NFA רגיל. נראה שניתן לחבר מסע אפסילון עם המעבר הבא על הקלט:

נתון אסל"ד עם מסעי אפסילון בו קבוצת מצבים Q, א"ב Σ, מצב התחלתי q_0, קבוצת מצבים מקבלים F_E, ופונקציית מעברים δ_E.
נבנה אסל"ד $A(Q,\Sigma,q_0,F_N,\delta_N)$. פונקציית המעברים מוגדרת: $\delta_N(q,a) \coloneqq \delta_E(q,a) = \bigcup_{p\in CL^{\epsilon}(q)} CL^{\epsilon}\big(\delta_E(p,a)\big)$. כלומר, קבוצת המצבים שניתן להגיע אליהם מ-q ע"י שימוש במסעי אפסילון וקריאת a.



הקבוצה F_N:

אם אפסילון לא שייך לשפת האוטומט E, אז $F_N = F_E$.

אם אפסילון כן שייך לשפת האוטומט אז $F_N = F_E \cup \{q_0\}$.

לכל $w \in \Sigma^+$, נוכיח באינדוקציה על $|w|$ שמתקיים: $\hat{\delta}_N(q_0,w) = \hat{\delta}_E(q_0,w)$.

בסיס: עבור $|w| = 1$, $w = \sigma$. לפי הגדרת δ_N: $\delta_N(q_0,\sigma) = \delta_E(q_0,\sigma)$.

צעד: נניח שהטענה נכונה עבור $u \neq \epsilon$ ונוכיח נכונות עבור $w = u\sigma$:

$$\hat{\delta}_N(q_0,u\sigma) = \delta_N\big(\hat{\delta}_N(q_0,u),\sigma\big) \overset{\text{א}}{=} \delta_N\big(\hat{\delta}_E(q_0,u),\sigma\big) \overset{\text{ב}}{=} \delta_E\big(\hat{\delta}_E(q_0,u),\sigma\big) \overset{\text{ג}}{=} \hat{\delta}_E(q_0,u\sigma)$$

א – לפי הנ"א, ב – לפי הגדרת δ_N לתו יחיד, ג – לפי הגדרת δ_E.

כדי להשלים את הוכחת השקילות עלינו להראות כי שני האוטומטים מקבלים את אותן המילים, כלומר ש:

$$\hat{\delta}_N(q_0,w) \cap F_N \neq \emptyset \Leftrightarrow \hat{\delta}_E(q_0,w) \cap F_E \neq \emptyset$$

עבור $w = \epsilon$, זה נובע מהגדרת F_N.

עבור $w \neq \epsilon$, זה נובע מטענת העזר (כי ההתקדמות לפי פונקציית המעברים זהה) והגדרת F_N.

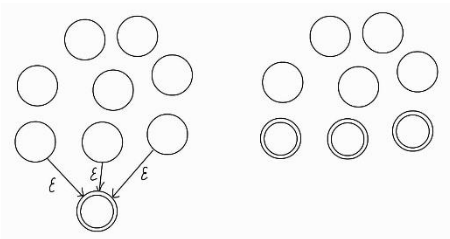
לכן, w מתקבלת ע"י האוטומט המקורי (עם מסעי אפסילון) אמ"מ היא מתקבלת ע"י האוטומט החדש.

4 מודלים שקולים נוספים

תרגיל: האם $\epsilon - NFA$ עם מצב מקבל יחיד שקול ל-DFA רגיל? כן.

כיוון ראשון: יהי $E = (Q,\Sigma,\delta,q_0,q_f)$. אז ניתן להמיר אותו ל-$NFA$, וכל NFA אפשר להמיר לDFA.

כיוון שני: יהי $A = (Q,\Sigma,\delta,q_0,F)$. (עם מספר לא ידוע של מצבים מקבלים).



נבנה E עם מצב מקבל יחיד כך: $E = (Q \cup \{q_f\},\Sigma,\delta_E,q_0,q_f)$,

$\forall \sigma \in \Sigma, \forall q \in Q : \delta_E(q,\sigma) = \delta(q,\sigma)$, $\forall q \in F : \delta_E(q,\epsilon) = q_f$

כלומר, ניקח את אותה קבוצת מצבים בתוספת q_f, (המצב המקבל) עם אותם המעברים.
ומכל מצב שהיה מצב מקבל באוטומט המקורי יהיה מסע אפסילון למצב המקבל:

האם DFA עם מצב מקבל יחיד שקול ל-DFA רגיל? לא.

נוכיח בעזרת שפה הדורשת שני מצבים מקבלים ב-DFA. (כלומר אי אפשר לייצר לה DFA רגיל).

תהי $L = \{\sigma w \sigma : \sigma \in \{0,1\}, w \in \{0,1\}^*\}$. נב"ש כי L דורשת מצב מקבל יחיד ב-DFA, נקרא לו q_f.

נשקול את המילים $1w1, 0w0$. שתיהן בשפה ולכן: $\hat{\delta}(q_0,1w1) = \hat{\delta}(q_0,0w0) = q_f$.

נוסיף את התו 0:

$$\hat{\delta}(q_0,0w00) = \delta\big(\hat{\delta}(q_0,0w0),0\big) = \delta\big(q_f,0\big) = q_f$$
$$\hat{\delta}(q_0,1w10) = \delta\big(\hat{\delta}(q_0,1w1),0\big) = \delta\big(q_f,0\big) = q_f$$

שזו סתירה, כי $1w10 \notin L$.

4.1 שקילות נוספת לשפה רגולרית

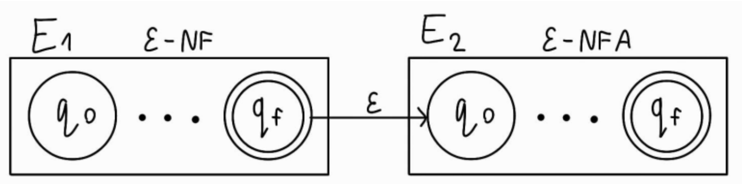
הוכנו כבר ששפה L רגולרית אמ"מ קיים לה אס"ד. כעת, הוכחנו ש: $\epsilon - NFA \equiv NFA \equiv DFA$, כלומר נוכל להרחיב את הטענה: **שפה L רגולרית אמ"מ קיים לה אס"ד / אסל"ד / אסל"ד אם מסעי אפסילון**.

כלומר, במקום להראות טענות על DFA נוכל להראות אותן על $\epsilon - NFA$ באותה מידה, ולקבל שהטענה עובדת גם עבור DFA.

הראנו בהרצאות הקודמות שבהינתן L_1, L_2 רגולריות, אזי גם $L_1 \setminus L_2, \overline{L_1}, L_1 \cap L_2, L_1 \cup L_2$ גם רגולריות. כמו כן, ראינו ש $L \subseteq L_1$ לא בהכרח רגולרית.

נטען שגם $(L_1 \cdot L_2), (L_1)^R, (L_1)^*, (L_1 \Delta L_2)$ כולן רגולריות:

4.2 $L_1 \cdot L_2$



L_1, L_2 רגולריות ולכן קיימים להן $\epsilon - NFA$ עם מצב מקבל יחיד. נבנה $\epsilon - NFA$ עבור $L_1 \cdot L_2$ בצורה הבאה:

4.3 L_1^R

L_1 רגולרית ולכן קיים לה $\epsilon - NFA$ עם מצב מקבל יחיד $E = (Q, \Sigma, \delta, q_0, q_f)$. נבנה $\epsilon - NFA$ עבור L_1^R:

$$E' = (Q, \Sigma, \delta', q_0', q_f')$$

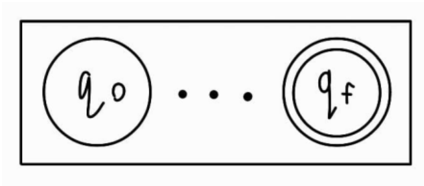
$q_0' = q_f$ כלומר, התחלת חישוב האוטומט החדש הוא מהמצב המקבל היחיד של האוטומט המקורי.

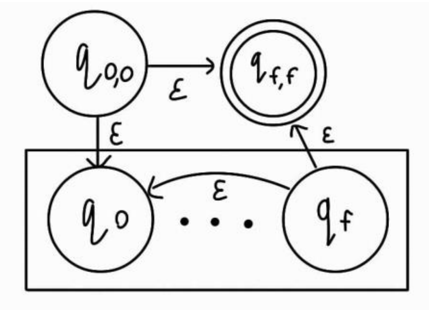
$$\forall q \in Q, \forall \sigma \in \Sigma \cup \{\epsilon\} : p \in \delta(q, \sigma) \Longrightarrow q \in \delta'(p, \sigma)$$

כלומר, הופכים את הכיוונים של כל החיצים מהאוטומט המקורי.

$q_f' = q_0$ כלומר, המצב המקבל הוא q_0 הישן. (ובפועל, גם כל מצב שיש ממנו מסע אפסילון למצב המקבל).

4.4 L_1^*





L_1 רגולרית ולכן קיים לה $\epsilon - NFA$ עם מצב מקבל יחיד. נבנה ממנו אוטומט:

נוכיח שקילות:

כיוון ראשון: ϵ מתקבל ע"י האוטומט כי יש מסע אפסילון מ-q_0^* ל-q_f^*. עבור כל מילה שהיא שרשור של מספר סופי של מילים מ-L, נעבור בתוך הלולאה הפנימית שבאיור האוטומט, ולבסוף נעבור עם מסע האפסילון ל-q_f^*. כך ניתן להגדיר מסלול חישוב מקבל עבור כל $w \in L^+$. פורמלית – צריך לעשות אינדוקציה על מספר השרשורים.

כיוון שני: נניח ש E^* מקבל את w, ונתבונן במסלול חישוב מקבל. מקרה א': המסלול מבצע מסע מ-q_0^* ל-q_f^*. במקרה זה, $w = \epsilon$, ולכן $w \in L^*$. מקרה ב': המסלול מגיע $k \geq 1$ פעמים ל-q_f, ולאחר כל ביקור ב-q_f (פרט לאחרון) מבצע מסע אפסילון ל-q_0. במקרה זה, ניתן לרשום $w = w_1 w_2 \cdots w_k$, כאשר כל w_i מתקבלת ע"י E. לכן במקרה זה $w \in L^*$.

4.5 $L_1 \Delta L_2$

$$L_1 \Delta L_2 = \{w \in L_1 \wedge w \notin L_2\} \cup \{w \in L_2 \wedge w \notin L_1\} = \{w \in L_1 \wedge w \in \overline{L_2}\} \cup \{w \in \overline{L_1} \wedge w \in L_2\}$$

כל הפעולות בדרך הן פעולות שמשמרות רגולריות.

4.6 $L_1 \cup L_2$

כבר הראינו באמצעות דה-מורגן. נראה בעוד דרך:

L_1 רגולרית ולכן קיים לה $\epsilon - NFA$. L_2 רגולרית ולכן קיים לה $\epsilon - NFA$. נבנה אוטומט עבור $L_1 \cup L_2$ בצורה הבאה:

$$E_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1), \qquad E_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$$

$$E = (Q_1 \cup Q_2 \cup \{q_0\}, \Sigma, \delta, q_0, F_1 \cup F_2)$$

$$\forall q \in Q_1, \sigma \in \Sigma : \delta(q,\sigma) = \delta_1(q,\sigma), \qquad \forall q \in Q_2, \sigma \in \Sigma : \delta(q,\sigma) = \delta_2(q,\sigma)$$

$$\delta(q_0, \epsilon) = \{q_{01}, q_{02}\}$$

הרצאה 4

1 ביטויים רגולריים

עוד דרך לתאר שפה רגולרית.

1.1 הגדרה

אוסף הביטויים הרגולריים מעל א"ב Σ מסומן R_Σ, ומוגדר באינדוקציה מבנית באופן הבא:

אטומים:

- $\phi, \epsilon \in R$. הקבוצה הריקה והתו הריק.
- $\forall\sigma \in \Sigma, \sigma \in R$. כל אות בא"ב.

פעולות יצירה:

- אם $r_1, r_2 \in R$, אזי: $(r_1 + r_2) \in R$, $(r_1 \cdot r_2) \in R$.
- אם $r \in R$ אז $r^* \in R$.

דוגמאות: הביטויים הבאים הם ביטויים רגולריים מעל $\Sigma = \{a, b\}$:

$$\phi, \quad \epsilon, \quad a, \quad b, \quad (\epsilon + b), \quad \big((\epsilon + b) \cdot b\big), \quad \phi^*[= \epsilon], \quad \big((\epsilon + a) \cdot b^*\big)$$

1.2 שפה של ביטוי רגולרי

תהי $L[r]$ השפה שמציין הביטוי r. נגדיר את הפונקציה: $L : R \to 2^{\Sigma^*}$:

- $L[\phi] = \phi$
- $L[\epsilon] = \epsilon$
- $\forall\sigma \in \Sigma : L[\sigma] = \sigma$
- אם $r_1, r_2 \in R$, אז:
  - $L[(r_1 + r_2)] = L[r_1] \cup L[r_2]$
  - $L[(r_1 \cdot r_2)] = L[r_1] \cdot L[r_2]$
- אם $r \in R$, אז $L[(r^*)] = (L[r])^*$

דוגמה: $r = \Big(\big((a + b) + c\big) + d\Big)^*$.

$$L[r] = L\left[\Big(\big((a + b) + c\big) + d\Big)^*\right] = \left(L\left[\Big(\big((a + b) + c\big) + d\Big)\right]\right)^* = \Big(L\big[\big((a + b) + c\big)\big] \cup L[d]\Big)^*$$
$$= (L[(a + b)] \cup L[c] \cup L[d])^* = \cdots = (a + b + c + d)^*$$

1.3 קיצורי כתיבה של ביטויים רגולריים

אם r ביטוי רגולרי, נסמן ב- r^+ את הביטוי הרגולרי $\big(r \cdot (r^*)\big)$. ("איטרציה לא ריקה").

נקבע סדר קדימויות כדי שנוכל להשמיט סוגריים:

1. איטרציה: * בקדימות גבוהה.
2. שרשור: $\cdot$ בקדימות בינונית.
3. איחוד: + בקדימות נמוכה.

בנוסף, בדרך כלל נשמיט את האופרטור של השרשור.

לעיתים נשתמש בביטוי הרגולרי לציון השפה שהוא מייצג.

1.4 דוגמאות נוספות

- $a^*b^*c^*$. $L = \{w \in \{a,b,c\}^* : w = a^i b^j c^k, 0 \le i,j,k\}$
- שפת כל המילים מעל $\Sigma = \{a,b\}$. $\Sigma^* = (a+b)^*$.
- שפת כל המילים באורך זוגי מעל $\Sigma = \{a,b\}$. $\big((a+b)(a+b)\big)^* = (\Sigma\Sigma)^*$.

2 שקילות ביטויים רגולריים לאוטומטים

2.1 כיוון ראשון

משפט: לכל $r \in R$ מעל Σ מתקיים כי $L[r]$ היא שפה רגולרית. הוכחה באינדוקציה מבנית על r:

בסיס: $L[\phi] = \phi,\ L[\epsilon] = \epsilon,\ \forall\sigma \in \Sigma : L[\sigma] = \{\sigma\}$ כולן רגולריות.

צעד האינדוקציה נובע ישירות מסגירות השפות הרגולריות תחת פעולות רגולריות:

נניח שהטענה נכונה עבור ביטויים r_1, r_2 ונוכיח שהיא נכונה עבור: $r_1 + r_2,\ r_1 r_2,\ (r_1^*)$. לפי סגירות לאיחוד, שרשור, ואיטרציה:

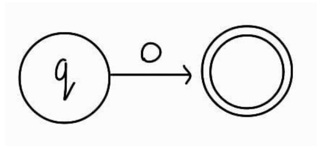
$L[r_1 + r_2] = L[r_1] \cup L[r_2],\ L[r_1 r_2] = L[r_1]L[r_2],\ L[r_1^*] = (L[r_1])^*$ כולן רגולריות.

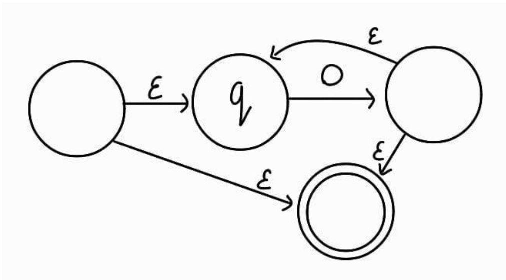
2.2 בניית אוטומט מתוך ביטוי רגולרי

נבנה אוטומט המקבל את השפה שמציין ביטוי רגולרי נתון לפי המשפט הקודם:

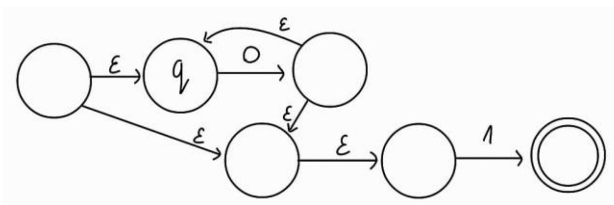
לדוגמה, תהי $L = 0^*1 + 1^*0$

נבנה אוטומט שמקבל את 0, ונרחיב אותו בשביל *0:

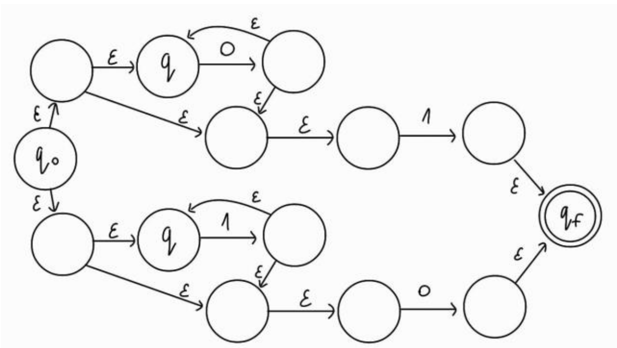




נוסיף את השרשור עם 1:



נבנה אחד דומה בשביל 1*0, ונחבר ביניהם:



2.3 כיוון שני

משפט: לכל שפה רגולרית $L \subseteq \Sigma^*$ קיים ביטוי רגולרי r כך ש- $L[r] = L$.

הוכחה: L רגולרית, לכן קיים לה אס"ד $A = (\Sigma, \{q_1 \dots q_m\}, q_1, \delta, F)$ כך ש-$L(A) = L$.

לכל i,j,k נסמן $L_{i,j}^k$ את השפה שכוללת את המילים שמובילות את האוטומט מ-q_i ל-q_j בלי לעבור דרך מצב שמספרו גדול מ-k.
("לעבור דרך" אינו כולל את המצב שממנו יוצאים והמצב שאליו מגיעים. זוכרים "קודקודי ביניים" במסלולים קצרים באלגו 1?)

פורמלית:

$$L_{i,j}^{k} = \{w :\ \hat{\delta}(q_i,w) = q_j, \forall u,v \neq w, uv = w : \delta(q_i,u) = q_\ell \Rightarrow n \le k\}$$

כל המילים w כך ש: w מובילה מ-q_i ל-q_j. ולכל שתי מילים (שהן לא w) שהשרשור שלהן הוא w, אם u מובילה מ-q_i ל-q_ℓ זה אומר ש $\ell \le k$.

לפי הגדרה הקודמת, כיוון שאין מצב גדול מ-m הרי ש-$L_{i,j}^{m}$ כוללת את כל המילים המובילות את האוטומט מ-q_i ל-q_j. ובפרט:

$$L(A) = \bigcup_{q_j \in F} L_{1,j}^{m}$$

שימו לב שסימנו את המצב ההתחלתי ב-q_1.

השפה $L(A)$ היא איחוד של מספר סופי של שפות. לכן, אם נמצא לכל $L_{i,j}^{k}$ ביטוי רגולרי, נוכל למצוא ביטוי רגולרי עבור $L(A)$.

נוכיח באינדוקציה כי לכל i,j,k ניתן לבנות ביטוי רגולרי ל-$L_{i,j}^{k}$. יהיו i,j.

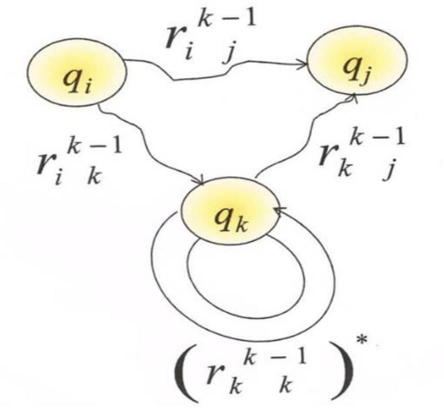
בסיס: עבור $k = 0$, יתכן רק מעבר ישיר מ-q_i ל-q_j. (צעד אחד אם הם שונים, אפס צעדים אם הם שווים).

לכן כל מילה $w \in L_{i,j}^{0}$ היא בעלת אורך לכל היותר 1. קל לראות כי לשפה זו יש ביטוי רגולרי. לדוגמה $\epsilon, (\sigma_1 + \sigma_2 + \cdots + \sigma_n)$.
אם לא קיים מעבר מ-q_i ל-q_j בצעד יחיד, הרי שהביטוי הרגולרי המתאים הוא ϕ.

הגדרה רקורסיבית של $L_{i,j}^{k}$: נניח כי עבור $k-1$ ניתן לבנות ביטוי רגולרי ל- $L_{i,j}^{k-1}$, ונבנה ביטוי רגולרי - $L_{i,j}^{k}$.

עבור $k > 0$, ב-$L_{i,j}^{k}$ קיימים שני סוגי מילים:

1. מילים שלא גורמות ל-A לעבור דרך q_k – אלה מילים ששייכות גם ל-$L_{i,j}^{k-1}$.
2. מילים שכן גורמות ל-A לעבור דרך q_k – ולכן לא שייכות ל-$L_{i,j}^{k-1}$.



את המילים הגורמות ל-A לעבור דרך q_k אפשר לחלק ל-3 חלקים uvw באופן הבא:

רישא u המובילה את A לביקור ראשון ב-q_k. $u \in L_{i,k}^{k-1}$.

חלק אמצעי v הגורם ל-A לבצע מספר סיבובים תוך חזרה ל-q_k. $v \in \left(L_{k,k}^{k-1}\right)^*$.

סיפא w המובילה את A מביקורו האחרון ב-q_k ל-q_j. $w \in L_{k,j}^{k-1}$.

סה"כ בניית הביטוי הרגולרי עבור האוטומט:

$$r_{i,j}^{k} = r_{i,j}^{k-1} + r_{i,k}^{k-1}\left(r_{k,k}^{k-1}\right)^* + r_{k,j}^{k-1}$$

2.4 משפט קליני

משפחת השפות הרגולריות היא הקבוצה הקטנה ביותר המכילה את כל השפות הסופיות והסגורה תחת הפעולות הרגולריות. הוכחה: בעצם נוכיח שקבוצת שפות מכילה את כל השפות הסופיות וסגורה לפעולות רגולריות אמ"מ היא משפחת השפות הרגולריות.

כיוון ראשון: אנחנו כבר יודעים שכל שפה סופית היא רגולרית. הוכחנו שקבוצת השפות הרגולריות סגורה לפעולות רגולריות.

כיוון שני: נובע מהמשפט האחרון, לפיו לכל שפה רגולרית קיים ביטוי רגולרי המציין אותה. כל קבוצה המכילה את השפות הסופיות והסגורה לפעולות רגולריות חייבת להכיל את כל השפות המצוינות ע"י ביטויים רגולריים, כלומר את השפות הרגולריות.

3 זהויות בין ביטויים רגולריים

לכל שפה קיימים הרבה ביטויים רגולריים המציינים אותה, ולכן נרצה לדעת מתי ביטויים רגולריים הם שקולים.

3.1 דוגמה 1

יהיו $r_1 = (0^*1)^*, \quad r_2 = \epsilon + (0+1)^*1$. נוכיח כי ביטויים אלה מייצגים את אותה השפה, ע"י הכלה דו כיוונית.

אינטואיטיבית: השפה הראשונה היא: איטרציה על: 0 איטרציה, משורשר עם 1.
השפה השנייה היא: אפשר לקחת: אפסילון, או שניקח מצד ימין: איטרציה על 0 או 1, ושרשור עם 1.

כיוון ראשון: נניח כי $w \in L[r_1]$. אזי $w = \epsilon$, או:

$$\exists w_1, \dots, w_n \in L[0^*1] : w = w_1 w_2 \cdots w_n$$

אם $w = \epsilon$ א בוודאי $w \in \{\epsilon\} \cup \{0,1\}^*\{1\} = L[r_2]$.

אחרת, w מסתיימת ב-1, ולכן $w \in \{\{0,1\}^*\{1\}\} \subseteq L[r_2]$.

כיוון שני: נניח כי $w \in L[r_2] = \{\epsilon\} \cup \{0,1\}^*\{1\}$.

אם $w = \epsilon$ אז בוודאי $w \in (\{0\}\{1\})^* = L[r_1]$.

אחרת, נוכל לכתוב $w = x1$ כאשר $x \in L[(0+1)^*]$. נניח כי ב-x יש k מופעים של 1.

במקרה זה נוכל לכתוב $x = y_1 1 y_2 1 \cdots y_k 1 y_{k+1}$, כאשר לכל $1 \leq n \leq k+1$ מתקיים $y_i \in \{0\}^*$.

ואז בעצם $w = (y_1 1)(y_2 1) \cdots (y_k 1)(y_{k+1} 1)$ ולכן $w \in L[r_1]$.

3.2 דוגמה 2

נראה שהביטויים $(0+1)^*, \ 0^* + 1^*$ אינם שקולים. אינטואיטיבית, כי הראשון זה כל המילים שהן רק 0 או רק 1, והשני זה כל המחרוזות הבינאריות. דוגמה נגדית פורמלית:

$01 \in L[(0+1)^*]$ כי $(0+1)^*$ היא שפת כל המילים מעל $\{0,1\}$. לעומת זאת, $01 \notin L[0^*]$ וגם $01 \notin L[1^*]$ ולכן $01 \notin L[0^*] \cup L[1^*] = L[0^* + 1^*]$.

3.3 הוכיחו/הפריכו

$(1+0)^* = (01^*)^* + (0^*1)^*$ – הפרכה: $10 \in L[(1+0)^*]$, אבל $10 \notin L[(0^*1)^*]$ וגם $10 \notin L[(01^*)^*]$.

$1(01)^* = (10)^*1$ – הוכחה: נראה הכלה דו-כיוונית:

כיוון ראשון: תהי $w \in L[1(01)^*]$. נראה באינדוקציה על $|w|$ ש $w \in L[(10)^*1]$.

בסיס: עבור $|w| = 1$, $w = 1$ ואכן $1 \in L[(10)^*1]$.

נניח שהטענה נכונה עבור $1w$ כאשר $|w| = t$, ונראה עבור $1x$ כאשר $|x| = t + 2$. היות ו- $1x \in L[1(01)^*]$, ניתן לרשום $1x = 1w01$. לפי הנ"א, $1w \in L[(10)^*1]$ ולכן:

$$1x = 1w01 = (10)^k 101 = (10)^{k+1} 1 \in L[(10)^*1]$$

כיוון שני: בתרגול.

4 מעבר מאוטומט לביטוי רגולרי

למדנו את הוכחת השקילות בין אוטומט לביטוי רגולרי. לפעמים, כדי לכתוב ביטוי לשפה רגולרית, יהיה נוח לבצע את המעבר הזה בפועל, אבל הפעלת השקילות עלולה להיות ארוכה.

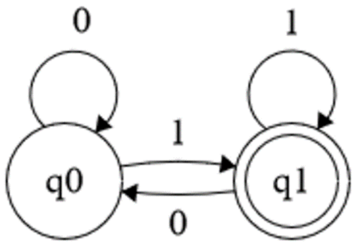
באופן כללי, המעבר יתבצע בצורה הבאה: $L(A) = \bigcup_{q \in F} L(q)$

ולכן נרצה לאפיין את השפה של כל מצב מקבל, ולבצע "איחוד" ביניהם.

בנוסף, באוטומטים יהיו מעגלים, ולכן נאפיין כיצד מבצעים את המעגלים האלו (דומה ברעיון להוכחה של $L_{k,k}^{k-1}$) בלי לעבור ב-q_0. כי אם חזרנו ל-q_0, חזרנו לרישא, אז המסלול הזה מיותר).

$$L[A] = (q_0 \to q_0)^* \left(q_0 \to q_f\right)\left(q_f \to q_f\right)^*$$

4.1 דוגמה 3



בנו ביטוי רגולרי לשפה: $L = \{w1 : w \in \{0,1\}^*\}$.

תחילה נבנה אוטומט:

ואז נבנה ממנו ביטוי רגולרי: נאפיין את המעגלים והמסלולים:

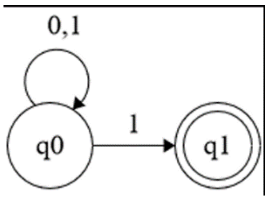
$$q_0 \to q_1 = 1, \qquad q_0 \to q_0 = 0 + 1^*0, \qquad q_1 \to q_1 = 1^*$$

(במעגל $q_1 \to q_1$, בלי לעבור ב-q_0).

ונרשום את השפה:

$$L = L[r] = (q_0 \to q_0)^*(q_0 \to q_1)(q_1 \to q_1)^* = (0^* + 1(1^*)0)^*1 \cdot (1^*)$$

4.2 דוגמה 3 שוב



בדוגמה הקודמת, היינו מגיעים לתשובה מהר יותר אם היינו בונים ב"ר מהאסל"ד של השפה:

כעת המעגלים פשוטים יותר: $q_0 \to q_0 = (0+1)^*$.

ולכן השפה היא $L = L[r] = (0+1)^*1$

וניתן להוכיח שקילות בין הביטוי מהדוגמה הקודמת לביטוי הנוכחי.

4.3 שיטת הבלוקים

לפעמים ציור האוטומט יהיה ארוך, והפקת הביטוי ממנו תהיה עוד יותר ארוכה כי יהיו הרבה מעגלים. לכן, יש שיטה נוספת להפקת ביטוי רגולרי, "שיטת הבלוקים". השיטה תעבוד על שפות בסגנון "כל המילים ללא תת מחרוזת x". "נפריד" את חלקי המילה לבלוקים של "מה כן מותר", ונראה מה קורה לפני הבלוק הראשון, בין הבלוקים, ולאחר הבלוק האחרון. ומשם נאפיין את השפה.

דוגמה: נבנה ב"ר לשפה $L = \{w \in \{a,b,c\}^* : w \text{ does not contain } 'abc'\}$.

נפריד בין כל שני a בעזרת בלוק, ונראה מה יכול להיות רשום בו: $\_\_a\_\_a\_\_a\_\_a\_\_$. בין כל שני a אסור שיהיה רשום bc. מה כן מותר? אם נהרוס את הרצף abc, הכל מותר. כלומר נראה c או bb, ואז באופן חופשי $(b+c)^*$. בנוסף, נראה b בודד או ϵ ואז נמשיך ל a הבאה.

כלומר, לאחר שראינו a ניתן לראות $(bb+c)(b+c)^* + b + \epsilon$. נסמן $r = (bb+c)(b+c)^* + b + \epsilon$ ונמשיך. אותו דבר אחרי ה-a האחרונה. לפני ה-a הראשונה אפשר לראות $(b+c)^*$ באופן חופשי. אפשר לראות את ar אינסוף פעמים, או אפס.
בסה"כ קיבלנו: $L = (b+c)^*(ar)^*$

4.4 תרגיל ממבחן

יהיו L_1, L_2 שפות רגולריות עם ביטויים רגולריים r_1, r_2 בהתאמה. נניח ש $L_1 \cap L_2 \neq \emptyset$. בנו ביטויים רגולריים לשפות:

$$L_3 = \{w_1 w_2 \dots w_n : \forall i \in [n]: w_i \in (L_1 \cup L_2) \wedge at\ most\ 4\ words\ are\ from\ L_2\}$$

באופן מגושם, נבנה כך שלכל היותר יש 4 מילים מ r_2:

$$L_3 = (r_1)^* r_2 (r_1)^* + (r_1)^* r_2 (r_1)^* r_2 (r_1)^* + (r_1)^* r_2 (r_1)^* r_2 (r_1)^* r_2 (r_1)^* + (r_1)^* r_2 (r_1)^* r_2 (r_1)^* r_2 (r_1)^* r_2 (r_1)^*$$

או באופן אלגנטי יותר:

$$L_3 = (r_1)^* (r_2 + \epsilon)(r_1)^* (r_2 + \epsilon)(r_1)^* (r_2 + \epsilon)(r_1)^* (r_2 + \epsilon)(r_1)^*$$

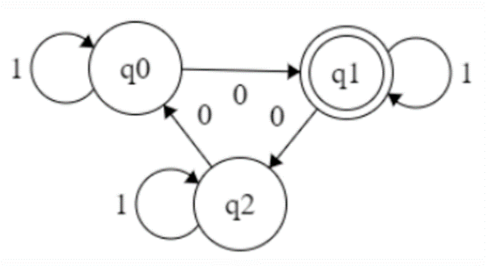
$$L_4 = \begin{Bmatrix} w_1 w_2 \dots w_n : \forall i \in [n]: w_i \in (L_1 \cup L_2) \\ \wedge\ no\ 3\ adjacent\ words\ are\ from\ L_1, L_2, L_2\ (in\ that\ order) \end{Bmatrix}$$

שיטת הבלוקים: יש בלוקים וביניהם r_1. בבלוק אמצעי, יכול להיות ϵ או r_2 בודד. כנ"ל באחרון. בבלוק הראשון זה $(r_2)^*$. כלומר:

$$L_4 = (r_2)^* \left(r_1 (r_2 + \epsilon)\right)^*$$

4.5 עוד דוגמה

בנו ב"ר לשפה: $L = \{w \in \{0,1\}^* : \#_0(w) \equiv 1 (mod\ 3)\}$. נבנה אוטומט:



ונבנה ממנו ב"ר:

$$L = L[A] = L[q_1] = (q_0 \to q_0)^* (q_0 \to q_1)(q_1 \to q_1)^*$$

$$(q_0 \to q_0) = 1 + 01^*01^*0, \qquad (q_0 \to q_1) = 0, \qquad (q_1 \to q_1) = 1$$

$$L = (1 + 01^*01^*0)^* 01^*$$

הרצאה 5

1 למת הניפוח לשפות רגולריות

הגדרה: קבוצת מצבים $C \subseteq Q$ תקרא **מעגל** אם: $\exists q \in C$ כך ש: $\hat{\delta}(q,w) = q$: $\exists w \in \Sigma^+$, (כלומר, אם קיים מצב q ב-C שעבורו קיימת מילה לא ריקה שאם נתחיל מ-q ונקרא את המילה, נחזור ל-q). ובנוסף, קריאת כל אות במילה משאיר את האוטומט במצב מ-C. פורמלית:

$$\{\forall u, v \neq w\ s.t\ uv = w :\ \hat{\delta}(q,u) \in C\}$$

1.1 בכל אס"ד קיים מעגל

הוכחה: אם קיימת ב-L מילה הארוכה ביותר, אזי קיים מצב בור עבור כל המילים שארוכות ממנה. בור הוא מעגל עבור כל מילה.

אם אין מילה ארוכה ביותר, אזי $L(A)$ אינסופית. היות ו- $|Q|$ סופי (כי זה **אס"ד**), בהכרח קיים מעגל עבור מילים שארוכות מ-$|Q|$ (שובך היונים): נב"ש שאין מעגלים. נבחר מילה שארוכה יותר מ-$|Q|$. אין מעגלים, כלומר בכל צעד הגענו למצב חדש. המצבים ייגמרו לפני שנסיים את המילה. הצעד הבא חייב להיות למצב שכבר היינו בו.

שפה היא רגולרית אמ"מ קיים אס"ד המקבל אותה. כל שפה סופית היא רגולרית. קיימות שפות לא רגולריות – ראינו הוכחה ישירה לשפה ספציפית. בהוכחה שראינו, השתמשנו בסופיות מספר מצבי האוטומט כדי לטעון שיש סוג של מחזוריות באופן הפעולה שלו. ננסה להכליל את הטיעון כדי לקבל תנאי הכרחי להיותה של שפה רגולרית:

1.2 למת הניפוח

תהי L שפה רגולרית. אזי, קיים $n \in \mathbb{Z}^+$ שעבורו לכל $z \in L$ המקיימת $|z| \geq n$, קיים פירוק $z = uvw$ המקיים:

א. $|uv| \leq n$
ב. $1 \leq |v|$
ג. $\forall i \in \mathbb{Z}^+ \cup \{0\} : uv^iw \in L$

כלומר: יש אורך מסוים, שכל מילה שארוכה ממנו מקיימת את התנאים.

למת הניפוח היא תנאי **הכרחי אך לא מספיק** עבור רגולריות. נראה בהמשך שפות שמקיימות את הלמה, אבל אינן רגולריות. כלומר, עיקר השימוש של הלמה יהיה כדי להפריך רגולריות של שפה.

1.3 הוכחת הלמה

תהי L שפה רגולרית, ויהי $A = (Q, \Sigma, q_0, \delta, F)$ אס"ד המקבל אותה. יהי $n = |Q|$.

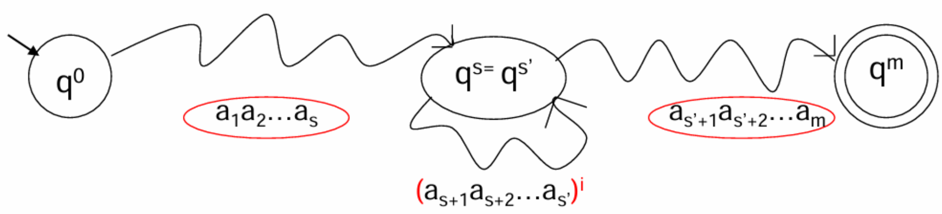
תהי $z = a_1a_2 \cdots a_m$ מילה כלשהי ב-L, כך ש $m \geq n$.

לכל $0 \leq i \leq m$ נסמן $q^i = \hat{\delta}(q_0, a_1a_2 \cdots a_i)$, כאשר $q^0 = q_0$. (המצב אחרי קריאת i תווים. $m+1$ יונים).

כיוון ש- $n = |Q|$ (שובכים), קיימים זוג אינדקסים s, s' המקיימים: $0 \leq s < s' \leq m$ כך ש: $q^s = q^{s'}$.

שימו לב ש $s \leq n$, כי כבר ברישא $a_1a_2 \cdots a_n$ קיימת חזרה על מצב פעמיים, (כי מילה באורך n צריכה $n+1$ מצבים, כי צריך מצב גם עבור המילה הריקה). ובחרנו את s, s' לפני החזרה הראשונה.

נתבונן בחישוב האוטומט על המילה $z = a_1a_2 \cdots a_m$:



מכיוון ש $z \in L$, נקבל ש $q^m \in F$. ונובע גם ש $a_1 a_2 \cdots a_s a_{s+1} \cdots a_m \in L$.

נוכיח באינדוקציה ש $a_1 a_2 \cdots a_s (a_{s+1} \cdots a_{s'})^i a_{s'+1} \cdots a_m \in L$, לכל $0 \le i$. נרשום:

$$z = \underbrace{a_1 a_2 \cdots a_s}_{u} \underbrace{a_{s+1} \cdots a_{s'}}_{v} \underbrace{a_{s'+1} \cdots a_m}_{w}$$

נראה כי שלושת תנאי הלמה מתקיימים:

א. $|uv| = s' \le n$ (לפי בחירת s, s')
ב. $1 \le |v|$ (כי $s < s'$)

נותר רק להוכיח את:

ג. $\forall i \in \mathbb{Z}^+ \cup \{0\} : uv^i w \in L$

עלינו להוכיח כי בפירוק הנ"ל, $\forall i \ge 0 : uv^i w \in L$, כאשר אנחנו יודעים ש $\hat{\delta}(q^s, v) = q^s$.

נוכיח ראשית כי לכל i מתקיים $\hat{\delta}(q^s, v^i) = q^s$, באינדוקציה על i.

בסיס: $\delta(q^s, \epsilon) = q^s,\ \ v^0 = \epsilon,\ \ i = 0$.

צעד: נניח ש $\hat{\delta}(q^s, v^i) = q^s$. מכאן נקבל: א – מהנ"א.

$$\hat{\delta}(q^s, v^{i+1}) = \hat{\delta}(q^s, v^i v) = \hat{\delta}(\hat{\delta}(q^s, v^i), v) \overset{\text{א}}{=} \hat{\delta}(q^s, v) = q^s$$

אז הוכחנו ש $\forall i \ge 0 : uv^i w \in L$.

כמו כן אנחנו יודעים ש $\hat{\delta}(q^s, w) = q^m \in F$, וכן $\hat{\delta}(q^0, u) = q^s$. נרשום: א – מהאינדוקציה שהוכחנו.

$$\hat{\delta}(q^0, uv^i w) = \hat{\delta}(\hat{\delta}(q^0, uv^i), w) = \hat{\delta}(\hat{\delta}(\hat{\delta}(q^0, u), v^i), w) = \hat{\delta}(\hat{\delta}(q^s, v^i), w) \overset{\text{א}}{=} \hat{\delta}(q^s, w) = q^m \in F$$

1.4 למת הניפוח בשפות סופיות

שימו לב, שלמת הניפוח לא מוגדרת רק עבור שפות אינסופיות. איך היא מתקיימת בשפות סופיות?

יהי t אורך המילה הארוכה ביותר בשפה. נבחר $n = t + 1$, ואכן מתקיים שכל מילה שאורכה לפחות n ניתנת לניפוח – אין מילים כאלה, אז הטענה מתקיימת באופן ריק.

1.5 הוכחת אי – רגולריות ע"י הלמה

כדי להוכיח ששפה אי – רגולרית:

נניח בשלילה שהיא רגולרית. ניקח את ה-n שקיומו מובטח בלמה. נבחר מילה z באורך $|z| \ge n$. נקבל פירוק כלשהו $z = uvw$. ("האויב" בוחר את הפירוק). מוצאים $i \ne 1$ שעבורו $uv^i w \notin L$, וכך מגיעים לסתירה.

למה תמיד $i \ne 1$? כי עבור $i = 1$, זה פשוט z. וברור שהיא שייכת לשפה.

2 דוגמאות לשימוש בלמת הניפוח

2.1 דוגמה 1

צ"ל שהשפה $L = \{x \in \{a,b\}^* : \#_a(x) = \#_b(x)\}$ אינה רגולרית.

נב"ש שהיא כן רגולרית. יהי n הקבוע שקיומו מובטח בלמת הניפוח. נבחר $z = a^n b^n$. בבירור $z \in L$, וגם $|z| \geq n$. יהי uvw פירוק של z כפי שמובטח בלמה. נבחר $i = 0$ ונגיע לסתירה: כיוון שמתקיים $|uv| \leq n$ וגם $|v| \leq 1$, נקבל:

$$z_0 = uv^0w = uw = a^{n-|v|}b^n \notin L$$

בסתירה ללמת הניפוח. כלומר, L אינה רגולרית.

שימו לב ש $a^{n-|v|}b^n \notin L$ בגלל שחרגנו מההגדרה על מספר ה-a, b ולא בגלל שחרגנו מהמבנה של $a^k b^k$.

2.2 דוגמה 2

צ"ל שהשפה $L = \{xx : x \in \{a,b\}^*\}$ אינה רגולרית.

נב"ש שהיא כן רגולרית. יהי n הקבוע שקיומו מובטח בלמת הניפוח. נבחר $z = a^n b a^n b$. יהיה קל לטפל במילה הזאת, כי ה-n התווים הראשונים זהים.

בבירור $z \in L$, וגם $|z| \geq n$. יהי uvw פירוק של z כפי שמובטח בלמה. בגלל ש ה-n התווים הראשונים זהים, כיוון שמתקיים $|uv| \leq n$, uv כולה מורכבת מ-a. וגם נתון ש $|v| \geq 1$.

נבחר $i = 0$ ונגיע לסתירה: נקבל:

$$z_0 = uv^0w = uw = a^{n-|v|}ba^nb \notin L$$

בסתירה ללמת הניפוח. כלומר, L אינה רגולרית.

2.3 דוגמה 3

צ"ל שהשפה $L = \{a^{k^2} : k \in \mathbb{N}\}$ אינה רגולרית.

נב"ש שהיא כן רגולרית. יהי n הקבוע שקיומו מובטח בלמת הניפוח. נבחר $z = a^{n^2}$. בבירור $z \in L$, וגם $|z| \geq n$. יהי uvw פירוק של z כפי שמובטח בלמה. נסמן $t = |v|$ (עבור $1 \leq t \leq n$ כלשהו).

אנחנו רוצים ש $uv^iw \notin L$. מתקיים: $uv^iw = a^{n^2+(i-1)|v|}$. נרצה למצוא i כך ש $n^2 + (i-1)|v|$ הוא לא ריבוע של אף מספר שלם. ננסה למצוא מספר בין הריבוע של n, לריבוע הבא. כלומר משהו שקטן מ: $(n+1)^2 = n^2 + 2n + 1$.

אם נוסיף ל-n^2 מספר שהוא בין 1 ל- $2n + 1$, זה יעבוד. נזכר ש $1 \leq |v| \leq n$, אז נביא למצב שמוסיפים את $|v|$.

נבחר $i = 2$ ונגיע לסתירה:

$$z_2 = uv^2w = uw = a^{n^2+t} \notin L$$

(כי $n^2 < n^2 + 1 \leq n^2 + t \leq n^2 + n < n^2 + 2n + 1 = (n+1)^2$, כלומר העלנו את a "בחזקת" מספר שהוא לא k^2 לאף k).

בסתירה ללמת הניפוח. כלומר, L אינה רגולרית.

3 שפה לא רגולרית הניתנת לניפוח

כאמור, למת הניפוח היא תנאי הכרחי אך לא מספיק לרגולריות. נוכיח את זה: נראה שפות לא רגולריות שכן ניתנות לניפוח.

לדוגמה: $L = \{a\}^* \cup \{b^j a^{k^2} : 1 \le j, k\}$.

3.1 קיום למת הניפוח

נראה שלמת הניפוח מתקיימת, עם $n = 1$. עבור כל $z \in L$ כך ש $|z| \ge 1$, נגדיר פירוק $z = uvw$ כך ש $u = \epsilon, |v| = 1$.

אם $z \in \{a\}^*$, אז לכל i גם $z_i = uv^i w \in \{a\}^* \subseteq L$.

אם z מהצורה $b^j a^{k^2}$ כאשר $j > 0$, אזי לכל i גם z_i מהצורה הזו. כלומר, למת הניפוח מתקיימת.

3.2 אי – רגולריות

נראה שהשפה לא רגולרית: נב"ש שהיא כן רגולרית. ניזכר שהשפה $L_1 = \{a\}^*$ רגולרית. אזי, מסגירות להפרש נקבל:

$L' = L \setminus L_1 = \{b^j a^{k^2} : 1 \le j, k\}$ רגולרית. מסגירות להיפוך נובע שגם $(L')^R = \{a^{k^2} b^j : 1 \le j, k\}$ רגולרית.

אבל ע"י אותה הוכחה מ-2.3, מתקבל כי $(L')^R$ לא ניתנת לניפוח. סתירה לכך שהיא רגולרית.

4 תרגילים ממבחנים

הוכיחו או הפריכו: השפות הבאות רגולריות:

א: $L_1 = \{a^i b^j : i \le j\}$. אינטואיטיבית, זה אוטומט שדורש לספור. אין אוטומט כזה. נוכיח שלא רגולרית:

נב"ש שכן רגולרית, כלומר למת הניפוח מתקיימת. יהי n הקבוע המובטח מהלמה. תהי $z = a^n b^n$. מתקיים $|z| = 2n \ge n$ ולכן קיים פירוק $z = uvw$ המקיים $|v| \ge 1, |uv| \le n$. נשים לב ש uv מורכבת רק מ-a.

עבור $i = 2$ נקבל שהמילה המנופחת היא: $uv^2 w = a^{n+(i-1)|v|} b^n = a^{n+2|v|} b^n$, לא בשפה.

ב: $L_2 = \{a^i b^j : i \neq j\}$

נב"ש שהיא רגולרית ויהי n הקבוע המובטח בלמה. תהי $z = a^n b^{n+n!}$. $z \in L$ וגם $|z| \ge n$. ולכן קיים פירוק $z = uvw$ כך ש:

$$|uv| \le n, \qquad |v| \ge 1, \qquad \forall i \in \mathbb{N}: uv^i w \in L$$

לכל i נקבל:

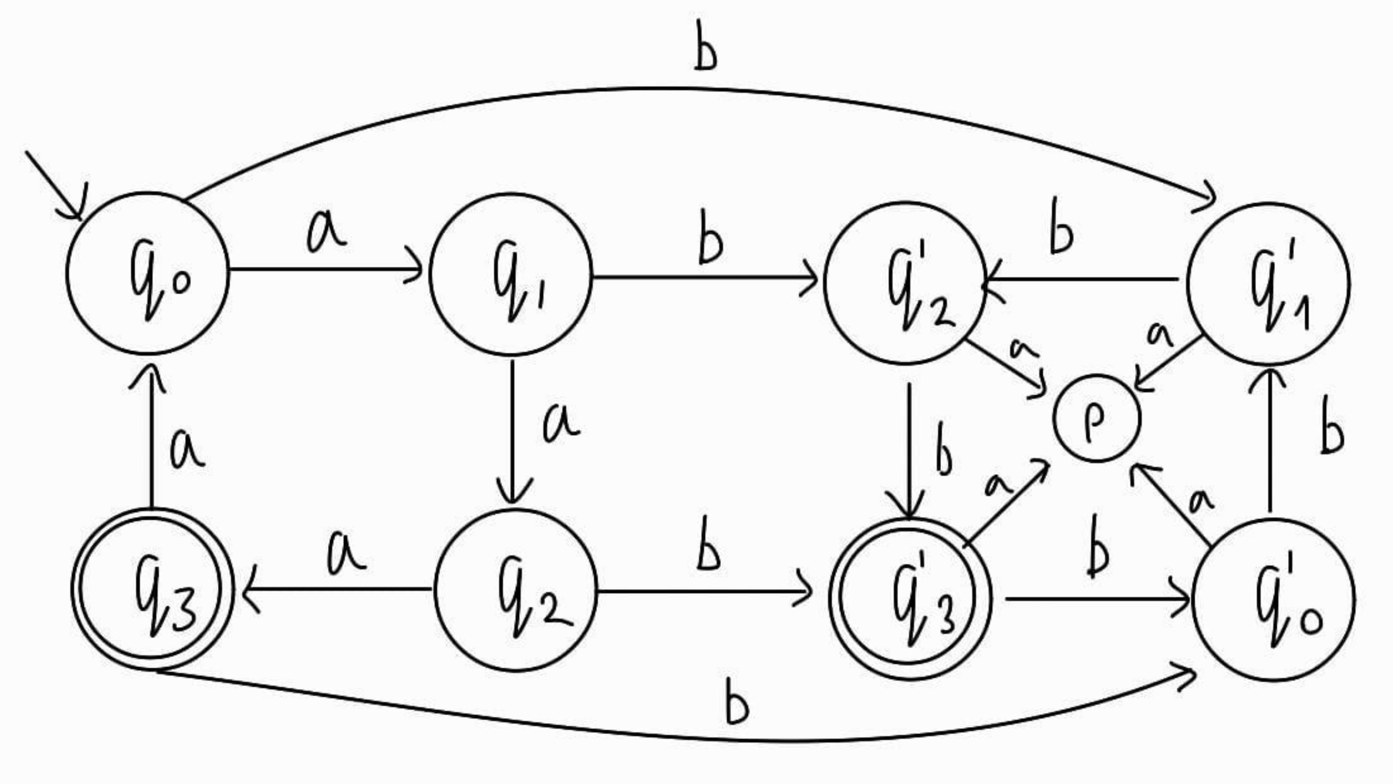
$$uv^i w = a^n a^{(i-1)\cdot|v|} b^{n+n!} = a^{n+(i-1)|v|} b^{n+n!}$$

מתי $n + (i-1)|v| = n + n!$?

$$(i-1)|v| = n! \to i|v| - |v| = n! \to i|v| = n! + |v| \to i = \frac{n!}{|v|} + 1$$

ג: $L = \{a^i b^j : i + j \equiv 3 (mod\ 4)\}$

בתור כלל אצבע, שפה עם מודולו תהיה רגולרית. נבנה אוטומט ונהפוך אותו לב"ר:



$$(q_0 \to q_0) = a^4, (q_0 \to q_3) = a^3, (q_3 \to q_3) = \epsilon, (q_0 \to q'_3) = b^3 + ab^2 + a^2b + a^3b^4, (q'_3 \to q'_3) = b^4$$

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

אפשר גם ישירות, בלי אוטומט:

$$(a^4)^*(a^3 + a^2b + ab^2 + b^3)(b^4)^*$$

בהתחלה ובסוף, כפולות של 4 לא משפיע על מודולו. ובאמצע, כל הדרכים להגיע לסכום שהוא 3 מוד 4.

ד: $L_4 = \{a^i b^j c^k : k^2 < j < 10i \wedge 0 \le i < k\}$

לכאורה, לא רגולרי כי צריך לספור. אבל נשים לב להגבלות:

אם $k^2 < j < 10i$ וגם $i < k$, זה אומר ש $k^2 < 10k$, כלומר $k < 10$. זה מגביל את i ולכן גם את j. $i < 9$, $j < 90$.

נכתוב מחדש את ההגדרה:

$$L_4 = \{a^i b^j c^k : k < 10, i < 9, j < 90\}$$

השפה סופית ולכן רגולרית.

ה: $L_5 = \{a^i b^j c^k : \min(i, j) \le k\}$

נב"ש שהיא רגולרית ויהי n הקבוע המובטח בלמה. אנחנו רוצים מילה שאם ננפח אותה, התנאי לא יתקיים. כלומר שהקטן מבין מספר ה-a, b יהיה גדול ממש ממספר ה-c.

תהי $z = a^n b^{2n} c^n$. $z \in L$ וגם $|z| \ge n$. ולכן קיים פירוק $z = uvw$ כך ש:

$$|uv| \le n, \qquad |v| \ge 1, \qquad \forall i \in \mathbb{N}: uv^i w \in L$$

כל ניפוח רק יגדיל את מספר ה-a, לדוגמה $i = 2$:

$$uv^2w = a^{n+|v|} b^{2n} c^n \notin L_5$$

ו: $L_6 = \{w \in \{0,1\}^* : \#_0(w) \equiv 1(mod\ 3) \wedge \#_1(w) \equiv 2(mod\ 3)\}$

אוטומט לכל אחד, ומכפלה.

תרגיל: הוכיחו/הפריכו:

א. אם L שפה שמקיימת את למת הניפוח עבור $n = 20$, אז היא מקיימת את למת הניפוח עבור $n = 17$.
ב. אם L שפה שמקיימת את למת הניפוח עבור $n = 17$, אז היא מקיימת את למת הניפוח עבור $n = 20$.

אינטואיציה:

אם כל מילה שארוכה מ20 מקיימת תנאי כלשהו, אז עדיין לא בהכרח כל מילה שארוכה מ17 מקיימת את התנאי. אז נמצא מילה בין 17-19 שלא מקיימת, וזה יפריך.

אם כל מילה שארוכה מ17 מקיימת תנאי, אז ברור שכל מילה שארוכה מ20 מקיימת.

פורמלית:

א. נפריך: נקח את השפה $L = \{w \in \Sigma^* : |w| \equiv 18\ (mod\ 20)\}$. לכל מילה ארוכה מ20, הפירוק: $u = \epsilon, |v| = 20$ מקיים את התנאי. אבל מילה באורך 17 לא תקיים, כי אין מעגל באורך 17.

תרגיל נוסף: תהי $L = \{w \in \{0,1\}^* : \#_0(w) \equiv \{2,3\}(mod\ 5)\}$

הראו כי השפה מקיימת את למת הניפוח. (בפרט, קבעו את ה-n המינימלי שעבורו הלמה מתקיימת).

נקבע $n = 5$. לכל $z \in L$ כך ש $|z| \geq n$ נקבע את הפירוק: $z = uvw$ כך שלכל $i \in \mathbb{N}$, $uv^iw \in L$.

אם 5 התווים הראשונים הם 0, נקבע $u = \epsilon, v = 0^5$. ואז כל ניפוח ישאיר את מספר האפסים שקול במוד 5.

אם ב5 התווים הראשונים יש 1, נקבע: u הוא מספר האפסים עד ה1 הראשון. $v = 1$. ואז כל ניפוח לא משפיע על מספר האפסים.

הרצאה 6

אפיון אלגברי של שפות רגולריות

מוטיבציה: ראינו את למת הניפוח לשפות רגולריות. ראינו גם שקיימות שפות לא רגולריות שניתנות לניפוח. אפשר להוכיח גרסה מוכללת של למת הניפוח, שבה ניתן לטפל בחלק מהשפות האלו. אבל זה עדיין תנאי **הכרחי אך לא מספיק** לרגולריות. נרצה לפתח אפיון מלא, כלומר, "שפה רגולרית אמ"מ מתקיים ___".

1 יחסים

תזכורת מתורת הקבוצות: יחס ייקרא:

- **רפלקסיבי** אם $\forall x : R(x,x)$,
- **סימטרי** אם $\forall x,y : R(x,y) \Longrightarrow R(y,x)$,
- **טרנזיטיבי** אם $\forall x,y,z : R(x,y) \wedge R(y,z) \Longrightarrow R(x,z)$

יחס R המקיים את שלושת התכונות לעיל ייקרא **יחס שקילות.**

יחס שקילות משרה חלוקה של העולם לתתי קבוצות זרות ומשלימות הנקראות **מחלקות שקילות.**

לדוגמה – פעולת mod היא יחס שקילות, ומגדירה מחלקות שקילות על הטבעיים.

יהי R יחס שקילות על קבוצה A. נסמן $index(R)$ את מספר מחלקות השקילות.

נציג מתוך המחלקות יסומן: $[x] = \{y : R(x,y)\}$.

נאמר כי יחס R' **מעדן את** R אם $R' \subseteq R$, כלומר:

$$\forall x,y \in A : R'(x,y) \Longrightarrow R(x,y)$$

כל מחלקת שקילות של R' מוכלת במחלקת שקילות של R.

1.1 יחסי שקילות מעל Σ^*

הגדרה: יחס שקילות R מעל Σ^* ייקרא **אינווריאנטי מימין** אמ"מ הוא מקיים:

$$\forall x,y \in \Sigma^* : R(x,y) \Longrightarrow \forall z \in \Sigma^* : R(xz,yz)$$

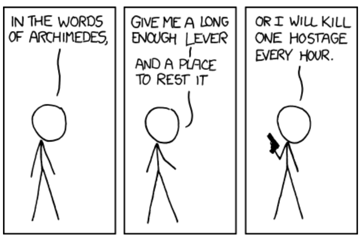
כלומר, אם נשרשר את אותה המילה מימין, היחס יישמר. לא בהכרח הם יישארו באותה מחלקת שקילות – יכול להיות ששניהם יעברו למחלקת שקילות אחרת (אבל שניהם יהיו באותה מחלקת שקילות).

דוגמאות ליחסים כאלו:

1. האות הימנית ביותר ב-x שווה לאות הימנית ביותר ב-y.
2. $|x| = |y|$

היחס $x = y^R$ אינו יחס שקילות אינווריאנטי מימין.

גם שני ילדים שקולים על מאזניים אינם אינווריאנטים מימין, כי בתוספת משקל זהה לצד ימין של שניהם, אחת התוספות רחוקה יותר מהמרכז ומשפיעה יותר, בגלל עיקרון המנוף:
וגם, מסיבה טכנית, זה לא מוגדר מעל Σ^*.



איור 1 - עקרון המנוף של ארכימדס

2 הקישור לאוטומט – יחס R_A

בהינתן $A = (Q, \Sigma, q_0, \delta, F)$, נגדיר יחס שקילות R_A:

$$R_A(x, y) \Leftrightarrow \hat{\delta}(q_0, x) = \hat{\delta}(q_0, y)$$

כלומר, שתי מילים שקולות אמ"מ הן מובילות אותנו לאותו מצב.

נוכיח תחילה שזה אכן יחס שקילות – נוכיח שהוא רפלקסיבי, סימטרי, וטרנזיטיבי:

- רפלקסיבי: $\forall x \in \Sigma^* :\ \hat{\delta}(q_0, x) = \hat{\delta}(q_0, x)$
- סימטרי: $\forall x, y \in \Sigma^* : R_A(x, y) \Longrightarrow \hat{\delta}(q_0, x) = \hat{\delta}(q_0, y) \Longrightarrow R_A(y, x)$
- טרנזיטיבי:

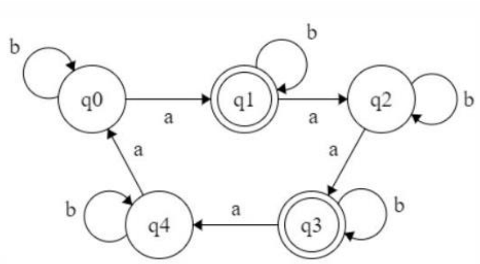
$$\forall x, y, z \in \Sigma^* : R_A(x, y) \wedge R_A(y, z) \Longrightarrow$$
$$\left[\hat{\delta}(q_0, x) = \hat{\delta}(q_0, y)\right] \wedge \left[\hat{\delta}(q_0, y) = \hat{\delta}(q_0, z)\right] \Longrightarrow$$
$$[\hat{\delta}(q_0, x) = \hat{\delta}(q_0, z)]$$

נניח כי $Q = \{q_0, q_1, \dots, q_{m-1}\}$, ונגדיר את השפות הבאות:

$$S_i = \{x : \hat{\delta}(q_0, x) = q_i\}$$

מחלקות השקילות של R_A הן ה-S_i-ים שאינן ריקות. כלומר: באס"ד שבו כל המצבים ברי השגה, ל-R_A יש $|Q|$ מחלקות שקילות.

דוגמה:



$$S_0 = \left(b + a(b^*)a(b^*)a(b^*)a(b^*)a(b^*)\right)^*$$
$$S_1 = S_0 \cdot a(b^*)$$
$$S_2 = S_1 \cdot (b^*)a(b^*)$$
$$S_3 = S_2 \cdot (b^*)a(b^*)$$
$$S_4 = S_3 \cdot (b^*)a(b^*)$$

2.1 שפת האוטומט

כעת שהגדרנו את מחלקות השקילות בתור שפות מילים המגיעות למצבים, נוכל להיעזר בכך כדי להגדיר את שפת האוטומט:

$$L(A) = \cup_{q_i \in F} S_i = \{x : \exists q \in F : \hat{\delta}(q_0, x) = q\}$$

בדוגמה לעיל, המצבים המקבלים הם q_1, q_3 ולכן שפת האוטומט היא:

$$L(A) = L_1 \cup L_3$$
$$r = (b + a(b^*)a(b^*)a(b^*)a(b^*)a)^*$$
$$S_1 = r \cdot a(b^*)$$
$$S_3 = r \cdot a(b^*)a(b^*)a(b^*)$$

2.2 R_A אינווריאנטי מימין

תזכורת: יחס שקילות R מעל Σ^* ייקרא **אינווריאנטי מימין** אמ"מ הוא מקיים:

$$\forall x, y \in \Sigma^* : R(x, y) \Longrightarrow \forall z \in \Sigma^* : R(xz, yz)$$

כלומר, אם נשרשר את אותה המילה מימין, היחס יישמר.

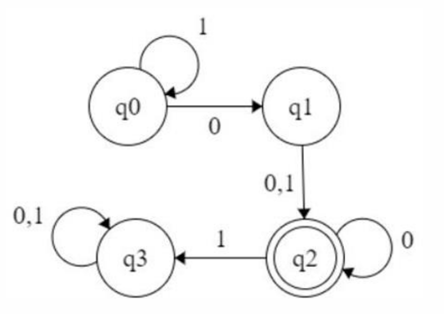
לפי הגדרה, R_A אומר ש $\hat{\delta}(q_0,x)=\hat{\delta}(q_0,y)$. נשרשר את z לשתי המילים:

$$\hat{\delta}(q_0,xz)=\hat{\delta}(\hat{\delta}(q_0,x),z)=\hat{\delta}(\hat{\delta}(q_0,y),z)=\hat{\delta}(q_0,yz)$$

כלומר, $R_A(xz,yz)$, כלומר R_A אינווריאנטי מימין.

2.3 עוד דוגמה

נתון האוטומט הבא:



מחלקות השקילות שלו הן:

$$S_0=1^*,\qquad S_1=1^*0,\qquad S_2=1^*0(0+1)0^*,$$
$$S_3=1^*0(0+1)0^*1(0+1)^*$$

$$L(A)=S_2$$

3 היחס R_L

הגדרה: עבור שפה $L\subseteq\Sigma^*$, היחס R_L מוגדר כך:

$$R_L(x,y)\Leftrightarrow(\forall z\in\Sigma^*:xz\in L\leftrightarrow yz\in L)$$

במילים: היחס R_L מקבל זוג מילים אמ"מ לכל סיפא z, שתי המילים המורחבות xz,yz שתיהן מתקבלות או שתיהן לא. במילים אחרות: x,y שקולות ביחס R_L אמ"מ לא קיימת סיפא המפרידה ביניהן.

בהמשך נראה שזה יחס שקילות.

3.1 דוגמאות

תהי $L_1=\Sigma^*$, מהן מחלקות השקילות של R_L?

ליחס זה יש מחלקת שקילות אחת בלבד, שכן לכל $z\in\Sigma^*$ מתקיים $xz\in L_1$ וגם $yz\in L_1$.

כנ"ל לגבי $L_2=\emptyset$ - לכל $z\in\Sigma^*$ מתקיים $xz\notin L_1$ וגם $yz\notin L_1$.

תהי $L_3=\{w\in\{0,1\}^*:|w|\equiv 1(mod\ 2)\}$. מהן מחלקות השקילות של R_L?

יש 2 מחלקות שקילות – זוגי, אי זוגי. לפי הגדרה: $R_L(x,y)\Leftrightarrow|x|\equiv|y|(mod\ 2)$, ולכן:

$$S_1=\{w\in\{0,1\}^*:|w|\equiv 0(mod\ 2)\}$$

$$S_2=\{w\in\{0,1\}^*:|w|\equiv 1(mod\ 2)\}$$

תהי $L_4=\{a,aa\}$. מהן מחלקות השקילות של R_L?

$$S_1=\{\epsilon\},\qquad S_2=\{a\},\qquad S_3=\{aa\},\qquad S_4=\Sigma^*\setminus\cup_{i=1}^3 S_i$$

תהי $L_5=\{x\in\{a,b\}^*:\#_a(x)=\#_b(x)\}$. מהן מחלקות השקילות של R_L?

לכל $k \in \mathbb{Z}$ יש מחלקת שקילות מהצורה: $S_k = \{x \in \{a,b\}^* : \#_a(x) - \#_b(x) = k\}$. אינטואיטיבית, כי סופרים את ההפרש בין a ל-b בכל מילה.

כדי להוכיח שאלה מחלקות השקילות, צריך להוכיח את הדברים הבאים:

1. שהן לא ריקות, ומכסות את Σ^*
2. שכל זוג מילים x, y מאותה מחלקה אינן ניתנות להפרדה ע"י אותה סיפא. פורמלית:
$\forall x, y \in S_j, \forall z \in S_i : xz \in S_{i+j} \wedge yz \in S_{i+j}$
3. כל זוג מילים ממחלקות שונות, כן ניתנות להפרדה ע"י סיפא כלשהי.

נראה את 3: יהיו $x \in S_i, y \in S_j$ עבור $i \neq j$ כלשהם. אזי, עבור: $z \in S_{-i}$, נקבל: $x \in L_5, y \notin L_5$.

3.2 עידון של שפה

יהי R יחס שקילות מעל Σ^*, ותהי $L \subseteq \Sigma^*$ שפה.

נאמר כי R **מעדן את** L אמ"מ מתקיים: $R(x,y) \Longrightarrow (x \in L \leftrightarrow y \in L)$.

כלומר, אם מתקיים ש: שתי מילים יהיו ביחס רק אם שתיהן בשפה או שתיהן לא בשפה. שקול ל: $(x \in L \; xnor \; y \in L)$.

טענה: לכל אס"ד A, היחס R_A מעדן את $L(A)$. (תזכורת: $R_A(x,y)$ אמ"מ שתי המילים מגיעות לאותו מצב).

הוכחה:

$$R(x,y) \Longrightarrow^{\text{א}} \left[\hat{\delta}(q_0,x) = \hat{\delta}(q_0,y)\right] \Longrightarrow^{\text{ב}} [x \in L(A) \leftrightarrow y \in L(A)]$$

א – הגדרת R_A, ב – הגדרת $L(A)$.

3.3 משפט האפיון של R_L

משפט: $\forall L \subseteq \Sigma^*$:

1. R_L הוא יחס שקילות אינווריאנטי מימין,
2. R_L מעדן את L,
3. אם R יחס שקילות אינווריאנטי מימין המעדן את L, אזי R מעדן גם את R_L.

במילים אחרות: R_L הינו יחס השקילות האינווריאנטי מימין המכיל הכי מעט מחלקות שקילות. כלומר, החלוקה ה"גסה" ביותר של Σ^* השומרת על העידון והאינווריאנטיות.

הוכחת 1: נוכיח שהוא יחס שקילות:

<u>רפלקסיבי</u>: נתון $\forall z \in \Sigma^* : [xz \in L \leftrightarrow xz \in L]$, ולכן לפי הגדרה: $R_L(x,x)$.

<u>סימטרי</u>: אם $R_L(x,y)$, זה אומר ש $xz \in L \leftrightarrow yz \in L$, כלומר $yz \in L \leftrightarrow xz \in L$, כלומר $R_L(y,x)$.

<u>טרנזיטיבי</u>: אם $R_L(x,y)$ וגם $R_L(y,w)$, אזי לכל $z \in \Sigma^*$ מתקיים: $[xz \in L \leftrightarrow yz \in L] \wedge [yz \in L \leftrightarrow wz \in L]$, ולכן גם מתקיים: $xz \in L \leftrightarrow wz \in L$. כלומר $R_L(x,w)$.

נוכיח שהוא אינווריאנטי מימין:

לפי הגדרה, אם $R_L(x,y)$ אזי $\forall z, w \in \Sigma^*$ מתקיים: $xzw \in L \leftrightarrow yzw \in L$.

לכן נוכל גם לכתוב: $\forall z \in \Sigma^*$ מתקיים: $(xz)w \in L \leftrightarrow (yz)w \in L$.

כלומר, $\forall z \in \Sigma^*$, מתקיים $R_L(xz, yz)$.

הוכחת 2: נוכיח ש R_L מעדן את L:

לפי הגדרה, $R_L(x,y) \Longrightarrow [\forall z \in \Sigma^* : xz \in L \leftrightarrow yz \in L]$.

בפרט, עבור $z = \epsilon$ נקבל $R_L(x,y) \Longrightarrow [x \in L \leftrightarrow y \in L]$.

כלומר, R_L מעדן את L.

הוכחת 3: כל יחס עם אותן תכונות מעדן את R_L:

יהי R יחס שקילות אינווריאנטי מימין המעדן את L. אזי:

$$R(x,y) \Longrightarrow^{\text{א}} [\forall z \in \Sigma^* : R(xz,yz)] \Longrightarrow^{\text{ב}} [\forall z \in \Sigma^* : (xz \in L \leftrightarrow yz \in L)] \Longrightarrow R_L(x,y)$$

א – אינווריאנטי מימין, ב – מעדן.

כלומר: $R(x,y) \Longrightarrow R_L(x,y)$, או במילים אחרות: $R \subseteq R_L$.

4 משפט נרוד

תהי $L \subseteq \Sigma^*$. אזי, הטענות הבאות שקולות:

1. L רגולרית,
2. קיים יחס שקילות R, שהוא אינווריאנטי מימין ומעדן את L ומקיים: $index(R) < \infty$,
3. $index(R_L) < \infty$.

תזכורת: עבור יחס שקילות R על קבוצה A. נסמן $index(R)$ את מספר מחלקות השקילות.

4.1 הוכחת $1 \rightarrow 2$

נתון: L רגולרית. צ"ל: קיים יחס שקילות R, שהוא אינווריאנטי מימין ומעדן את L ומקיים: $index(R) < \infty$.

אם L רגולרית, קיים עבורה אס"ד. כבר הראנו ש R_A הוא יחס שקילות אינווריאנטי מימין המעדן את L.

כמו כן, $index(R_A) \leq |Q_A|$ ובפרט $index(R_A) < \infty$, כנדרש.

4.2 הוכחת $2 \rightarrow 3$

נתון: קיים יחס שקילות R, שהוא אינווריאנטי מימין ומעדן את L ומקיים: $index(R) < \infty$. צ"ל: $index(R_L) < \infty$.

לפי משפט האפיון: כל יחס שקילות אינווריאנטי מימין המעדן את L מעדן גם את R_L. לכן מתקיים:

$$index(R_L) \leq index(R) < \infty$$

4.3 הוכחת $3 \rightarrow 1$

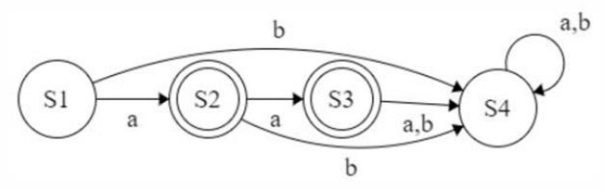
נתון: $index(R_L) < \infty$. צ"ל: L רגולרית.

הרעיון: נבנה אוטומט שמצביו הם מחלקות השקילות של R_L.

לדוגמה, עבור $L = \{a, aa\}$ מצאנו ארבע מחלקות שקילות:

$$S_1 = \{\epsilon\}, \qquad S_2 = \{a\}, \qquad S_3 = \{aa\}, \qquad S_4 = \Sigma^* \setminus \cup_{i=1}^{3} S_i$$

נבנה אס"ד ע"פ R_L:



נבנה אס"ד A_L המקבל את L.

קבוצת המצבים מוגדרת ע"י: $Q_{A_L} = \{[x] \,:\, x \in \Sigma^*\}$ כלומר, כל מחלקת שקילות היא מצב.

נתון $index(R_L) < \infty$ ולכן $|Q_{A_L}| < \infty$.

המצב ההתחלתי הוא $q_0 = [\epsilon]$.

פונקציית המעברים: $\delta([x], a) = [xa]$.

היות והיחס אינווריאנטי מימין, לא משנה איזה נציג נבחר מהמחלקה. שכן $[x] = [y] \Longrightarrow [xa] = [ya]$.

קבוצת המצבים המקבלים: $F = \{[x] \,:\, x \in L\}$.

4.4 $\boldsymbol{L = L(A)}$

נותר להראות שהאוטומט שבנינו אכן מתאים לשפה:

טענה 1: לכל x מתקיים: $\hat{\delta}(q_0, x) = [x]$.

הוכחה באינדוקציה על $|x|$:

בסיס: עבור $|x| = 0$, בהכרח $x = \epsilon$ ואכן $\delta(q_0, \epsilon) = q_0 = [\epsilon]$. (כך הגדרנו את q_0).

נניח שהטענה מתקיימת עבור $|w| = n$ ונוכיח עבור $w\sigma$.

$$\hat{\delta}(q_0, w\sigma) =^{א} \delta\left(\hat{\delta}(q_0, w), \sigma\right) =^{ב} \delta([w], \sigma) =^{ג} [w\sigma]$$

א – הגדרת $\hat{\delta}$, ב – הנ"א, ג – הגדרת δ.

טענה 2: לכל $x \in \Sigma^*$ מתקיים: $[x] \in F \iff x \in L$

כיוון ראשון: מיידי, מהגדרת F.

כיוון שני: נניח ש $[x] \in F$. מהגדרת F, קיימת מילה $y \in [x]$ כך ש $y \in L$ (המחלקות לא ריקות).

היות ו- $R_L(x, y)$, ו- R_L מעדן את L, נובע גם ש $x \in L$.

לכן קיבלנו סה"כ:

$$x \in L(A_L) \Longleftrightarrow^{א} \hat{\delta}(q_0, x) \in F \Longleftrightarrow^{ב} [x] \in F \Longleftrightarrow^{ג} x \in L$$

א – הגדרת F, ב – טענה 1, ג – טענה 2.

4.5 שימושי משפט נרוד

1. אפיון אמ"מ לשפות רגולריות: **שפה L רגולרית אמ"מ** $\infty > index(R_L)$
2. מינימליות אס"ד: כמות מחלקות השקילות של R_L יוצרות את האס"ד המינימלי לשפה.
3. שימושי לשאלות במבחן :)

5 דוגמאות לשימוש

5.1 הוכיחו כי: $L = \{b^*\} \cup \{a^k b^{2^n} : n, k \in \mathbb{N}\}$ אינה רגולרית.

למת הניפוח לשפות רגולריות לא עוזרת פה, כי "האויב" יכול לכפות ניפוח ב-a, ולא להפר את b^{2^i}.

לפי משפט נרוד, מספיק להראות ש $index(R_L) = \infty$.

איך מראים שיש אינסוף מחלקות שקילות? ע"י בחירת קבוצת מילים אינסופית, והוכחה כי כל שתי מילים ניתנות להפרדה ע"י סיפא כלשהי. מכאן ינבע כי כל שתי מילים שייכות במחלקות שקילות נפרדות, ומכאן שיש אינסוף מחלקות שקילות, כנדרש.

נגדיר את קבוצת המילים האינסופית $A = \{ab^{2^k} : k \in \mathbb{N}\}$.

יהיו $i \neq j$, ונסתכל על שתי המילים ab^{2^i}, ab^{2^j}. למילים אלו קיימת סיפא מפרידה: $z = b^{2^j}$.

$$ab^{2^i}b^{2^j} = ab^{2^i+2^j} \notin L$$

$$ab^{2^j}b^{2^j} = ab^{2^{j+1}} \in L$$

כלומר, לכל שתי מילים קיימת סיפא מפרידה, ולכן ליחס R_L יש אינסוף מחלקות שקילות.

לכן, לפי משפט נרוד, השפה L אינה רגולרית.

5.2 תהי $L = \{a^n c^n\} \cup \{a^n b^n c^n\}$. הוכיחו כי השפה אינה רגולרית, בעזרת משפט נרוד.

נגדיר את קבוצת המילים האינסופית $A = \{a^n\}$. יהיו $i \neq j$. נסתכל על שתי המילים a^i, a^j. למילים אלו קיימת סיפא מפרידה $z = c^i$: $a^i c^i \in L,\ a^j c^i \notin L$. מצאנו סיפא מפרידה בין כל שתי מילים מהקבוצה האינסופית, ולכן ליחס R_L קיימות אינסוף מחלקות שקילות. לכן, לפי משפט נרוד, השפה L אינה רגולרית.

5.3 תהי $L = \{a^n b^m c^m\} \cup \{a^n b^n c^m\}$. הוכיחו כי השפה אינה רגולרית, בעזרת משפט נרוד.

נגדיר את קבוצת המילים האינסופית $A = \{a^n\}$. יהיו $i \neq j$. נסתכל על שתי המילים a^i, a^j. שתיהן בשפה.

למילים אלו קיימת סיפא מפרידה: $z = b^i c^{i+1}$. כי $a^i b^i c^{i+1} \in L, a^j b^i c^{i+1} \notin L$.

מצאנו סיפא מפרידה בין כל שתי מילים מהקבוצה האינסופית, ולכן ליחס R_L קיימות אינסוף מחלקות שקילות. לכן, לפי משפט נרוד, השפה L אינה רגולרית.

5.4 תהי $L = \{a^n b^{3n} c^n\} \cup \{a^n b^{3n} c^{3n}\}$. הוכיחו כי השפה אינה רגולרית, בעזרת משפט נרוד.

נגדיר את קבוצת המילים האינסופית $A = \{a^n b^{3n}\}$. יהיו $i < j$ (בה"כ). נסתכל על שתי המילים $a^i b^{3i}, a^j b^{3j}$. למילים אלו קיימת סיפא מפרידה $z = c^i$. כי $a^i b^{3i} c^i \in L, a^j b^{3j} c^i \notin L$ (כי $i < j$ אז לא יכול להיות ש $i = 3j$).

מצאנו סיפא מפרידה בין כל שתי מילים מהקבוצה האינסופית, ולכן ליחס R_L קיימות אינסוף מחלקות שקילות. לכן, לפי משפט נרוד, השפה L אינה רגולרית.

הרצאה 7

1 תזכורות מהרצאה 6

יחס שקילות ייקרא אינווריאנטי מימין אם לכל $z \in \Sigma^*$ מתקיים: $R(x,y) \Leftrightarrow R(xz,yz)$.

עבור שפה $L \subseteq \Sigma^*$, היחס R_L מוגדר: $R_L(x,y) \Leftrightarrow \forall z \in \Sigma^* : xz \in L \leftrightarrow yz \in L$.

משפט האפיון של R_L: לכל $L \subseteq \Sigma^*$ מתקיים:

R_L אינווריאנטי מימין, ומעדן את L. ואם יחס R אינווריאנטי מימין ומעדן את L, הוא מעדן את R_L.

משפט נרוד: שלושת התנאים הבאים שקולים:

- השפה L רגולרית,
- קיים יחס שקילות R שהוא אינווריאנטי מימין ומעדן את L, וגם $index(R) < \infty$,
- $index(R_L) < \infty$.

1.1 שימוש במשפט נרוד, כדי להראות ששפה אינה רגולרית

נגדיר קבוצת מילים אינסופית, ונראה שלכל שתי מילים בקבוצה ניתן למצוא סיפא מפרידה.

זה יראה שכל שתי מילים יושבות במחלקת שקילות נפרדת של R_L. ולכן צריכות להיות אינסוף מחלקות שקילות.

לפי משפט נרוד, נקבל שהשפה לא רגולרית.

האם זה תקף תמיד? האם בכל שפה שבה נוכל למצוא אינסוף זוגות מילים עם סיפא מפרידה, זה מראה שהשפה לא רגולרית?

נשקול את השפה: $L = \{w \in \Sigma^* : |w| \equiv 0 (mod\ 2)\}$. מילים באורך זוגי.

קיימות רק שתי מחלקות שקילות. אבל לכל מילה, אפשר למצוא אינסוף מילים שלא שקולות לה.

מה ההבדל?

בדוגמה הראשונה הראנו באופן כללי על שתי מילים, וזה מוכיח ש**כל** שתי מילים בקבוצה יהיו במחלקת שקילות נפרדת.

בדוגמה השנייה קבענו את אחת המילים. לא הראנו שזה עובד לכל שתי מילים שנבחר.

2 מינימליזציה של אס"ד

כיצד נבנה אוטומט עם מספר מינימלי של מצבים?

ניעזר בטענה, שההוכחה שלה מסתמכת על כמה נקודות שראינו בהוכחה של משפט נרוד:

2.1 טענה:

אם A אס"ד המקבל את L, אזי $index(R_L) \leq |Q_A|$ הוא מספר מצבי האוטומט A_L (האוטומט שראינו בהוכחת משפט נירוד).

הוכחה: לפי ההגדרה, $L = L(A)$ ולכן R_A מעדן את L. ממשפט האפיון של R_L מתקיים כי R_A מעדן גם את R_L. ולכן $index(R_L) \leq index(R_A)$. מתקיים $|Q_{A_L}| = index(R_L) \leq index(R_A) \leq |Q_A|$, ובסה"כ: $|Q_{A_L}| \leq |Q_A|$, כנדרש.

קיבלנו כי R_L הינה עם כמות מחלקות שקילות מינימלית, ו-A_L הוא האוטומט בעל כמות המצבים המינימלית.

2.2 אלגוריתם לבנייה

1. נמצא את כל מחלקות השקילות של השפה.
2. בניית A_L:
   2.1. לכל מחלקת שקילות, נקצה מצב באוטומט.
   2.2. נסדר את δ לפי המחלקות.

בצורה זו, נקבל אוטומט בעל כמות מצבים מינימלית.

הבעיה: אי אפשר לתכנת את זה. מחשב (עדיין) לא יודע לפענח מהן מחלקות השקילות.

לכן, נציג את האלגוריתם של *Hopcroft*, שאותו אפשר לתכנת: סדר הפעולות שלנו: נציג את האלגוריתם, אינטואיציה למה הוא נכון, ודוגמת הרצה.

2.3 אלגוריתם *Hopcroft*

קלט: אס"ד $A = (Q', \Sigma, q_0', \delta', F')$ כלשהו.

פלט: אס"ד לאותה שפה, עם כמות מינימלית של מצבים. (יש רק אחד כזה, עד כדי איזומורפיזם).

השלבים:

1. נמחק את המצבים שאינם ישיגים מ-q_0'. נסמן ב A את האוטומט המתקבל. (מציאת המצבים ע"י *BFS*).
2. נגדיר יחס: $R \leftarrow \phi$. זהו יחס סימטרי. (בינארי על Q. כלומר הוא מתייחס לזוגות – זוג מסויים מקיים או לא מקיים את היחס).
   2.1. נעדכן: $R \leftarrow \{\{p,q\} : q \in F \wedge p \notin F\}$. כל הזוגות שאחד מהם מקבל והשני לא.
   2.2. שלב S:
      2.2.1. נחשב $S \leftarrow \{\{p,q\} \notin R : \exists \sigma \in \Sigma : \{\delta(q,\sigma), \delta(p,\sigma)\} \in R\}$.
      נבדוק זוגות שעוד לא ב-R: (כלומר כרגע, שניהם מתקבלים או שניהם לא). נבדוק אם יש תו בא"ב שמפריד ביניהם.
      2.2.2. אם $S \neq \phi$, נעדכן את $R \leftarrow R \cup S$ ונחזור ל 2.2.1. כאשר S ריקה, אין זוגות חדשים, נעבור לשלב הבא:
3. אם $\bar{R}$ (המשלים) אינו יחס שקילות על Q, נחזיר *FAIL*. (זה לא אמור לקרות, אבל זה *failsafe* בשביל נכונות האלגוריתם).
4. נבנה *DFA* מצומצם $\tilde{A} = (\tilde{Q}, \Sigma, \widetilde{q_0}, \tilde{\delta}, \tilde{F})$ באופן הבא:
   $\tilde{Q}$ = קבוצת מחלקות השקילות של $\bar{R}$.
   $\widetilde{q_0}$ = מחלקת השקילות של q_0 ב $\bar{R}$. נסמן ב $[q_0]_{\bar{R}}$.
   $\tilde{F} = \{\tilde{q} : \tilde{q} \cap F \neq \phi\}$.
   $\forall \tilde{q} \in \tilde{Q}, \forall \sigma \in \Sigma : \tilde{\delta}(\tilde{q}, \sigma) = [\delta(q,\sigma)]_{\bar{R}}$

2.4 אינטואיציה לאלגוריתם

בהינתן A (ללא מצבים לא ישיגים), מתקיים:

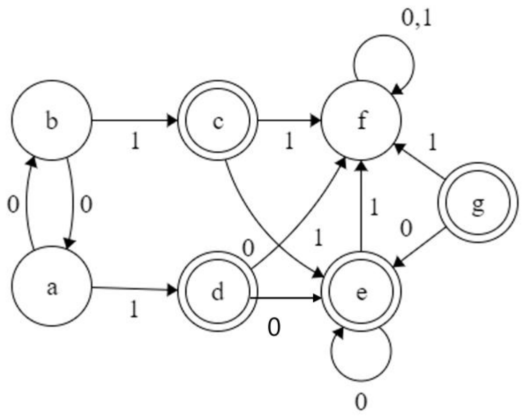
קבוצת מחלקות השקילות של R_A הן בדיוק $S = \{L_q : q \in R\}$.

R_A מעדן את R_L, ו-R_L מעדן את L.

האלגוריתם יקבל את המצבים p כך שה L_q-ים הם תתי קבוצות של אותה מחלקת שקילות ב-R_L יחד, ובין כל השאר היחס R יפריד.

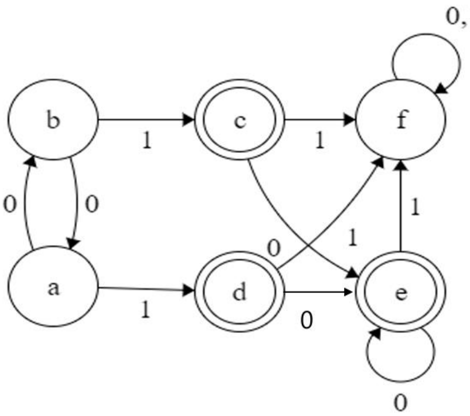
2.5 דוגמת הרצה 1

נתון האוטומט הבא: (שימו לב לחיצים שיוצאים מ- c, d – התחתון שיוצא מ-c זה 0, העליון שיוצא מ-d זה 1).



שלב 1: סילוק מצבים לא ישיגים.

כאן, המצב היחיד שלא ישיג הוא g. נוציא אותו ונקבל את האוטומט הבא:



שלב 2: נאתחל יחס $R = \phi$.

נכניס את כל הזוגות שאחד מהם ב-F והשני לא:

$$R = \{\{a,c\},\{b,c\},\{f,c\},\{a,d\},\{b,d\},\{f,d\},\{a,e\},\{b,e\},\{f,e\}\}$$

שלב 2.2.1: נחשב את S, קבוצת כל הזוגות שאינם ב-R, אך שעל ידי קריאת אות כן יהיו בו.

$\{a,f\}$ ייכנס ל-S כי האות 1 מפרידה ביניהם: $\delta(a,1) \in F, \delta(f,1) \notin F$.

כנ"ל $\{b,f\}$.

כלומר, קיבלנו ש

$$R = \{\{a,c\},\{b,c\},\{f,c\},\{a,d\},\{b,d\},\{f,d\},\{a,e\},\{b,e\},\{f,e\},\{a,f\},\{b,f\}\}$$

באיטרציה השנייה, S תהיה ריקה. האלגוריתם יעצור.

מה האוטומט המתקבל? איך מפענחים מ-R מהן מחלקות השקילות?

האלגוריתם אמר להסתכל על $\bar{R}$. נעשה את זה: איזה זוגות לא ב-R?

$$\{a,b\} \notin R, \qquad \{\{c,d\},\{d,e\},\{c,e\}\} \nsubseteq R$$

נתבונן היטב: כל המצבים נמצאים ביחס עם f, כלומר, R מפריד בין כל המצבים לבין f. ולכן f הוא מחלקת שקילות בפני עצמו.
וגם אלו שציינו למעלה הם מחלקות שקילות.

טבלת עזר: נבנה טבלת "הצלבות" בין המצבים, כדי לראות מי לא ב-R:

טבלה התחלתית: (שימו לב שהיא סימטרית).

	a	b	c	d	e	f
a	X					
b	X	X				
c	X	X	X			
d	X	X	X	X		
e	X	X	X	X	X	
f	X	X	X	X	X	X

אחרי השלב הראשון:

	a	b	c	d	e	f
a	X		✓	✓	✓	✓
b	X	X	✓	✓	✓	✓
c	X	X	X			✓
d	X	X	X	X		✓
e	X	X	X	X	X	✓
f	X	X	X	X	X	X

ובסוף האלגוריתם:

	a	b	c	d	e	f
a	X		V	V	V	V
b	X	X	V	V	V	V
c	X	X	X			V
d	X	X	X	X		V
e	X	X	X	X	X	V
f	X	X	X	X	X	X

ולכן $\bar{R}=\{\{a,b\},\{c,d\},\{c,e\},\{d,e\},\{f\}\}$. ($f$ כי היא מופרדת מכולם – השורה ועמודה שלה מלאות).

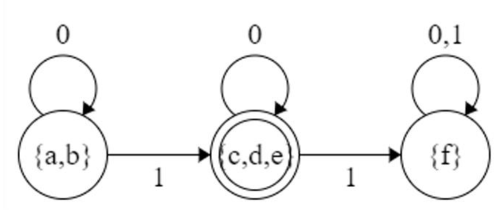
ונקבל את מחלקות השקילות הבאות: $\{a,b\},\{c,d,e\},\{f\}$.

המצב הראשון היה a ולכן $\{a,b\}$ הינה מחלקת השקילות של q_0 החדש.

להן הולכים משם? לפי האוטומט המקורי, עבור 0 אנחנו נשארים במקום. 1 מעביר ל-c או d, כלומר ל- $\{c,d,e\}$.

מכל המצבים משם, 0 משאיר אותנו בקבוצת המצבים הזאת. עבור 1 היה לנו $\tilde{\delta}(c,1)=\delta(d,1)=\delta(e,1)=f$.

מצב f היה ונשאר בור. קיבלנו את האוטומט הבא:



ועכשיו קל לראות מה השפה: $L=\{w\in\{0,1\}^*:\#_1(w)=1\}$.

3 **הוכחת האלגוריתם** (רק טענה 0 הייתה השנה)

צריך להוכיח שהוא עוצר, וצריך להוכיח שהוא מחשב נכון.

בהקשר של ייצור אוטומט שקול, "מחשב נכון" הכוונה שבניית האוטומט נכונה – $\tilde{\delta}$ מוגדרת היטב, $\tilde{F}$ מקבל רק מילים רלוונטיות, האלגוריתם לא מחזיר *FAIL*, ולבסוף ששפת האוטומט שמתקבלת אכן זהה לשפת האוטומט המקורי.

נשתמש בכמה טענות עזר:

טענה 0: האלגוריתם תמיד עוצר.

הוכחת 0: האלגוריתם עוצר כאשר S ריקה. היות ויש $T=\binom{|Q|}{2}$ זוגות אפשריים, האלגוריתם יעצור לאחר לכל היותר $T+1$ איטרציות. (אפשר לשפר את זה, אבל לצורך עצירת האלגוריתם זה חסם מספיק טוב).

שאר ההוכחה לא הייתה בחומר השנה!

טענה 1: אינווריאנטה:

לכל $i\in\mathbb{N}$ מתקיים: בהינתן $q\neq q\in Q$ כלשהו, המחרוזת הקצרה ביותר z כך שקיימים $x\in L_p,y\in L_q$ כך ש $xz\in L(A)\wedge yz\notin L(A)$ היא באורך i אמ"מ $\{p,q\}$ מצטרף אל R בסוף האיטרציה ה i.

מסקנה 1: (מטענה 1): האלגוריתם אינו מחזיר *FAIL*. כלומר, $\bar{R}$ שמתקבל בסוף האלגוריתם הוא אכן יחס שקילות.

הוכחה: $(p,q)\in\bar{R}\Leftrightarrow L_p,L_q$ הן תתי קבוצות (לא ריקות) של אותה מחלקת שקילות של $R_L(A)$.

מסקנה 2: $index(R_L(A))=index(\bar{R})$. אם נצייר את הוכחת המסקנה הקודמת נקבל בקלות את המסקנה הזו. $\bar{R}$ הפריד בין שפות מצבים בדיוק כמו ש $R_L(A)$ אמור להפריד.

טענה 2: אם $\tilde{q}\in\tilde{F}$, אזי $\tilde{q}\subseteq F$.

הוכחה: מההגדרה של $\tilde{F}$ נובע כי אם $\tilde{q}\in\tilde{F}$ אזי קיים $q\in\tilde{q}$ כך ש $q\in F$. נב"ש שקיים $p\in\tilde{q}$ אבל $p\notin F$. מכאן, p,q ניתנים להפרדה ע"י ϵ, כלומר לפי טענה 1, הם צריכים להשתייך למחלקות שקילות שונות של $\bar{R}$ (ולהיכנס ל R כבר באיטרציה 0). בסתירה לכך ש $p,q\in\tilde{q}$, באותה מחלקת שקילות של $\bar{R}$.

טענה 3: $\tilde{\delta}$ מוגדרת היטב. (מוגדרת באותה צורה לכל מועמד $q\in\tilde{q}$ שאנו בוחרים).

הוכחה: הגדרנו $\tilde{\delta}(\tilde{q},\sigma) = [\delta(q,\sigma)]_{\bar{R}}$, כך ש $q \in \tilde{q}$. נראה שאם $p,q \in \tilde{q}$ אזי $[\delta(q,\sigma)]_{\bar{R}} = [\delta(p,\sigma)]_{\bar{R}}$.

נב"ש שזה לא נכון – נסמן $\delta(q,\sigma) = q', \delta(p,\sigma) = p'$. ונניח ש $[p']_{\bar{R}} \neq [q']_{\bar{R}}$. ההנחה הזו שקולה לכך ש $(p',q') \notin \bar{R} \Leftrightarrow (p',q') \in R$. לכן, מטענה 1, קיים z המפריד בין p', q'. אבל אז σz מפריד בין p לבין q בסתירה לכך ש $(p,q) \in \bar{R}$.

טענה 4: $L(A) = L(\tilde{A})$. לשם כך נוכיח את טענת עזר 5:

טענה 5: $\forall w \in \Sigma^* : \hat{\tilde{\delta}}(\widetilde{q_0}, w) = \left[\hat{\delta}(q_0, w)\right]_{\bar{R}}$.

הוכחה באינדוקציה על $|w|$:

בסיס: נבחר $|w| = 0$, כלומר $w = \epsilon$. הטענה מתקיימת לפי הגדרת $\widetilde{q_0}$.

צעד: נניח שהטענה נכונה עבור $|w| < n$ כלשהו, ונוכיח עבור $|w| = n$:

נסמן $w = x\sigma$ כך ש $|x| = n-1$, לכן מתקיים:

$$\hat{\tilde{\delta}}(\widetilde{q_0}, x\sigma) = \tilde{\delta}\left(\hat{\tilde{\delta}}(\widetilde{q_0}, x), \sigma\right) =^{\text{א}} \tilde{\delta}\left(\left[\hat{\delta}(q_0, x)\right]_{\bar{R}}, \sigma\right) =^{\text{ב}} [\delta(p,\sigma)]_{\bar{R}}$$

א – מהגדרת $\hat{\delta}$, והנחת האינדוקציה. ב – מהגדרת $\tilde{\delta}$

(כאשר $p \in \left[\hat{\delta}(q_0, x)\right]_{\bar{R}}$)

הוכחת טענה 4:

$$\begin{aligned}
&w \in L(\tilde{A}) \Leftrightarrow \\
&\hat{\tilde{\delta}}(\widetilde{q_0}, w) \in \tilde{F} \Leftrightarrow \\
&\forall q \in \hat{\tilde{\delta}}(\widetilde{q_0}, w) : q \in F \Leftrightarrow \\
&\forall q \in \left[\hat{\delta}(q_0, w)\right]_{\bar{R}} : q \in F \Leftrightarrow \\
&\hat{\delta}(q_0, w) \in F \Leftrightarrow w \in L(A)
\end{aligned}$$

לפי הסדר: מהגדרת $\tilde{F}$, מטענה 2, מטענה 5, מטענה 2, הגדרת F.

כעת נוכיח את טענה 1: נזכיר אותה:

אינווריאנטה: לכל $i \in \mathbb{N}$ מתקיים: בהינתן $q \neq q \in Q$ כלשהו, המחרוזת הקצרה ביותר z כך שקיימים $x \in L_p, y \in L_q$ כך ש $xz \in L(A) \wedge yz \notin L(A)$ היא באורך i אמ"מ $\{p,q\}$ מצטרף אל R בסוף האיטרציה ה i.

הוכחה באינדוקציה על i:

בסיס: (שלב 1 של האלגוריתם, הגדרת R)

צעד: נניח שהטענה נכונה עבור i כלשהו, ונוכיח עבור $i+1$.

נניח ש z שאורכה $i+1$ היא המחרוזת הקצרה ביותר המפרידה בין זוג $p,q \in Q$ כלשהו. נסמן $z = x\sigma$ ונתבונן ב $q' = \delta(q,\sigma), p' = \delta(p,\sigma)$. לפי ההנחה, w הינה המחרוזת הקצרה ביותר המפרידה בין p', q'. מהנ"א, נקבל כי p', q' הופרדו באיטרציה ה-i של האלגוריתם.

לכן, אם p,q לא הופרדו עוד קודם לכן, הם יופרדו ע"י בדיקת התו σ באיטרציה ה-i. ואכן, הם לא הופרדו קודם לכן שכן אחרת הייתה מחרוזת z' המפרידה ביניהם (ואין כזו, בגלל הנ"א).

זו הוכחה בכיוון אחד. הכיוון השני בדרך כלל במטלה.

4 מבוא לדקדוקים

עד כה הכרנו שתי דרכים להגדיר שפות: $L = \{\ldots\}$, או אוטומט שפת – $L(A)$. נלמד דרך שלישית – דקדוק (שכתוב).

מהו דקדוק? קבוצת משתנים V, קבוצת טרמינלים T, וקבוצה של כללי שכתוב P. כמו כן, יש משתנה התחלתי S. (כל הקבוצות סופיות).

לדוגמה: $S \to BA, B \to a, A \to bab|cab$. אלו הכללים שרשומים ב P. בנוסף, ניתן לראות ש $T = \{a, b, c\}, V = \{S, A, B\}$.

4.1 דקדוק כמגדיר שפה

קבוצת המילים המתקבלות מהדקדוק מוגדרות כאוסף של כל המילים שניתן לקבל כאשר מתחילים במחרוזת S ומבצעים סדרת צעדים בהם משכתבים תת מחרוזת כלשהי, באמצעות אחד מכללי השכתוב. עוצרים כאשר מתקבלת מחרוזת מעל T^*. בדוגמה לעיל נקבל:

$$S \to BA \to aA \to \begin{cases} abab \\ acab \end{cases}$$

כאשר החץ הראשון מכלל הגזירה $S \to BA$, השני מהכלל $B \to a$, השלישי מהכלל של A.

כדי להוכיח נכונות בנייה של דקדוק (כלומר להוכיח כי $L = L(G)$) נצטרך להראות הכלה דו כיוונית (בד"כ באינדוקציה על אורך הגזירה או אורך המילה).

פורמלית:

דקדוק הוא רביעייה $G = (T, V, P, S)$ כאשר:

V קבוצה סופית ולא ריקה של משתנים.

T קבוצה סופית ולא ריקה של טרמינלים (אותיות), הזרה ל-V.

P קבוצה של כללי שכתוב (או כללי גזירה) כך שכל כלל ב-P הוא מהצורה $\alpha \to \beta$ כאשר:
$\beta \in (V \cup T)^*$, $\alpha \in (V \cup T)^* \cdot V \cdot (V \cup T)^*$. (ב-$\alpha$ יש לפחות משתנה אחד, ועוד משתנים וטרמינלים. ב-β לא חייב להיות – זה יכול להיות כל ערבוב של משתנים וטרמינלים).

לדוגמה: הרשום להלן הינו דקדוק - $V = \{S\},\ T = \{a, b\},\ P = \{S \to aSb|ab\}$.

4.2 גזירה

יהי $G = (T, V, P, S)$ דקדוק, ויהיו $\gamma_1, \gamma_2 \in (V \cup T)^*$. נאמר כי γ_2 נגזר ישירות מ γ_1 ונסמן:

$\gamma_1 \Rightarrow_G \gamma_2$ (γ_1 גוזר ישירות בדקדוק G את γ_2),

אם ורק אם:

קיימים $\gamma, \chi, \alpha, \beta \in (V \cup T)^*$ כך ש: $\alpha \to \beta \in P$ וגם $\gamma_1 = \gamma\alpha\chi$, $\gamma_2 = \gamma\beta\chi$.

במלים אחרות: אם ניתן להגיע מ γ_1 אל γ_2 ע"י הפעלה בודדת של כלל שכתוב כלשהו מ P.

לדוגמה, בדקדוק לעיל: $S \Rightarrow ab$.

4.3 עוד קצת סימונים

נסמן $\alpha \Rightarrow_G^* \beta$ אם ניתן להגיע מ-α אל β במספר כלשהו של צעדים (כולל 0 צעדים). בפרט, $\alpha \Rightarrow_G^* \alpha$ לכל α ע"י הפעלת 0 כללים.

נסמן $\alpha \Rightarrow_G^i \beta$ אם ניתן להגיע מ-α אל β ב-i שלבים. לדוגמה, $\alpha \Rightarrow^1 \beta \Leftrightarrow \alpha \Rightarrow \beta$. בד"כ נשמיט את ה G כאשר ברור על איזה דקדוק מדובר.

4.4 שפה של דקדוק

הגדרה: בהינתן דקדוק G, השפה של $L(G)$ מוגדרת כך:

$$L(G) = \{x \in T^* : S \Rightarrow^* x\}$$

כלומר, קבוצת המילים מעל T^*, שאפשר להגיע אליהם מהמשתנה ההתחלתי על ידי כללי הגזירה.

לדוגמה: נתון הדקדוק הבא: $V = \{S\}, T = \{a, b\}, P = \{S \to aSb|\epsilon\}$.

מהי שפת הדקדוק? נעשה כמה צעדים כדי לקבל תחושה:

$$S \to aSb \to aaSbb \to aaaSbbb \to aaaaSbbbb \to a^nSb^n \to a^nb^n$$

זה דקדוק עבור השפה $L = \{a^nb^n : n \geq 0\}$. כי בכל שלב אפשר להוסיף רק a בהתחלה ו-b בסוף.

רואים שלדקדוק יש כוח שלא היה לשפות רגולריות. בעזרת דקדוק די פשוט קיבלנו שפה לא רגולרית.

4.5 דוגמה לבניית דקדוק עבור שפה

נבנה דקדוק לשפת המספרים העשרוניים.

הטרמינלים יהיו $T = \{0,1, \dots ,9\} \cup \{,\}$.

נייצר את המשתנים:

צריך משתנה עזר שמשכתב את $\{0,1, \dots ,9\}$: נקרא לו N_1. הכלל הוא: $N_1 \to 0|1| \dots |9$.

נרשום משתנה עבור קטגוריית מספר דו ספרתי: N_2. ונכניס ל-P את כלל הגזירה הבא: $N_2 \to N_1N_1$.

כנ"ל לגבי תלת ספרתי: $N_3 \to N_1N_1N_1$.

עבור מספרים של יותר מ-3 ספרות, צריך דרך להכניס פסיק: $S \to S, N_3$.

אם יש לנו S, אפשר להוסיף מימין מספר תלת ספרתי אחרי פסיק. ובגלל שה-S תמיד משמאל, המבנה של שלוש ספרות בין כל פסיק יישאר. לדוגמה:

$$S \to S, N_3 \to S, N_3, N_3 \to S, 123{,}456$$

כאשר המעבר האחרון נובע מכללי הגזירה שקבענו בהתחלה. ונרצה גם דרך להפוך את ה-S למספר. הוא יכול להיות מספר חד, דו, או תלת ספרתי: $S \to S, N_3|N_1|N_2|N_3$. ועכשיו נוכל לסיים את השכתוב:

$$S, 123{,}456 \to 78{,}123{,}456$$

הרצאה 8

1 הוכחת נכונות דקדוק

בהרצאה 7 ראינו את הדקדוק:

$$G = (V, T, S, P):$$
$$V = \{S\}, \qquad T = \{a, b\}, \qquad P = \{S \to aSb|ab\}$$

ואמרנו שהשפה המתאימה היא $L = \{a^n b^n : n \in \mathbb{N}^+\}$.

כעת נוכיח את זה באינדוקציה. כלומר נראה כי $L = L(G)$ בהכלה דו כיוונית.

בכל כיוון נבצע אינדוקציה על המשתנה המתאים (גודל המילה, מספר צעדי הגזירה).

1.1 כיוון ראשון: $L \subseteq L(G)$

נוכיח באינדוקציה חזקה על $|w|$:

בסיס: נראה נכונות עבור $|w| \in [2]$:

עבור $|w| \in \{0,1\}$, לא קיימת כזו ב L, ולכן התנאי מתקיים באופן ריק.

עבור $|w| = 2$, רק ab מילה מתאימה, ואכן מתקיים $S \Rightarrow^1 ab$ כי קיים הכלל $S \to ab$.

צעד: יהי $i \geq 3$. נניח כי הטענה נכונה לכל $|w| = j < i$, ונוכיח עבור $|w| = i$.

נחלק לשלושה מקרים:

מקרה א: $|w| = 2k + 1$ (אי זוגי) – אין מילים בשפה באורך אי זוגי. אז מתקיים באופן ריק.

מקרה ב: $|w| = 2k$ (זוגי) – המילה היחידה בשפה באורך הזה היא $w = a^k b^k$. נראה סדרת גזירה עבור w.

$$S \to aSb \Rightarrow^* aa^{k-1}b^{k-1}b = a^k b^k$$

כאשר המעבר השני נובע מהנחת האינדוקציה. ■

1.2 כיוון שני: $L(G) \subseteq L$

תחילה נוכיח טענה חזקה יותר: לכל $n \geq 1$ ולכל צורה פסוקית $\gamma \in (V \cup T)^*$:

אם $S \Rightarrow^n \gamma$ אזי $\gamma = a^n b^n$ או $\gamma = a^n S b^n$. (כלומר, אם הגענו למשהו מ-S ב-n צעדים, אנחנו יודעים איך הוא נראה).

ואכן, הוכחת הכיוון הזה נגמרת כשנסיים עם הטענה החזקה. כי לכל $w \in T^*$, אם $S \Rightarrow^* w$ אזי מההגדרה קיים n כך שמתקיים $S \Rightarrow^n w$ צעדים, ומהטענה, w (שאינה מכילה משתנים) חייבת אם כך להיות מהצורה $w = a^n b^n$ ולכן שייכת ל-L.

נוכיח את הטענה החזקה, באינדוקציה על אורך הגזירה:

בסיס: $n = 1$. ואכן $\gamma = aSb$ או $\gamma = ab$. שכן מפעילים בדיוק כלל אחד מ-P ומתקיים $S \to ab|aSb \in P$.

צעד: נניח שהטענה מתקיימת עבור n כלשהו ונוכיח עבור $n + 1$: תהי γ כך ש $S \Rightarrow^{n+1} \gamma$. אזי מתקיים: $S \Rightarrow^n \beta \Rightarrow \gamma$.

מהנ"א נקבל $\beta = a^n S b^n$ או $\beta = a^n b^n$. אבל המקרה השני לא אפשרי, שכן $\beta \Rightarrow \gamma$ אבל אם $\beta = a^n b^n$ לא ניתן להמשיך לגזור.

אם כך, $\beta = a^n S b^n$. יש שתי אפשרויות לצעד הגזירה האחרון בסדרה (לפי כללי הגזירה של S):

$S \to ab$ ולכן $\beta \Rightarrow a^{n+1} b^{n+1}$, או: $S \to aSb$ ולכן $\beta \Rightarrow a^{n+1} S b^{n+1}$.

ואכן קיבלנו את γ כנדרש בטענה החזקה. מש"ל.

1.3 הוכחת נכונות של דקדוק – דוגמה 2

יהי $G = (\{S\}, \{a, b\}, S, P)$, ורשימת כללי הגזירה: $S \to aSb|aSbb|aSbbb|ab|abb|abbb$.

מהי שפת הדקדוק?

$$L(G) = \{a^n b^k : 1 \le n, k; n \le k \le 3n\}$$

כי בכל שלב שמוסיפים a, מוסיפים לפחות b אחד ולכל היותר 3 b.

נסמן את התוצאה לעיל ב L. הוכיחו כי $L = L(G)$.

כיוון ראשון: $L(G) \subseteq L$. לשם כך נוכיח טענה חזקה יותר:

תהי $\gamma \in (V \cup T)^*$ תבנית פסוקית הנגזרת בדקדוק ב-n צעדי גרירה. אזי $\gamma = a^n S b^k$ או $\gamma = a^n b^k$ עבור $n \le k \le 3n$.

מהטענה החזקה נקבל את הכיוון, שכן כל מילה טרמינלית שנגזרת בדקדוק עבור $\gamma = a^n b^k$ אכן עונה על תנאי השפה.

נוכיח באינדוקציה על אורך הגזירה כי הטענה נכונה.

בסיס: עבור צעד גזירה יחיד $(n = 1)$ נקבל: $S \to ab|abb|abbb|aSb|aSbb|aSbbb$, כנדרש.

נניח כי הטענה מתקיימת עבור t צעדי גזירה, ונוכיח עבור $t + 1$ צעדים.

מהנ"א, אחרי t צעדים, יש לנו משהו מהצורה:

$$S \Rightarrow^t a^t S b^{k'}$$

כאשר $k' \in [t, 3t]$. חייב שיהיה S באמצע, שכן אחרת לא ניתן להמשיך לגזור את הצעד ה- $t + 1$.

נסמן $r \in [1,3]$. לפי כללי הגזירה אחד מהשניים הבאים יכול להתקיים:

$$S \to^t a^t S b^{k'} \to a^{t+1} S b^{k'+r}$$

$$S \to^t a^t b^{k'} \to a^{t+1} b^{k'+r}$$

ואכן $t + 1 \le k' + r \le 3t + 3$, כנדרש.

כיוון שני: $L \subseteq L(G)$. תהי $w \in L$. נוכיח באינדוקציה על $|w|$ כי $w \in L(G)$:

בסיס: עבור $|w| \le 4$, מתקיים: $w = ab, abb, abbb, aabb$. ואכן את שלושת הראשונות אפשר לגזור בצעד יחיד, שכן שלוש כללי הגזירה הרלוונטיים קיימים. והרביעים מגזרת ע"י $S \to aSb \to aabb$. ואלה כל המילים הרלוונטיות.

צעד: נניח עבור מילים עד אורך g, ונוכיח עבור w באורך g. כאשר $g \ge 5$. אם $w \notin L$, הטענה מתקיימת באופן ריק. אחרת, נסמן: $w = a^n b^k$. כאשר $n \le k \le 3n$.

אבחנה: נוכל להניח ש $n \ge 2$. (אחרת w עד אורך 4, כמו בבסיס).

נסמן $k = \alpha n$ כאשר $1 \le \alpha \le 3$, והוא לאו דווקא שלם. יהי $r = \lfloor \alpha \rfloor$ ונסמן:

$$w = a \cdot \underbrace{a^{n-1} b^{k-r}}_{u} b^r = a u b^r$$

שימו לב שהקושי הטכני הוא בבחירת r טובה, כך ש $u \in L$ ולכן נוכל להשתמש עליו בהנחת האינדוקציה.

אם $u \in L$, אזי קיימת ל-w סדרת כללי הגזירה $S \Rightarrow aSb^r \Rightarrow^* aub^r = w$. כאשר הגזירה עם הכוכבית נובעת מכך ש $u \in L$, ומהנחת האינדוקציה. לכן $w \in L(G)$, כנדרש.

נותר להוכיח כי $u \in L$: תחילה נוכיח כי $1 \le n - 1$, ואז נראה כי $1 \le \frac{k-r}{n-1} \le 3$.

החלק הראשון נובע מכך ש $n \ge 2$ (ראינו לעיל). לגבי החלק השני, נתבונן בשבר:

$$Q = \frac{k-r}{n-1} = \frac{\alpha n - \lfloor\alpha\rfloor}{n-1} = \frac{\alpha n - \alpha + \alpha - \lfloor\alpha\rfloor}{n-1} = \frac{\alpha(n-1) + \alpha - \lfloor\alpha\rfloor}{n-1} = \alpha\frac{\alpha - \lfloor\alpha\rfloor}{n-1} \geq \alpha$$

כאשר המעבר האחרון נובע מכך ש $\alpha - \lfloor\alpha\rfloor \geq 0$.

נותר להראות ש $Q \leq 3$. נחלק לשלושה מקרים:

מקרה ראשון: $\alpha = 3$. במקרה זה $Q = 3 \leq 3$.

מקרה שני: $\alpha \leq 2$: במקרה זה:

$$Q = \alpha + \frac{1}{n-1} \leq 2 + 1 = 3$$

מקרה שלישי: $2 < \alpha < 3$: היות ומתקיים $\alpha = k/n$ עבור $k, n \in \mathbb{N}$, אז בהכרח $\alpha \leq 3 - 1/n$ וכן $\alpha - \lfloor\alpha\rfloor \leq \frac{n-1}{n}$. נציב במשוואה של השבר ונקבל:

$$Q \leq 3 - \frac{1}{n} + \frac{\frac{n-1}{n}}{n-1} = 3 \leq 3$$

2 ההירככיה של חומסקי

הוגדרה על ידי נועם חומסקי[1] ב1956, והיוותה בסיס לחקר השפות הפורמליות במדעי המחשב.

הרעיון הכללי הוא, שעל ידי השמת "מגבלות" על דקדוקים, נוכל לסווג אותם לכמה סוגים שכל אחד מהם מתאים למודל "אוטומטים" שונה.

הרעיון המקורי היה שעל ידי השמת "מגבלות" על דקדוקים, נוכל לסווג אותם למורפולוגיה אנגלית (שפות רגולריות), או דקדוק אנגלי (שפות חסרות/תלויות הקשר).

לכל מודל דקדוקי קיים מודל חישובי, סוג של אוטומט, שכוחו מספיק בדיוק לקבלת שפות ממשפחת הדקדוקים המתאימה.

2.1 ההירככיה עצמה

קיימים ארבעה סוגי דקדוקים, ובכל קבוצה עוקבת מוסיפים מגבלות נוספות.

$$\Gamma_3 \subseteq \Gamma_2 \subseteq \Gamma_1 \subseteq \Gamma_0$$

שפה תיקרא מטיפוס i אם קיים דקדוק Γ_i היוצר אותה. יעניין אותנו מהו המודל ה"חלש ביותר" שיכול ליצור אותה.

$\boldsymbol{\Gamma_0}$ – **דקדוקים כלליים:** אין מגבלות. בפרט, מותר שיהיו כללים מקצרי אורך. לדוגמה, $aBAb \to bbc$. לא נעסוק בהם בקורס. המודל הזה כה חזק שהוא שקול למכונת טיורינג כללית, ומכונת טיורינג שקולה למחשב כללי. (נראה בקורס חישוביות).

$\boldsymbol{\Gamma_1}$ – **דקדוקים תלויי הקשר:** כאן יש רק כללים שאינם מקצרי אורך. כלומר, אם $\alpha \to \beta$, אזי $|\alpha| \leq |\beta|$. יוצא דופן הוא $S \to \epsilon$, שמותר. המודל הזה שקול למכונת טיורינג שיש עליה מגבלת זיכרון לינארי. (נראה לקראת סוף הקורס).

$\boldsymbol{\Gamma_2}$ – **דקדוקים חסרי הקשר:** כאן נוספה מגבלה הדורשת שבאגף שמאל של הכללים יהיה תמיד משתנה בודד. כלומר, כל כללי הגזירה מהצורה $\alpha \to \beta$ דורשים ש $\alpha \in V^1$. ושוב, מותר $S \to \epsilon$. השם "חופשיי הקשר" נובע מכך ש α הוא משתנה בודד, ולא משנה היכן הוא מופיע בתוך הצורה הפסוקית, ניתן החליף אותו ב β.

המודל הזה שקול לאוטומט מחסנית אי דטרמיניסטי, ואנחנו נעסוק בו בהמשך הקורס.

---

[1] הוא יהודי-אמריקאי ממוצא פולני, לימד 50 שנה ב-*MIT* ומתגורר במסצ'וסטס (מגניב!). הוא גם מתנגד לעצם הקיום של מדינת ישראל, ותומך מוצהר של חמאס וחיזבאללה (פחות מגניב).

"למשפחת דקדוקים זאת חשיבות מיוחדת כיוון שחלק גדול מהתחביר של שפות התכנות ניתן לתיאור באמצעות דקדוקים חופשיי הקשר". – הטכניון

Γ_3 – דקדוקים רגולריים: דקדוק לינארי ימני או לינארי שמאלי.

כל כללי השכתוב הם מהצורות: $A \to aB$ (או $A \to Ba$, אם זה לינארי שמאלי), $A \to a$, $S \to \epsilon$ $(\forall A, B \in V, \forall \sigma \in \Sigma)$.

מודל זה שקול לשפות רגולריות, כלומר, לכל המודלים הבאים: *NFA*, *DFA*, $\epsilon - NFA$.

2.2 דקדוקים רגולריים – בניית דקדוק בהינתן אוטומט

נחלק את Γ_3 בין דקדוק לינארי ימני (כל הכללים מהצורה $A \to aB$) ובין דקדוק לינארי שמאלי (כל הכללים מהצורה $A \to Ba$). וגם, נוכיח שהם שקולים בכוחם.

נוכיח גם כי דקדוק לינארי ימני שקול לשפות רגולריות, בהכלה דו כיוונית:

משפט 1: קבוצת השפות הרגולריות מוכלת ב Γ_3.

הוכחה: תהי L שפה רגולרית. לכן קיים עבורה אוטומט $A = (Q, \Sigma, q_0, \delta, F)$. נבנה דקדוק G כך ש $L(G) = L(A) = L$.

רעיון הבנייה: נרצה לסמלץ את פעולת A בעזרת תהליך הגזירה בדקדוק של G. נקצה משתנה לכל מצב של A. נרצה שיתקיים:

$$\forall w \in \Sigma^*, \forall p, q \in Q(A) : \hat{\delta}(p, w) = q \Leftrightarrow p \Rightarrow^* wq$$

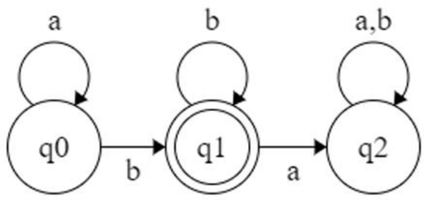
כמו כן, נצטרך לטפל במצבים סופיים (ב F).

פורמלית, נגדיר: $G = (V = Q, T = \Sigma, S = q_0, P)$.

לכל $a \in \Sigma, q \in Q$ נוסיף ל-P את הכלל $q \to ap$ אם $p = \delta(q, a)$.

אם $p \in F$, נוסיף גם את הכלל $q \to a$.

לדוגמה: נתון האוטומט הבא:



נמיר אותו לדקדוק:

$$T = \{a, b\}, \qquad S = \{q_0\}, \qquad V = \{q_0, q_1, q_2\}$$

$$\begin{aligned} P = \{q_0 \to aq_0, &\qquad q_0 \to bq_1, \\ q_1 \to bq_1, &\qquad q_1 \to aq_2, \\ q_2 \to aq_2, &\qquad q_2 \to bq_2, \\ q_0 \to b, &\qquad q_1 \to b\} \end{aligned}$$

2.3 לקראת הוכחת נכונות הבנייה

נוכיח תחילה עבור שפות שבהן $\epsilon \notin L$, ואז נראה איך מכלילים.

נשתמש בכמה טענות עזר:

טענת עזר 1: יהי G דקדוק לינארי ימני. אזי, אם מתקיים $A \Rightarrow^* \alpha$ (כאשר $A \in V^1,\ \alpha \in (V \cup T)^*$) אז α מהצורה wB או w כאשר $w \in T^*$, ו- $B \in V^1$.

הוכחה: באינדוקציה על אורך סדרת הגזירה של α.

בסיס: $A \Rightarrow^0 \alpha$. במקרה זה בהכרח $\alpha = A$ כלומר היא אכן מהצורה wB כאשר $B = A,\ w = \epsilon \in (V \cup T)^0$.

נניח שהטענה נכונה עבור סדרת גזירות באורך i, ונוכיח עבור סדרת גזירות באורך $i+1$:

תהי α כך ש $A \Rightarrow^{i+1} \alpha$. אפשר לרשום את סדרת הגזירה בתור $A \Rightarrow^i \beta \Rightarrow \alpha$.

מהנ"א, β מהצורה w או wB כאשר $B \in V,\ w \in T^*$.

היות וגזרנו מ-β את α, אזי β חייבת להיות מהצורה wB. מהגדרת דקדוק לינארי, בצעד האחרון ניתן להפעיל אחד מהכללים הבאים:

כלל מהצורה $B \to \sigma C$ ואז נסיק $\alpha = w\sigma C$, כנדרש,

או כלל מהצורה $B \to \sigma$ או $B \to \epsilon$. במקרה זה נקבל מילה טרמינלית $(w, w\sigma)$, כנדרש.

טענת עזר 2: $\forall w \in \Sigma^*, \forall p, q \in Q(A) : \hat{\delta}(p,w) = q \Leftrightarrow p \Rightarrow^* wq$.

לכל מילה מעל הא"ב ולכל שני מצבים באוטומט: המעבר ממצב אחד למצב אחר על ידי קריאת המילה, שקול לכלל שיכתוב.

הוכחה: (רק בכיוון אחד), באינדוקציה על $|w|$. נניח ש $\hat{\delta}(p,w) = q$:

בסיס: עבור $|w| = 0$ נקבל $w = \epsilon$, מתקיים $\hat{\delta}(p', \epsilon) = p'$ (לכל אוטומט), ואכן $p' = p'$ ע"י אפס צעדים. (תזכורת: בחרנו DFA עבור השפה L, ואם היינו לוקחים $\epsilon - NFA$ הבנייה הייתה מטפלת בזה בהתאם).

נניח כי הטענה נכונה עבור $|w| = i$ ונוכיח עבור $|w| = i+1$. כרגיל, נסמן $w = u\sigma$. מתקיים:

$$\hat{\delta}(p,w) = \delta\big(\hat{\delta}(p,u), \sigma\big) = \delta(q', \sigma) = q$$

כאשר השוויון השני נובע מהנ"א.

מהנ"א, נקבל $p \Rightarrow^* uq'$, ומהגדרת הבנייה נקבל שבדקדוק קיים הכלל $q' \to \sigma q$ ולכן קיימת סדרת הגזירה $p \Rightarrow^* uq' \to wq$, כנדרש.

ניגש להוכיח את הטענה המרכזית של הבנייה, בהכלה דו כיוונית:

2.4 הוכחת נכונות הבנייה

כיוון ראשון: $L(A) \subseteq L(G)$: תהי $y \in L(A)$. היות ו- $\epsilon \notin L(A)$, נקבל $y \neq \epsilon$.

נסמן $y = x\sigma$ (כאשר יכול להיות ש $x = \epsilon$). נסמן $\hat{\delta}(q_0, x) = p'$.

מטענת עזר 2 מתקיים $q_0 \Rightarrow^* xp'$. בנוסף: $\hat{\delta}(q_0, y) = \delta\big(\hat{\delta}(q_0, x), \sigma\big) = \delta(p', \sigma) \in F$.

מהבנייה, היות ו- $q \in F$, קיים ב-G כלל הגזירה $p \to \sigma$. לכן, קיימת ב-G סדרת הגזירה $q_0 \Rightarrow^* xp' \Rightarrow x\sigma = y$.

כלומר, אכן $L(A) \subseteq L(G)$.

כיוון שני: $L(G) \subseteq L(A)$: תהי $w \in L(G)$, אזי קיימת סדרת גזירה באורך לפחות 1 מהצורה $q_0 \Rightarrow^* w$. (באורך לפחות 1 כי q_0 הוא משתנה, ו-w הוא טרמינל).

נרשום $q_0 \Rightarrow^* \beta \Rightarrow w$. מטענת עזר 1, בהכרח β מהצורה uB כאשר: $B \in V, u \in T^*$.

כלומר, הצעד האחרון הוא מהצורה $uB \Rightarrow w$ (כמובן ש-w טרמינלית).

כלומר, בהכרח בצעד הזה השתמשנו בכלל גזירה מהצורה $B \to \sigma$.

מהבנייה, בהכרח $\delta(B,\sigma) = q \in F$. שכן אחרת אין לנו כלל $B \to \sigma$. מטענת עזר 2 נקבל $\hat{\delta}(q_0,u) = B$ ולכן:

$$\hat{\delta}(q_0,w) = \delta\big(\hat{\delta}(q_0,u),\sigma\big) = \delta(B,\sigma) = q$$

היות ו- $q \in F$, נקבל $w \in L(A)$, כנדרש.

2.5 הכללה עבור אפסילון

ומה אם $\epsilon \in L$?

נבנה דקדוק כמו קודם, אבל כעת נוסיף את הכלל $q_0 \to \epsilon$ ונקרא לדקדוק החדש G'. נוכיח כי $L(G') = L(A)$.

הכיוון הראשון $L(A) \subseteq L(G')$ הוא פשוט. לכל $w \neq \epsilon$ הוכחנו כבר, ועבור $w = \epsilon$ יש לנו את כלל הגזירה החדש: $q_0 \to \epsilon$. ולכן $q_0 \Rightarrow \epsilon$ כלומר $\epsilon \in L(G')$.

הכיוון השני $L(G') \subseteq L(A)$ יותר קשה.

כפי שראינו לעיל, $\epsilon \in L(A), L(G')$ ולכן נותר להראות שלא הוספנו מילים חדשות ל-$L(G)$ שלא היו מקודם.

כלומר, נראה שאם $\epsilon \neq w \in L(G')$ אז $w \in L(G)$.

המצב הבעייתי היחיד הוא אם השתמשנו בכלל $q_0 \to \epsilon$ במהלך סדרת הגזירה.

בעזרת טענת עזר 1, זה יכול להתרחש רק בצעד הגזירה האחרון. (כי העלמנו את המשתנה).

נסמן את סדרת הגזירה: $q_0 \Rightarrow_G^* uq_0 \to_G u$.

כעת, מכיוון ש $q_0 \in F$ (שכן $\epsilon \in L(A)$) בהכרח, בצעד לפני האחרון התבצעה גזירה $\beta = lp \to l\sigma q_0$ (כאשר $l\sigma = u$).

אבל, היות ו-$q_0 \in F$, קיים הכלל $p \to \sigma$ ולכן כבר בצעד הזה יכולנו להשתמש בו ולהגיע אל u מבלי להשתמש בכלל $q_0 \to \epsilon$.

כלומר יכולנו לייצר את המילה גם בלי הכלל החדש, כנדרש.

זה מוכיח שבהינתן *DFA* ניתן לבנות דקדוק לינארי ימני.

2.6 בניית אוטומט בהינתן דקדוק

כעת, נראה את הבנייה של הכיוון השני, בעזרת $\epsilon - NFA$:

בהינתן $G = (V,T,S,P)$ לינארי ימני, נבנה $A_G = \big(Q = V \cup \{q_f\}, \Sigma = T, q_0 = S, \delta, \{q_f\}\big)$. כאשר δ מוגדרת:

לכל $A, B \in V$, $\sigma \in T$, אם $A \to aB \in P$ נוסיף את B ל- $\delta(A,\sigma)$.

אם $A \to \sigma \in P$, נוסיף את q_f ל- $\delta(A,\sigma)$.

אם $A \to \epsilon \in P$, נוסיף את q_f ל- $\delta(A,\epsilon)$.

כי אלו המצבים היחידים שהתווספו ל- $\delta(A,\sigma)$-ים השונים.

גם את זה אפשר להוכיח באינדוקציה.

2.7 דקדוק לינארי ימני ושמאלי

תזכורת: דקדוק רגולרי הוא המודל המוגבל ביותר. קיימים בו רק את הכללים $S \to \epsilon, A \to \sigma$, ואמרנו שבדקדוק לינארי ימני יש $A \to \sigma B$ ובדקדוק לינארי שמאלי יש $A \to B\sigma$.

בהמשך נראה שאם מאפשרים גם לינארי ימני וגם לינארי שמאלי באותו דקדוק, כבר הגדלנו את כוח המודל ועברנו לדקדוק חפשי הקשר. כעת נוכיח שדקדוק לינארי ימני שקול לדקדוק לינארי שמאלי. ונקבל שלינארי שמאלי גם שקול לשפות רגולריות.

תהי L שפה רגולרית. לכן, קיים דקדוק לינארי ימני G היוצר אותה. בגלל ש-L רגולרית, גם L^R (רוורס) רגולרית, ולכן קיים דקדוק לינארי ימני G_R שיוצר את L^R. כלומר $L^R = L(G_R)$.

נתאים ל- G_R דקדוק לינארי שמאלי G_L כך:

כל כלל מהצורה $A \to \sigma$ מ-G_R ייכנס גם אל G_L.

כל כלל מהצורה $A \to \sigma B$ מ-G_R ייכנס גם אל G_L עם היפוך: $A \to B\sigma$.

טענה: $x \in L(G_L) \Leftrightarrow x^R \in L(G_R)$. (לא נוכיח). בהגיון – אם הפכנו את הכיוון של הכללים, אז כל בנייה תקרה פשוט הפוך.

הוכחה: הכלה דו כיוונית באינדוקציה על אורך הגזירה.

נשתמש בזה כדי לסיים את ההוכחה: $L(G_L) = \left(L(G_R)\right)^R = (L^R)^R = L$, ולכן הדקדוק הלינארי השמאלי אכן גוזר את השפה L, כנדרש.

אלגוריתם הנובע מההוכחה: יהי $A = (Q, \Sigma, q_0, \delta, F)$. נבנה את A^R. נמיר אותו לדקדוק לינארי ימני. בדקדוק נחליף כל $A \to \sigma B$ ב- $A \to B\sigma$, וכך יצרנו דקדוק לינארי שמאלי לשפה.

הרצאה 9

1 דקדוק חסר הקשר – Γ_2

דקדוק $G = (T, V, P, S)$ ייקרא חסר הקשר (ח"ה) אם כל כלל גזירה ב-P הוא מהצורה $A \to \alpha$ כאשר $A \in V$, $\alpha \in (V \cup T)^*$.

בנוסף, הכלל היחיד שמקצר הוא $S \to \epsilon$. (משתנים אחרים לא יכולים).

1.1 שפה חסרת הקשר

נאמר ששפה $L \subseteq \Sigma^*$ היא ח"ה אם קיים דקדוק ח"ה G כך ש $L = L(G)$.
(נובע מכאן שגם שפות רגולריות הן ח"ה, אך לא להיפך).

דוגמה לבניית דקדוק לשפה:

תהי $L = (a^l b^m c^m d^n : 0 \leq l \leq n,\ 1 \leq m)$. נבנה לשפה דקדוק ח"ה.

קל לראות ש L אינה רגולרית (בעזרת למת הניפוח לשפות רגולריות. ואינטואיטיבית, כי צריך לספור). נראה ש L ח"ה ע"י בניית דקדוק מתאים. בכללי הגזירה נלך מהחוץ פנימה.

קל לראות ש $T = \{a, b, c, d\}$.

נרצה משתנה שבהינתן a ייצר גם d (כדי לקיים את התנאי $l \leq n$). $R \to aRd \mid ad$.

אותו משתנה גם יכול להוסיף עוד d כרצוננו, מבלי תלות באותיות אחרות. $R \to Rd \mid d$.

נרצה משתנה שייצר את $b^m c^m$ של האמצע (ראינו כבר בנייה כזו). $M \to bMc \mid bc$.

וגם, צריך להבטיח שנשתמש פעם אחת לפחות ב-M. כדי שיתקיים $1 \leq m$. $R \to M$.

הבעיה היא שזה לא **מכריח** אותנו להגיע ל-M. אז נשנה את הכללים שכבר כתבנו – לא נאפשר ל-R לעצור בלי להגיע ל-M. כלומר נוריד את האפשרות לשכתב רק טרמינלים. בסה"כ, נקבל את הכללים:

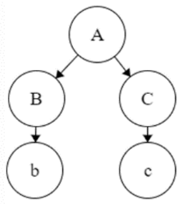
$$P = \{R \to aRd \mid Rd \mid M, \qquad M \to bMc \mid bc\}$$

והדקדוק: $G = (\{a, b, c, d\}, \{R, M\}, P, R)$.

1.2 עצי גזירה

לסדרת גזירה של w כלשהו בדקדוק נתון G, בד"כ קיימות סדרות גזירה נוספות השונות ממנה רק בסדר של "ניתוח" המשתנים. בד"כ נרצה לחשוב על סדרות כאלה בתור שקולות. למשל, לקומפיילר אין חשיבות לסדר "פיתוח" המשתנים.

דוגמה: $G = (\{A, B, C\}, \{b, c\}, A, P)$ שבו $P = \{A \to BC, B \to b, C \to c\}$, למילה $w = bc$ יש שתי סדרות גזירה שונות עד כדי סדר הניתוח: $A \to BC \to bC \to bc, \quad A \to BC \to Bc \to bc$. נאמר כי שתי גזירות אלה שקולות.



כדי לתפוס את כל הסדרות השקולות לסדרה נתונה, במובן זה, ניתן לבטא את תהליך הגזירה כ"עץ גזירה". המילה המתקבלת היא רצף העלים משמאל לימין. כאמור, כל סדרת גזירה מגדירה עץ גזירה יחיד, אבל עץ גזירה יכול להגדיר מספר סדרות גזירה שונות בסדר הפיתוח.

הגדרה פורמלית: עץ גזירה עבור דקדוק ח"ה $G = (T, V, P, S)$ הוא עץ סופי, <u>סדור</u> (סדר הבנים חשוב), המקיים:

1. כל צומת מסומנת בסימן מתוך $V \cup T \cup \{\epsilon\}$.
2. השורש יסומן בד"כ S (לפעמים נדבר על תתי עצים, ושם זה לא בהכרח מתקיים).
3. הסימון של כל צומת פנימי שאינו עלה, הוא משתנה מ-V.
4. צומת המסומן ב-ϵ הוא בן יחיד.
5. אם צומת פנימי מסומן ב-A, ולבניו יש סימונים $X_1, X_2, \ldots X_t$ (בסדר הזה!), אזי $(A \to X_1 X_2 \cdots X_t) \in P$. (כלומר A משכתב את $X_1 X_2 \cdots X_t$).

1.3 חזית של עץ גזירה

הגדרה: המחרוזת המתקבלת ע"י שרשור של כל העלים משמאל לימין הינה **חזית** של עץ הגזירה. אם החזית של עץ הגזירה מורכבת ממילה טרמינלית, נאמר שהעץ הזה הוא **עץ גזירה מלא**.

מה הקשר בין עץ גזירה לבין תבנית פסוקית של דקדוק?

טענה: α היא מילת חזית עץ הגזירה עם שורש A (בדקדוק נתון G) אמ"מ $A \Rightarrow_G^* \alpha$.

ולכן עצי גזירה שימושיים להבנת המילים המתקבלות.

הוכחת הטענה באינדוקציה: כיוון אחד. נראה שאם α הינה חזית של עץ גזירה חוקי ששורשו A, אזי $A \Rightarrow_G^* \alpha$.
באינדוקציה חזקה על מספר הצמתים הפנימיים בעץ:

בסיס: עבור $n = 0$ צמתים פנימיים, כל העץ הוא A. ולכן חזית העץ היא $A = \alpha$ ואכן $A \Rightarrow_G^* \alpha$ כי $A \Rightarrow^0 A$.
(במקרה זה, העץ אינו עץ גזירה מלא, שכן חזית העץ אינה מילה טרמינלית).

יהי $i \in \mathbb{N}^+$. נניח שהטענה נכונה לכל $j < i$ צמתים פנימיים, ונראה עבור i צמתים.

היות ו- $i > 0$, יש ל-A בנים כלשהם. נסמנם $X_1, X_2, \dots X_t$ (בסדר הזה).

מהגדרת עץ הגזירה, בדקדוק G קיים הכלל $A \to X_1 X_2 \cdots X_t$.

נתבונן בתתי העצים של $X_1, X_2, \dots X_t$: ממבנה העץ כולו מתקיים ש $\alpha = \alpha_1 \alpha_2 \cdots \alpha_t$.
(כאשר לכל i, הביטוי α_i הינו החזית של תת העץ הנובע מ X_i).

היות וקיימים הצמתים $X_1, X_2, \dots X_t$ (לפחות אחד), לכל אחד מתתי העצים הללו יש לכל היותר $i - 1$ צמתים פנימיים, ולכן הנ"א חלה עליהם. לכן, בדקדוק G, לכל i קיימות סדרות הגזירה $X_i \Rightarrow^* \alpha_i$. ולכן קיימת הסדרה:

$$A \to X_1 X_2 \cdots X_t \Rightarrow^* \alpha_1 \alpha_2 \cdots \alpha_t = \alpha$$

מסקנה: קבוצת מילות החזית של כל עצי הגזירה המלאים בדקדוק G היא $L(G)$. כי ראינו שכל מילת חזית היא מילה בשפה. וגם שכל מילה בשפה, היא מילת חזית.

1.4 תת עץ גזירה

תת עץ גזירה הוא חלק מהעץ המכיל צומת כלשהו ואת כל צאצאיו.

טענה (לא נוכיח): יהי T_1 תת עץ גזירה של T_2, ותהיינה α_1, α_2 מילות החזית של T_1, T_2 בהתאמה. אזי, α_1 תת מילה של α_2.

2 צורות נורמליות

2.1 פישוט דקדוק

בהינתן דח"ה, לפעמים הוא ייראה מסובך מדי, ונרצה לפשט אותו כדי להוכיח עליו טענה מסויימת. לדוגמה, בשביל למת הניפוח.
הרעיון הכללי של הפישוט יקביל, במקצת, לרעיון הצמצום של אס"ד.

ארבעה אלגוריתמים הנוגעים לפישוט דקדוקים:

1. סילוק משתנים שאינם טרמינליים,
2. סילוק משתנים לא ישיגים,
3. סילוק כללי אפסילון,
4. סילוק כללי יחידה.

(לא נראה אותם בקורס הזה).

שתי צורות נורמליות של דקדוקים:

הצורה הנורמלית של חומסקי: כל כללי הגזירה הם מהצורה $A \to BC$ או $A \to \sigma$.
(באופן טבעי, אין דקדוק רגולרי בצורה הנורמלית של חומסקי).

הצורה הנורמלית של גרייבך: כל כללי הגזירה הם מהצורה $A \to a\beta$. (a היא אות טרמינלית).
כלומר, בכל צעד גוזרים לפחות אות אחת, ואז עוד מחרוזת מעל $V \cup T$.

נתייחס לשפות שלא מכילות ϵ (אחרת מסתבכים עם דקדוקים שבהם $S \to AS$ וכדומה).

משפט: כל שפה חסרת הקשר (ללא אפסילון) ניתנת לייצור ע"י דקדוק בצורה נורמלית. נוכיח לגבי הצורה הנורמלית של חומסקי.
(בצורה של גרייבך לא נשתמש בקורס).

2.1 מעבר לצורה הנורמלית של חומסקי

נתרגם את G המייצר את L לצורה הנורמלית של חומסקי: (לא למדנו את שלבים 1,2).

1. סילוק כללי אפסילון (כלומר, כללים מהצורה $A \to \epsilon$) וסילוק משתנים שאינם ניתנים להשגה.
2. סילוק כללי יחידה ($A \to B$). (זה לא מייצר כללי אפסילון – היזכרו בחלק השלישי של כללי הגזירה מהאלגוריתם ההוא).
3. תרגום כללים מהצורה $A \to X_1X_2 \cdots X_k$ (כאשר $k > 2$) לצורה חוקית:
   a. אם $X_i = \sigma$, אזי נוסיף משתנה חדש C_σ ואת הכלל $C_\sigma \to \sigma$, ונחליף את X_i ב C_σ. נבצע את זה לכל כלל גזירה ב-P ולכל טרמינל בו.
   b. לכל $A \to B_1 \cdots B_k$ עבור $k \geq 3$: נוסיף משתנים $D_1 \dots D_{k-2}$ ונחליף את הגזירה בקבוצת הגזירות

$$A \to B_1D_1, \qquad D_1 \to B_2D_2$$
$$D_{k-3} \to B_{k-2}D_{k-2}, \qquad D_{k-2} \to B_{k-1}B_k$$

3 למת הניפוח לשפות חסרות הקשר

אם L שפה חסרת הקשר, אזי קיים n כך שלכל מילה $z \in L$ המקיימת $|z| \geq n$ קיים פירוק מהצורה $z = uvwxy$ כאשר:

1. $|vwx| \leq n$
2. $1 \leq |v| + |x|$
3. $\forall i \geq 0 : uv^iwx^iy \in L$

בנוסף, אם G דקדוק ח"ה מהצורה הנורמלית של חומסקי יוצר את $L \setminus \{\epsilon\}$ וגם $|V(G)| = k$,
(כלומר אם יצרנו דקדוק ח"ה לשפה לפי הצורה הנורמלית של חומסקי ויש לנו k משתנים), אזי $n \leq 2^k$.

3.1 הוכחת הלמה – מבט על

נראה שבהינתן מילה z מספיק ארוכה השייכת לשפה, נוכל לפרק אותה לחמישה חלקים $z = uvwxy$, המקיימים (עבור משתנים A, S מהדקדוק):

1. $S \to^* uAy$
2. $A \to^* vAx$
3. $A \to^* w$

למה? כי אם ניתן להגיע למשתנה A, והוא גוזר את עצמו עם תוספת, קיבלנו ניפוח (ופינצ'ור) בדומה למעגל שראינו בלמת הניפוח לשפות רגולריות.

3.2 מסלול בעץ

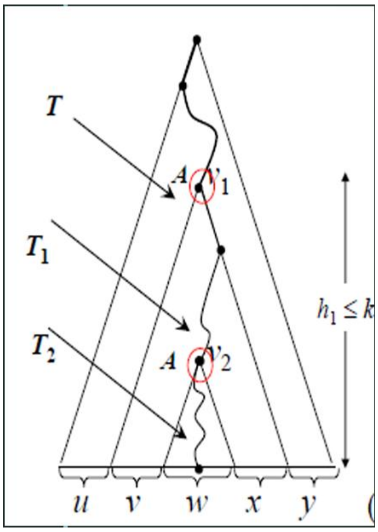
יהי G דקדוק בצורה הנורמלית של חומסקי היוצר את $L \setminus \{\epsilon\}$, ויהי $|V(G)| = k$. תהי $z \in L$.
(אגב, מה עושים עם אפסילון? טכנית, מחקנו מילה מהשפה. זה בסדר כי היא לא ארוכה יותר מ-n).

גובה עץ הגזירה של z הינו לפחות $\log_2(|z|)+1$. (למה? כי אם נתעלם רגע מהשלב האחרון של הטרמינלים, העץ העמוק ביותר שיכול להתקבל הוא עץ בינארי מלא בן $|z|$ עלים. ולכן גובה העץ הוא לפחות $\log_2(|z|)$. נחזיר את הטרמינלים, ולכן הוספנו עוד 1 לגובה העץ).

יהי $n=2^k$, ותהי $z\in L$ כך ש $n\le|z|$. אזי נקבל שגובה עץ הגזירה של z מקיים:

$$k+1=\log_2(2^k)+1\le\log_2(|z|)+1\le h$$

כלומר, מסלול ארוך ביותר בעץ מכיל לפחות $k+2$ צמתים, שהם לפחות $k+1$ צמתים פנימיים שמסמנים משתנים. (ספירת העומר, שם שם). היות ויש רק k משתנים, לפי עקרון שובך היונים קיים במסלול משתנה המופיע פעמיים, כלומר שני צמתים המכילים את אותו המשתנה. זה ה"מעגל" – אם היינו בשלב מסויים במשתנה כלשהו, עשינו צעדים והגענו שוב לאותו משתנה. וכמובן נוכל לעשות את אותם הצעדים שוב.



במסלול הארוך ביותר הזה, נסתכל מהעלה כלפי מעלה, עד שנגיע למשתנה שמופיע פעמיים – בפעם העליונה, זה יקרה עד הגובה $h_1\le k$. (למה? בהמשך).

נסמן את המשתנה הזה ב-A, ואת שני הצמתים עם A נסמן בתור v_1 (הקרוב לשורש), ו-v_2.
נסמן T_1 את תת-העץ המושרש ב-v_1, ו-T_2 את תת-העץ המושרש ב-v_2.

בגלל שאנחנו בצורה הנורמלית של חומסקי, ה-A הראשון משכתב שני משתנים. אם אחד מהם (בה"כ הימני) הוא ה-A השני, אז במקרה הזה x יהיה ריק. אבל אז v יהיה בוודאות לא ריק, אז אנחנו בסדר.

נסמן את מילת החזית של T_2 ב-w. לפי הטענה מתחילת ההרצאה: $A\Rightarrow^* w$.

T_2 הוא תת עץ של T_1 ולכן מילת החזית של T_1 היא vwx, ושוב: $A\Rightarrow^* w$.
ויותר מכך, נוכל לרשום $A\Rightarrow^* vAx$. (נסמן ב **).

לסיום – מילת החזית של כל העץ היא $z=uvxyz$ ולכן $S\Rightarrow^* uAy$. (נסמן *).

במסלול הארוך ביותר הזה, נסתכל מהעלה כלפי מעלה, עד שנגיע למשתנה שמופיע פעמיים – בפעם העליונה. זה יקרה עד הגובה $h_1\le k$. למה?

אם המשתנה העליון גבוה יותר, אזי קיים מישהו מתחתיו שגם מופיע פעמיים, וניקח אותו במקום. בנוסף, היות ולקחנו את המסלול הארוך ביותר, אין מסלול המוביל מ-A לעלים שאורכו גבוה יותר. חסם הגובה הה מאד יעיל, בזכות חומסקי קיבלנו שמילת החזית של הA הראשון חסומה באורך 2^k.

3.3 הפירוק

כעת נראה כי הפירוק $z=uvwxy$ מקיים את תנאי הלמה:

1. כיוון שגובה T_1 מקיים $h_1\le k$, נקבל כי חזית העץ vwx מקיימת $|vwx|\le 2^k=n$. פה השתמשנו בכך שבצורה הנורמלית של חומסקי, כל הכללים מהצורה $A\to BC$.
2. $1\le|v|+|x|$. נראה כי לא ייתכן ששניהם ϵ.
בצעד הראשון מ-v_1 לכיוון v_2 הפעלנו כלל מהצורה $A\to BC$.
אם T_2 נגזר מ-B, אזי בתת העץ C יש לפחות אות אחת (ואז $1\le|x|$).
אם T_2 נגזר מ-C, אזי בתת העץ B יש לפחות אות אחת (ואז $1\le|v|$).

התנאי השלישי: נוכיח באינדוקציה על i כי $S\Rightarrow^* uv^iAx^iy$.

בסיס: עבור $i=0$ נקבל $S\Rightarrow^* uAy$. ואכן $S\Rightarrow^* uAy$, כפי שהראנו בבנייה.

נניח כי הטענה מתקיימת עבור $i-1$, כלומר $S\Rightarrow^* uv^{i-1}Ax^{i-1}y$. נוכיח עבור i:

$$S \Rightarrow^* uv^{i-1}Ax^{i-1}y \Rightarrow^* uv^{i-1}vAxx^{i-1} = uv^iAx^iy$$

כאשר החץ הראשון נובע מהנ"א, והשני מ **.

כדי לסיים את הוכחת הלמה נותר רק להשתמש בעובדה ש $A \Rightarrow^* w$, כפי שהראנו בבנייה (חזית העץ T_2). ונקבל uv^iwx^iy.

4 דוגמאות

4.1 דוגמה 1

תהי $L = \{a^jb^jc^j : j \geq 1\}$ מעל $\Sigma = \{a, b, c\}$. הוכיחו כי L אינה ח"ה.

נב"ש שהיא כן ח"ה, ויהי n הקבוע המובטח בלמה. נבחר: $z = a^nb^nc^n \in L$. מתקיים $|z| = 3n \geq n$. נתבונן בפירוקים המקיימים את תנאי הלמה:

$$a \ldots {}_1 \ldots a \;{}_4\; b \ldots {}_2 \ldots b \;{}_5\; c \ldots {}_3 \ldots c$$

המספרים 1-5 מסמנים מיקום אפשרי של vwx.

1. $vwx = a^{|vwx|}$
2. $vwx = b^{|vwx|}$
3. $vwx = c^{|vwx|}$
4. $vwx = a \ldots b$
5. $vwx = b \ldots c$

עבור פירוק 1, נבחר $i = 2$ ונקבל $a^{n+|vx|}b^nc^n \notin L$, בסתירה ללמה. מקרים 2,3 דומים.

עבור פירוק 4 נבחר $i = 0$ ונקבל $a^lb^kv^n \notin L$, שכן $l < n$ או $k < n$. מקרה 5 דומה.

בכל הפירוקים מצאנו i שעבורו המילה המנופחת לא בשפה, ולכן L אינה ח"ה.

4.2 דוגמה 2

נוכיח ש $L = \left\{a^{j^2} : j \geq 1\right\}$ אינה ח"ה:

נב"ש שהיא כן, ויהי n הקבוע המובטח בלמה. נבחר $z = a^{n^2}$. מתקיים $|z| = n^2 \geq n$ ולכן ע"פ הלמה קיים פירוק $z = uvwxy$ המקיים: $|v| + |x| \geq 1$, $|nwx| \geq n$. בכל פירוק: $vwx = a^{|vwx|}$. נסמן: $v = a^t, x = a^k$ (כאשר $t \geq 1 \vee k \geq 1$). עבור $i = 2$ נקבל: $uv^2wx^2y = a^{n^2+t+k}$. מתקיים: $n^2 < n^2 + t + k \leq n^2 + n < (n+1)^2$ (כי $|vwx| \leq n$). כלומר המילה המנופחת לא בשפה, בסתירה ללמה. אז L אינה ח"ה.

אם השפה הייתה: $L = \left\{a^{n^2}b^k : n \geq 1, k \geq 0\right\}$, לא היינו מצליחים. כי אז "האויב" היה בוחר פירוק שבו $vwx \in b^*$, ואז כל ניפוח מנפח רק את ה-b, והמילה הייתה נשארת בשפה.

4.3 דוגמה 3

תהי $L = \{a^nb^n : 1 \leq n\}$, נראה שהיא מקיימת את למת הניפוח לשפות חסרות הקשר:

נבחר n (נגיד בהמשך מה הוא), לכל $z \in L$ כך ש $|z| \geq n$ נגדיר פירוק $z = uvwxy$:

אנחנו צריכים פירוק שעונה על תנאי הלמה, כך שכל ניפוח של v, x ישאיר את המילה בשפה. נבחר: $v = a, x = b, w = \epsilon$. האותיות u, y יהיו הקצוות. נגדיר פורמלית: לשם נוחות, נגדיר $t = |z|/2$. ואז $u = a^{t-1}, y = b^{t-1}$. ואז כל ניפוח מוסיף את אותו מספר ש a, b, והסדר שלהם נשמר (כל ה-a ואז כל ה-b).

צריך שזה יתקיים גם עבור $i = 0$. במקרה הזה, ה-x, v נעלמים ונישאר רק עם u, y. נדרוש שהם יהיו ארוכים מספיק: $1 \le t - 1$. כלומר $2 \le t$, כלומר $4 \le |z|$. וזה מגדיר את ה-n שלנו.

4.4 דוגמה 4

תהי $L = b^*c^*d^* \cup a^* \cdot \{b^j c^j d^j : j \ge 1\}$. נראה שהיא מקיימת את למת הניפוח לשפות חסרות הקשר:

כל מילה מהצורה $b^*c^*d^*$, אפשר לנפח בלי בעיה.

כל מילה מהצורה $a^* \cdot \{b^j c^j d^j : j \ge 1\}$, נבחר $v = a, x = \epsilon$ וננפח. כדי שנוכל לבחור $v = a$, נדרוש $n = 1$.

אם הבחירה של a^* היא ϵ, המילה היא בעצם מהצורה $b^*c^*d^*$.

כלומר, לא הצלחנו להראות שהיא לא ח"ה. בשביל זה נצטרך תכונות סגור.

5 תכונות סגור

כמו שיש תכונות סגור לשפות רגולריות, יש תכונות סגור לשפות ח"ה.

בהינתן L, L_1, L_2 שפות ח"ה, ו-R שפה רגולרית, גם השפות הבאות ח"ה:

1. איחוד: $L_1 \cup L_2$.
2. שרשור: $L_1 \cdot L_2$.
3. איטרציה: L^*.
4. היפוך: L^R.
5. חיתוך עם שפה רגולרית: $L \cap R$.
6. הומומורפיזם: $h(L)$.
7. הומומורפיזם הפוך: $h^{-1}(L)$.
8. הצבה חסרת הקשר.

את תכונות 6-8 לא נסביר ולא נשתמש בהן.

הערה חשובה: יכול להיות מצב שהפעולה הפכה שפה ח"ה לשפה רגולרית. זה תקין, כי כל שפה רגולרית היא גם ח"ה. אפשר לעלות בדרגות ההיררכיה של חומסקי, לא לרדת.

5.1 איחוד

בהינתן L_1, L_2 שפות ח"ה, קיימים להן דקדוקים ח"ה $G_1 = (V_1, T_1, S_1, P_1),\ \ G_2 = (V_2, T_2, S_2, P_2)$. אזי הדקדוק הבא מתאים לשפה $L_1 \cup L_2$:

$$G_{1\cup 2} = (V_1 \cup V_2 \cup \{S\}, \qquad T_1 \cup T_2, \qquad S, \qquad P_1 \cup P_2 \cup \{S \to S_1 | S_2\})$$

זה בהנחה כי $V_1 \cap V_2 = \emptyset$, וגם $S \notin V_1 \cup V_2$. אחרת צריך להחליף שמות, אבל הרעיון זהה.

הוכחה על רגל אחת: בצעד הגזירה הראשון נבחר האם אנחנו רוצים מילה מ-L_1 או מ-L_2.

הוכחה על שתי רגליים:

כיוון ראשון: תהי $w \in L_1 \cup L_2$, יש להראות כי $w \in L(G_{1\cup 2})$. מתקיים:

$w \in L_1$ ואז $S_1 \Rightarrow^* w$, או $w \in L_2$ ואז $S_2 \Rightarrow^* w$.

כיוון שני: תהי $w \in L(G_{1\cup 2})$. היות וכל המילים הנגזרות ב-$G_{1\cup 2}$ הן מהצורה $S \to S_1 \Rightarrow^* w$ או $S \to S_2 \Rightarrow^* w$, אזי $w \in L_1$ או $w \in L_2$.

5.2 שרשור

בהינתן L_1, L_2 שפות ח"ה, קיימים להן דקדוקים ח"ה $G_1 = (V_1, T_1, S_1, P_1),\ G_2 = (V_2, T_2, S_2, P_2)$. אזי הדקדוק הבא מתאים לשפה $L_1 \cdot L_2$:

$$G_{1\cdot 2} = (V_1 \cup V_2 \cup \{S\}, \qquad T_1 \cup T_2, \qquad S, \qquad P_1 \cup P_2 \cup \{S \to S_1 S_2\})$$

הוכחה על רגל אחת: בצעד הגזירה הראשון נגזור למשתנה ההתחלתי של L_1 וגם למשתנה ההתחלתי של L_2, וכך נקבל מילה מ-L_1 משורשרת עם מילה מ-L_2.

5.3 איטרציה

בהינתן L שפה ח"ה, קיים לה דקדוק ח"ה $G = (V_1, T_1, S_1, P)$. ניצור דקדוק עבור L^*:

$$G_* = (V_1 \cup \{S\}, \qquad T_1, \qquad S, \qquad P \cup \{S \to S_1 S \mid \epsilon\})$$

הוכחה על רגל אחת: בצעד הגזירה הראשון נאפשר לקבל חזרה את S כדי לגזור מילה נוספת מ-L_1.ושיימאס לנו נבחר ב-ϵ ונעצור. וכמובן, אפשר לבחור בו מלכתחילה, שכן איטרציה מכילה את אפסילון.

5.4 אי סגירות לחיתוך

בהינתן L_1, L_2 שפות ח"ה, השפה $L_1 \cap L_2$ לא בהכרח חסרת הקשר.

הוכחה: תהי $L_1 = \{a^n b^n c^m : n, m \geq 1\} \cap \{a^n b^m c^m : n, m \geq 1\}$. כל תת שפה הינה חסרת הקשר (ניתן לבנות דח"ה או אוטומט מחסנית מתאים) אך החיתוך ביניהן הוא $L_\cap = \{a^n b^n c^n : n \geq 1\}$ שזו שפה תלוית הקשר (נכשלת בלמת הניפוח).

מסקנה מזה – אי סגירות למשלים (כי אחרת, הייתה סגירות לחיתוך, לפי כללי דה מורגן).

5.5 היפוך

ניעזר בצורה הנורמלית של חומסקי.

תהי L ח"ה ויהי $G = (V, T, P, S)$ כך ש $L = L(G)$. אזי הדקדוק הבא יוצר את L^R:

$$G^R = (V, T, P', S)$$

$$P' \coloneqq \{A \to CB : A \to BC \in P\} \cup \{A \to \sigma | A \to \sigma \in P\}$$

הוכחה על רגל אחת: הפכנו את סדר הגזירה של כל המשתנים, ולכן כל המילים יוצאות הפוכות.

5.6 חיתוך עם שפה רגולרית

ההוכחה מסתמכת על אוטומט מחסנית, ומשתמשת באוטומט מכפלה.

יהי M_L א"מ עבור L, ויהי A_L אס"ד עבור R. נבנה אוטומט מכפלה ביניהם. נתקדם במקביל בשני האוטומטים. זה טיפה יותר מסובך כי יש מסעי אפסילון (שבהם לא מתקדמים ב-A_L), ויש אי-דטרמיניזם. לכן, נתקדם בא"מ, ושם נגדיר את מצב הבקרה, מ-A_L.

6 תרגילים

הוכיחו/הפריכו בעזרת תכונות סגור ולמת הניפוח: בהינתן $L_1 = \{a^n b^n : n \geq 1\}, L_2 = \{a^{2n} b^{3n} : n \geq 2\}$ שפות ח"ה, גם השפות הבאות ח"ה:

ראשית, נראה שגם L_2 ח"ה: ניצור דקדוק ח"ה עבורה: $S \to aaSbbb \mid \epsilon$.

$L_3 = \{a^n b^n a^{2n} b^{3n} : n \geq 1\}$. לכאורה נראה שרשור של $L_1 L_2$, אבל נשים לב שהדרישה פה קשה יותר: כי זה אותו n. זאת תהיה הפרכה.

$L_4 = \{a^n b^n a^n b^n : n \geq 1\}$. גם פה לכאורה שרשור $L_1 L_1$, אבל זה אותו n. גם הפרכה.

$L_5 = \{a^n b^n a^{2k} b^{3k} : n, k \geq 1\}$. זה אכן שרשור.

$L_6 = \{a^n b^n \cup a^{2n} b^{3n} : n \geq 1\}$. למרות שזה אותו n, זה איחוד. כי כל פעם בוחרים מילה מכאן <u>או</u> מכאן.

$L_7 = \{d^*\} \cup \{a^n b^n c^n : n \geq 1\}$. נב"ש שהיא ח"ה, כלומר $L_7 \cap \{a^* b^* c^*\}$ גם ח"ה (חיתוך עם שפה רגולרית). אבל החיתוך הוא $a^n b^n c^n$, שהיא לא ח"ה (ראינו מקודם).

הפרכה עבור L_3: נראה באמצעות למת הניפוח שהיא לא ח"ה:

נב"ש שהיא ל"ה ויהי n הקבוע המובטח בלמה. נבחר $z = a^n b^n a^{2n} b^{3n} \in L_3$, המילה ארוכה מספיק ולכן קיים פירוק לפי הלמה. אפשרויות לפירוק: $a \ldots_1 a \;^5\; b \ldots_2 b \;^6\; a \ldots_3 a \;^7\; b \ldots_4 b$.

עבור מקרה 1: נבחר $i = 2$ ונקבל: $a^{n+|vx|} b^n a^{2n} b^{3n} \notin L_3$. מקרים 2,3,4 דומים.

עבור מקרה 5 נבחר $i = 0$ ונקבל: $a^{n-r} b^{n-k} a^{2n} b^{3n} \notin L_3$ (כי $1 \leq r$ או $1 \leq k$).

לכל מקרה מצאנו i שעבורו המילה המנופחת לא בשפה, בסתירה ללמת הניפוח, כלומר השפה אינה ח"ה.

הפרכה עבור L_4 עובדת בצורה דומה. השוויון של ה-n מחייב זיכרון שאפילו אוטומט מחסנית לא מספק.

הרצאה 10

תזכורת: כאשר דיברנו על ההיררכיה של חומסקי, אמרנו כי לכל מודל דקדוקי יש מודל תיאורטי ששקול בכוחו לדקדוק. ראינו שדקדוק לינארי ימני/שמאלי שקול לשפות רגולריות (אס"ד).

אמרנו שדקדוק חסר הקשר שקול לאוטומט מחסנית (א"מ). דקדוק תלוי הקשר שקול למכונת טיורינג עם הגבלת זיכרון. דקדוק כללי שקול למכונת טיורינג ללא הגבלה – מחשב. כעת נכיר אוטומט מחסנית.

1 אוטומט מחסנית

לעומת אס"ד, שהזיכרון שלו חסום (כמות המצבים סופית), באוטומט מחסנית יש כלי נוסף – זיכרון מחסנית "אינסופי".

1.1 הגדרות

- באס"ד, פונקציית המעברים δ הייתה פונקציה של Q, Σ. בא"מ, הפונקציה δ תלויה גם בתו העליון שיש במחסנית. כלומר בכל צעד עושים pop למחסנית, והתו הזה משפיע על הפונקציה. אם המחסנית ריקה, אי אפשר להתקדם.
- על מנת להבדיל בין אותיות הקלט לאותיות שיש במחסנית, בד"כ נסמן את אותיות הקלט בא"ב קטן $(a, b, \dots, z)$ ואת האותיות בזיכרון המחסנית בא"ב גדול $(A, B, \dots, Z)$.
- על מנת להבדיל בין **המילים** של הקלט ו**המילים** במחסנית, בד"כ נסמן את מילות הקלט באותיות לטיניות (באנגלית) – x, y, z, ואת המילים במחסנית ב $\alpha, \beta, \dots, \gamma$.
- תחתית המחסנית תסומן על ידי תו מיוחד: $\perp$, שאינו חלק מהא"ב.

1.2 הגדרת האוטומט

אוטומט מחסנית הוא שביעייה: $A = (Q, \Sigma, q_0, \delta, F, \Gamma, \perp)$.

הסימן Γ הוא הא"ב של המחסנית עצמה, ובד"כ לא נגביל אותו.

נגדיר רשמית את פונקציית המעברים:

$$\delta: Q \times (\Sigma, \epsilon) \times \Gamma \to 2^{Q \times \Gamma^*}$$

קלט: מצב, אות קלט (או אפסילון), ואת ראש המחסנית.

פלט: זוג סדור: מצב, ו"מה לכתוב בראש המחסנית".

למה זוגות? כי אוטומט המחסנית יהיה אי דטרמיניסטי.

כמה אפשר לכתוב בראש המחסנית? כמות סופית כלשהי. נרשום כך שהתו הימני ביותר של המחרוזת הוא זה שנכנס ראשון למחסנית.

1.3 קבלה של אוטומט מחסנית

נגדיר שתי אפשרויות לקבלת מילה באוטומט מחסנית:

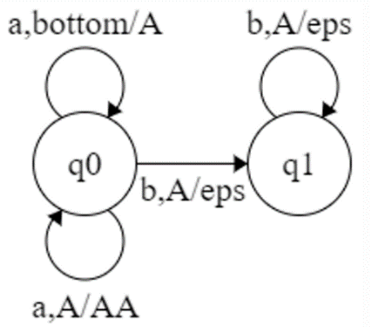
1. קבלה על ידי **מצב מקבל**. כלומר, אם האוטומט יגיע למצב ששייך ל-F והמילה נגמרה, אז המילה מתקבלת באוטומט. זו הדרך שאנחנו רגילים מהאוטומטים בתחילת הקורס.
2. קבלה על ידי **ריקון המחסנית**. כלומר, אם המחסנית התרוקנה וגם המילה נגמרה, אזי המילה מתקבלת באוטומט. ללא תלות במצבי האוטומט.

בהמשך נוכיח שקילות בין שני אופני הקבלה.

1.4 דוגמה

נבנה אוטומט מחסנית (המקבל ע"י ריקון) לשפה $L = \{a^n b^n : n \geq 1\}$.

נצטרך מצב שבו מוסיפים אות למחסנית, ומצב שבו מרוקנים אות מהמחסנית. וכך נוכל להשוות בין כמות ה-a לכמות ה-b.



אם היינו ב-q_0, וקראנו a, ובמחסנית עשינו pop וקיבלנו את התו שמסמל את תחתית המחסנית – אז במצבים נישאר באותו מצב, ונכניס A למחסנית. (זה יקרה רק פעם אחת, בקריאת התו הראשון במילה).

אם היינו ב-q_0, וקראנו a, ובמחסנית קיבלנו A – אז נישאר באותו מצב, ונכניס AA למחסנית.

אם היינו ב-q_0, וקראנו b, ובמחסנית קיבלנו A – אז נעבור למצב q_1, ונכניס ϵ למחסנית.

אם היינו ב-q_1, וקראנו b, ובמחסנית קיבלנו A – אז נישאר באותו מצב, ונכניס ϵ למחסנית.

פורמלית, נגדיר:

$$M = (\{q_0, q_1\}, \{a, b\}, \{A, \perp\}, q_0, \perp, \phi, \delta)$$

$$\begin{aligned} \delta(q_0, a, \perp) &= (q_0, A) \\ \delta(q_0, a, A) &= (q_0, AA) \\ \delta(q_0, b, A) &= (q_1, \epsilon) \\ \delta(q_1, b, A) &= (q_1, \epsilon) \end{aligned}$$

1.5 תיאור רגעי

עבור א"מ M, **תיאור רגעי** (*instantaneous Description - ID*) הוא שלישיה: (q, w, γ), כאשר:

- $q \in Q$ הוא מצב הבקרה הנוכחי,
- $w \in \Sigma^*$ היא יתרת הקלט,
- $\gamma \in \Gamma^*$ זה תוכן המחסנית.

בא"מ שבנינו קודם, עבור הקלט $aaabbb$, לאחר שני צעדים התיאור הרגעי הוא $(q_0, abbb, AA)$. לאחר שלושה צעדים התיאור הרגעי הוא (q_0, bbb, AAA).

כעת נוכל לתאר את חישובי האוטומט כרצף צעדים בין ID עוקבים.

1.6 תיאור רגעי עוקב

נאמר כי $ID_2 = (p, w, \beta\gamma)$ **עוקב ל-** $ID_1 = (q, \sigma w, Z\gamma)$ אמ"מ $(p, \beta) \in \delta(q, \sigma, Z)$.

במילים – אם ממצב q ע"י קריאת הקלט σ וראש המחסנית Z ניתן להגיע למצב p ולרשום במחסנית β. (ואז נשאר לקרוא בקלט רק את המילה w). כלומר, היינו במצב q, הקלט שנשאר הוא σw (וקראנו σ), והמחסנית הייתה $Z\gamma$ (Z בראש, ושאר המחסנית היא המילה γ). קראנו את σ וקיבלנו מהמחסנית Z, ועברנו למצב p, ועכשיו במחסנית יש $\beta\gamma$ – כלומר הכנסנו את β.

נסמן יחס זה כך: $(q, \sigma w, Z\gamma) \vdash_M (p, w, \beta\gamma)$.

וכפי שכבר התרגלנו, הסימון $\vdash_M^*$ מתאר סדרת צעדי חישוב. כלומר:

$$(q, w, \gamma) \vdash_M^* (p, x, \beta)$$

אמ"מ קיימת סדרת תיאורים רגעיים עוקבים המתחילה ב (q, w, γ) ומסתיימת ב (p, x, β).

כרגיל, ניעזר גם ב $\vdash^k$, $\vdash^+$. (כן, יהיו פה אינדוקציות. בהצלחה).

2 מודי קבלה

כעת נוכל להגדיר באופן רשמי את שני מודי הקבלה של א"מ. יהי אוטומט מחסנית: $M = (Q, \Sigma, \Gamma, q_0, \delta, F, \perp)$.

2.1 הגדרה פורמלית של מודי הקבלה

השפה המתקבלת על ידי **הגעה למצב מקבל** מסומנת: $L_f(M)$, ומוגדרת:

$$L_f(M) = \{w \in \Sigma^* : \exists p \in F, \gamma \in \Gamma^* \ s.t{:}\ (q_0, w, \perp) \vdash_M^* (p, \epsilon, \gamma)\}$$

קבוצת המילים כך ש: קיים מצב מקבל, ומחרוזת במחסנית, כך ש: אם היינו במצב ההתחלתי ועם מחסנית ריקה, וקראנו את המילה, כשהמילה נגמרה היינו במצב מקבל. לא מעניין אותנו מה יש במחסנית.

השפה המתקבלת על ידי **ריקון המחסנית** מסומנת $L_\epsilon(M)$, ומוגדרת:

$$L_\epsilon(M) = \{w \in \Sigma^* : (q_0, w, \perp) \vdash_M^* (p, \epsilon, \epsilon)\}$$

קבוצת המילים כך ש: אם היינו במצב ההתחלתי ועם מחסנית ריקה, וקראנו את המילה, כשהמילה נגמרה המחסנית הייתה ריקה. לא מעניין אותנו לאיזה מצב הגענו.

2.2 שקילות שני אופני הקבלה

נוכיח כי משפחת השפות המתקבלות ע"י א"מ שמקבל ע"י ריקון, זהה למשפחת השפות המתקבלות ע"י א"מ המקבל ע"י מצב מקבל.

נשים לב שההוכחה טענת כי בהינתן א"מ מצורה אחת, ניתן לבנות א"מ מהצורה השנייה. **היא לא טוענת** ש $L_f(M) = L_\epsilon(M)$.

לדוגמה, יהי $M = (\{q_0\}, \{a\}, \{\perp, A\}, \delta, q_0, \perp \ \{q_0\})$, כאשר $\delta(q_0, a, \perp) = (q_0, A \perp), \delta(q_0, a, A) = (q_0, AA)$. אזי: $L_f(M) = a^*$, $L_\epsilon(M) = \phi$. כי יש רק מצב אחד והוא מקבל, אז כל מילה מתקבלת. אבל האוטומט אף פעם לא מתרוקן.

משפט: $L_f(M) \equiv L_\epsilon(M)$.

2.3 כיוון ראשון - בניית אוטומט ריקון בהינתן אוטומט מצב

בהינתן $L_f(M)$, נראה שניתן לבנות M' מ- L_ϵ.

הטענה הפורמלית: תהי M_1 שעבורה $L = L_f(M_1)$, ויהי $M_1 = (Q, \Sigma, \Gamma, q_0, \delta, F, \perp)$. אזי קיים א"מ M_2 כך ש $L = L_\epsilon(M_2)$.

רעיון הבנייה: M_2 יתנהג כמו M_1, אך כאשר M_1 ייכנס למצב מקבל, נכניס את M_2 למצב שמרוקן את המחסנית.

נצטרך לזהות מתי M_1 רוקנה את המחסנית. לכן נשתמש ב"תחתית כפולה". (כי אם δ קוראת ϵ מראש המחסנית, האוטומט נתקע).

לכן, בתחילת הריצה M_2 תכיל את התחתית $\perp_2$, ובצעד הראשון היא תכניס את $\perp_1$. לכן, יהיה גם q_0' שיבצע את ההכנסה הזו, לפני הרצת M_1.

הגדרת M_2: נקבל את הא"מ הבא:

$$M_2 = (Q \cup \{q_e, q_0'\}, \Sigma, \Gamma \cup \{\perp_2\}, \delta', q_0', \perp_2, \phi)$$

כאשר $q_e, q_0' \notin Q$ וגם $\perp_2 \notin \Gamma$. נגדיר את δ':

$$\delta'(q_0', \epsilon, \perp_2) = \{(q_0, \perp_1 \perp_2)\}$$

אם $q \notin F$ או $\sigma \neq \epsilon$ אזי M_2 מתנהג כמו M_1, כלומר לכל $Z \in \Gamma$:

$$\delta'(q, \sigma, Z) = \delta(q, \sigma, Z)$$

אם $q \in F$, אזי: לכל $Z \in \Gamma \cup \{\perp_2\}$ נגדיר:

$$\delta'(q,\epsilon,Z) = \delta(q,\epsilon,Z) \cup \{(q_e,\epsilon)\}$$

כלומר, אפשר להתקדם רגיל כמו ב-M_1 ואפשר ללכת למצב שבו מרוקנים את כל המחסנית.

לכל $Z \in \Gamma \cup \{\perp_2\}$ נגדיר: $\delta(q_e,\epsilon,Z) = (q_e,\epsilon)$ (נשארים במקום ומרוקנים).

נוכיח כי $L_f(M_1) = L_\epsilon(M_2)$, ע"י הכלה דו כיוונית:

2.3.1 כיוון ראשון: $L_f(M_1) \subseteq L_\epsilon(M_2)$

תהי $x \in L_f(M_1)$. כלומר, קיים חישוב $(q_0,x,\perp_1) \vdash^* (p,\epsilon,\gamma)$ עבור $p \in F, \gamma \in \Gamma^*$.

נמשיך את המסלול של x עד שיתקבל גם ב-M_2, כך:

ע"פ כלל א' (של δ): $(q_0',\epsilon,\perp_2) \vdash_{M_2} (q_0,x,\perp_1\perp_2)$

ע"פ כלל ב': M_2 מתנהגת כמו M_1 בכל קריאת המילה x, ולכן:

$$(q_0,x,\perp_1) \vdash^*_{M_2} (p,\epsilon,\gamma\perp_2)$$

לכן, נעבור משם למצב q_e בעזרת כלל ג' (ריקון אות) ונמשיך לרוקן עם כלל ד':

$$(p,\epsilon,\gamma\perp_2) \vdash^*_{M_2} (q_e,\epsilon,\epsilon)$$

לסיכום, מתקיים כי:

$$(q_0',x,\perp_2) \vdash^*_{M_2} (q_e,\epsilon,\epsilon)$$

כלומר, $x \in L_\epsilon(M_2)$.

2.3.2 כיוון שני: $L_\epsilon(M_1) \subseteq L_f(M_2)$

תהי $x \in L_\epsilon(M_1)$. כלומר, בקריאת x המחסנית התרוקנה לחלוטין.

הדרך היחידה לעשות את זה היא ע"י הגעה למצב q_e (שכן שאר צעדי M_1 לא יכולים לרוקן את M_2).

לכן, סדרת הצעדים ש M_2 מבצע על x, (פרט לצעד הראשון), קיימת גם ב M_1. ולכן $x \in L_f(M_1)$.

לסיכום, $L_f(M_1) = L_\epsilon(M_2)$. כנדרש.

2.4 כיוון שני – בניית אוטומט מצב בהינתן אוטומט ריקון

בהינתן $L_\epsilon(M)$, נראה שניתן לבנות M' מ- L_f.

רעיון הבנייה: כאשר הא"מ הראשון ירוקן את המחסנית, נבצע צעד אפסילון למצב מקבל (יחיד) ונרוקן את התחתית הנוספת (כי כל צעד מבצע pop לראש המחסנית, ואם היא ריקה האוטומט ייתקע).

תהי $L = L_\epsilon(M_2)$ עבור $M_2 = (Q,\Sigma,\Gamma,q_0,\perp_2,\delta,\phi)$. אזי קיים א"מ M_1 כך ש $L = L_f(M_1)$.

$$M_1 = \left(Q \cup \{q_f,q_0'\},\Sigma,\Gamma \cup \{\perp_1\},\perp_1,\delta',q_0',q_f\right)$$

$$\delta'(q_0',\epsilon,\perp_1) = (q_0,\perp_2\perp_1)$$

לכל $q \in Q, Z \in \Gamma, \sigma \in \Sigma \cup \{\epsilon\}$ נגדיר: $\delta'(q,\sigma,Z) = \delta(q,\sigma,Z)$.

לכל $q \in Q$: $\delta'(q,\epsilon,\perp_1) = \left(q_f,\epsilon\right)$ (הסבר: מכל מצב נוסיף מסע אפסילון למצב מקבל, במקרה שהמחסנית של M_2 התרוקנה).

2.5 הוכחת נכונות הבנייה

כיוון ראשון: $L_f(M_1) \subseteq L_\epsilon(M_2)$

תהי $x \in L_f(M_1)$. כלומר הגענו למצב q_f. איך?

הדרך היחידה להגיע אל q_f היא ע"י סיום ריקון המחסנית כבמסלול החישוב של M_2, ואז כשמגיעים לתחתית הנוספת $\perp_1$ עוברים למצב מקבל.

לכן, $x \in L_\epsilon(M_2)$, כי רוקנו את המחסנית המקורית של M_2.

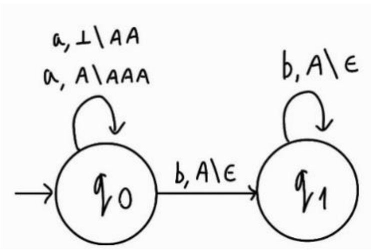
כיוון שני: $L_\epsilon(M_2) \subseteq L_f(M_1)$.

תהי $x \in L_\epsilon(M_2)$. אזי המחסנית של M_2 מתרוקנת בסיום קריאת המילה. לכן, היות והאוטומט M_1 רץ כמו M_2 על המילה (החל מהצעד השני), הוא גם ירוקן את המחסנית ויזהה את $\perp_1$ בתחתית, ומשם יעבור למצב מקבל, ולכן $x \in L_f(M_1)$, כנדרש.

2.6 דוגמאות

לכל שפה נבנה אוטומט ריקון ואוטומט מצב מקבל:

$$L_1 = \{a^n b^{2n} : 1 \le n\}$$

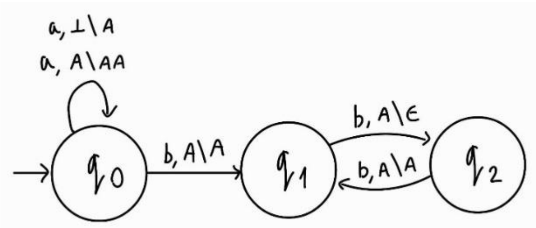


בקריאת האות הראשונה (בהנחה שזה a. אם זה b אז האוטומט נתקע), המחסנית תהיה ריקה. נכניס AA כי זה כמות ה-b שנרצה לראות.

בכל קריאה של a, מוציאים A מהמחסנית ונכניס AAA. ככה בעצם הוספנו AA למחסנית.

בקריאה הראשונה של b, מוציאים A ועוברים למצב השני.

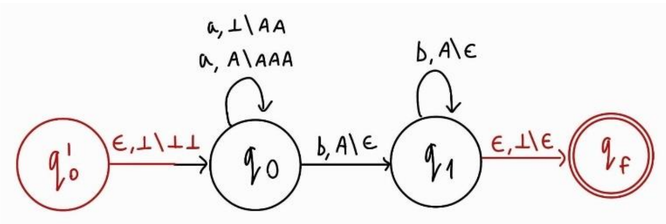
בכל קריאה אחרת של b, מוציאים A. אם המילה נגמרה והמחסנית ריקה, זה אומר שהיה בדיוק b^{2n}.



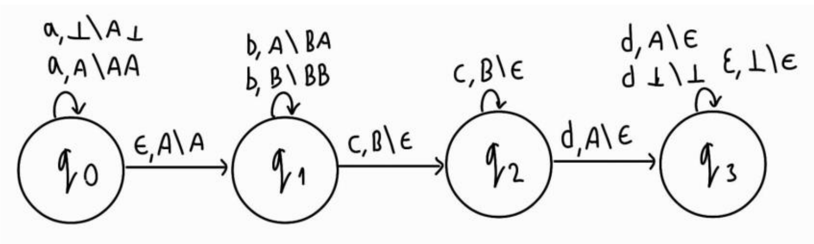
עוד אפשרות: כל a יוסיף A אחד, ונעשה שני מצבים של b, שאחד מהם מוריד A והשני לא. זה אותו אפקט:

איך נהפוך את זה לאוטומט מצב מקבל?

נוסיף מצב התחלתי, עם מסע אפסילון שבקריאת תחתית המחסנית מכניס שתי "תחתיות". ומהמצב האחרון, מסע אפסילון למצב מקבל. המסע אפסילון הזה קורה רק בקריאת תחתית המחסנית:



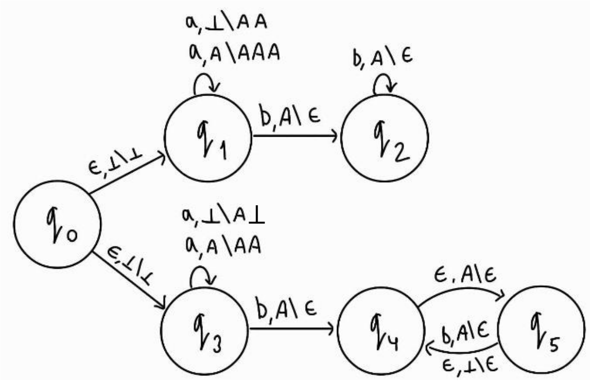
$$L_2 = \{a^l b^m c^m d^n : 1 \le l \le n, 1 \le m\}$$



נספור כמה a יש. נעבור למצב b. צריך להיות אותו מספר c כמו b. ואז צריך להיות d לפחות כמו a, ואפשר יותר. לכן המצב האחרון מאפשר לרוקן את המחסנית.

המעבר למצב 1 דורש A כי חייב לפחות a אחד. אם אין דרישה כזו, אז צריך להוסיף מסע אפסילון.

מעבר לאוטומט מצב מקבל יעבוד באותה דרך כמו בסעיף הקודם. נוסיף מצה התחלתי ומצב מקבל עם מסעי אפסילון.



$$L_3 = \{a^n b^{2n} \cup a^{2n} b^n : 1 \leq n\}$$

אוטומט לכל שפה, ומסע אפסילון שנותן לנו לבחור את אחת מהן.

באוטומט הראשון, נספור פעמיים את כמות ה-a ואז כל b יוציא אחד.

באוטומט השני, נספור כמה a יש. ואז כל b שני יוציא A אחד. וכשנגיע לסוף המחסנית, נרוקן אותה.

3 אוטומט מחסנית דטרמיניסטי

נאמר שאוטומט מחסנית הוא דטרמיניסטי אם הוא מקיים:

1. לכל $q \in Q, \sigma \in \Sigma, Z \in \Gamma$: $|\delta(q, \sigma, Z)| \leq 1$. כלומר מכל מצב, עבור אותה אות ותו מחסנית – יש רק חץ אחד. לדוגמה, האוטומט האחרון לא מקיים את זה, כי מq_0 יש שני חיצים עם אותם תנאים.

2. לכל $q \in Q, Z \in \Gamma$: אם $\delta(q, \epsilon, Z) \neq \phi$, אזי לכל $\sigma \in \Sigma$ מתקיים: $\delta(q, \sigma, Z) = \phi$.
כלומר: אם קיים מסע אפסילון עבור ראש מחסנית כלשהו, אז זה המסע היחיד עבור אותו ראש מחסנית.

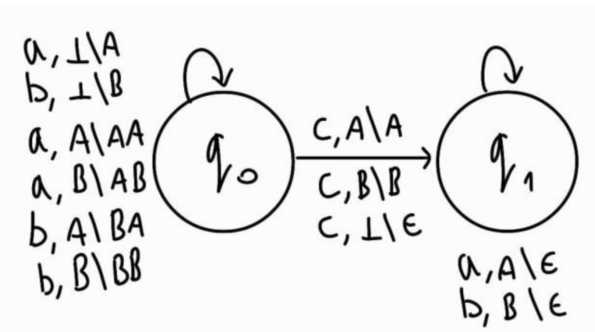
3.1 דוגמה 1

נבנה א"מ דטרמיניסטי המתקבל ע"י ריקון לשפה הבאה:

$$L_{mp} = \{wcw^R : w \in \{a, b\}^*\}, \Sigma = \{a, b, c\}$$

נצטרך מצב שבו קוראים את המילה w, ומצב שבו מרוקנים את המחסנית ומוודאים שאכן יש לנו w^R. המעבר ביניהם הוא ע"י קריאת c, בלי לגעת בראש המחסנית.

למחסנית נצטרך להכניס את המילה, ולכן $\Gamma = \{A, B, \perp\}$. צריך להגדיר את δ לכל תו קלט ולכל ראש מחסנית (ולשני המצבים).



הכנסת תו ראשון למחסנית:

$$\delta(q_0, a, \perp) = (q_0, A), \qquad \delta(q_0, b, \perp) = (q_0, B)$$

הכנסת המילה w למחסנית:

$$\forall \sigma \in \{a, b\}, \forall \gamma \in \{A, B\} : \ \delta(q_0, \sigma, \gamma) = (q_0, \sigma\gamma)$$

מעבר למצב השני, שבו מרוקנים ומוודאים שהמילה הנוספת היא אכן w^R: $\forall \gamma \in \{a, b\} : \delta(q_0, c, \gamma) = (q_1, \gamma)$

ריקון המחסנית: $\delta(q_1, a, A) = (q_1, \epsilon), \ \delta(q_1, b, B) = (q_1, \epsilon)$

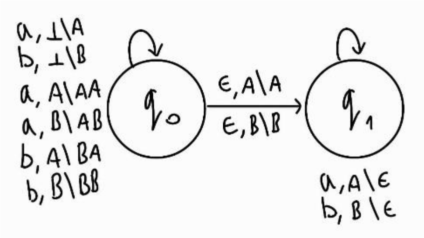
טיפול במצב שבו $w = \epsilon$: $\delta(q_0, c, \perp) = (q_1, \epsilon)$

ניתן להוכיח באינדוקציה כי $L_\epsilon(M) = L$ (כמובן).

3.2 דוגמה 2

נבנה א"מ א"ד המתקבל ע"י ריקון, לשפה $L_{nmp} = \{ww^R : w \in \{a, b\}^*\}$.

כאן אין את התו c שמכריע אם הגענו לסוף w, אלא נצטרך לנחש בצורה אי דטרמיניסטית. וניעזר במחסנית בשביל לוודא שיש את w^R.



הכנסת תו ראשון למחסנית: $\delta(q_0,\sigma,\perp)=(q_0,\sigma)$

הכנסת המילה w: $\forall\sigma\in\{a,b\},\forall\gamma\in\{a,b\}:\ \delta(q_0,\sigma,\gamma)=(q_0,\sigma\gamma)$

מעבר אי דטרמיניסטי למצב הבא: $\delta(q_0,a,a)=(q_1,\epsilon),\ \delta(q_0,b,b)=(q_1,\epsilon)$

ריקון המחסנית כל עוד המילה באמת הפוכה:

$$\delta(q_1,a,a)=(q_1,\epsilon),\qquad \delta(q_1,b,b)=(q_1,\epsilon)$$

טיפול במצב שבו $w=\epsilon$: $\delta(q_0,\epsilon,\perp)=(q_0,\epsilon)$

למה פה לא עוברים ל-q_1? זה לא קריטי. העיקר הוא שאם ראש המחסנית הוא ϵ, לא ניתן להתקדם, ואז לא משנה איפה אנחנו.

4 שקילות בין א"מ לדח"ה

במודל הרגולרי, ראינו שאי-דטרמיניזם לא מוסיף כוח (כי האוטומטים שקולים). לעומת זאת, במודל ח"ה, ראינו שפה שלא ניתן לבדוק באוטומט מחסנית דטרמיניסטי, אלא רק בא"מ א"ד. כלומר הוא מוסיף כוח. ונצטרך אותו כשנראה שקילות לדח"ה.

מוטיבציה: לפעמים יהיה נוח יותר לבנות אחד ולא את השני. לדוגמה, בשפות תכנות נוח להשתמש בדח"ה. א"מ נוח לתיאור של אלגוריתם או מודל חישובי שבודק תחביר של שפה.

נראה כיוון אחד של הוכחת השקילות: בהינתן דקדוק ח"ה, נבנה אוטומט מחסנית שקול.

4.1 רעיון ההוכחה

טענה: לכל שפה ח"ה L, קיים M כך ש $L=L_\epsilon(M)$.

לכל שפה ח"ה קיים דקדוק ח"ה שיוצר אותה (לפי הגדרה).

נבנה א"מ שיחקה **גזירה שמאלית ביותר** בדקדוק. גזיר שמאלית ביותר – אם יש לנו רצף של משתנים, תמיד נגזור את השמאלי ביותר. אם גזרנו וקיבלנו עוד משתנים – גם הפעם, נגזור את השמאלי ביותר לפני שנמשיך. (מבנה קצת רקורסיבי).

התבנית הפסוקית הנוכחית תיוצג ע"י שרשור רישא הקלט שהאוטומט קרא, יחד עם תוכן המחסנית (ב-ID השארנו את מה שנותר לקרוא, ואילו כאן יעניין אותנו מה שכבר קראנו).

המטרה: לסיים את מילת הקלט ואת תוכן המחסנית בבת אחת (L_ϵ).

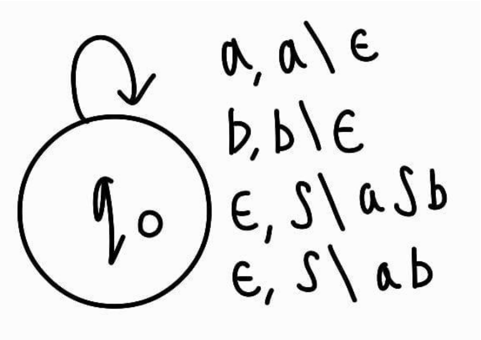
האוטומט יפעיל את האלגוריתם הבא:

4.2 אלגוריתם לאוטומט המחסנית

1. נאתחל מחסנית בסימן יחיד S.
2. אם בראש המחסנית מופיע המשתנה הדקדוקי $A\in V$,
נבחר (באופן אי דטרמיניסטי) באחד מכללי הגזירה $A\to\alpha$,
ונכניס למחסנית את α. (מסעי אפסילון).
3. אם בראש המחסנית מופיע σ וגם אות הקלט הבאה היא σ, נקרא את האות ונוציא את σ מראש המחסנית.
4. אם המילה לא נגמרה, נחזור לשלב 2.

שימו לב: אין כאן יצירת מצבים. כלומר, נשארנו בכל הריצה ב-q_0.

4.3 דוגמת הרצה



תהי $L = \{a^n b^n : n \ge 1\}$. ויהי $G = (\{a,b\},\{S\},S,S \to aSb \mid ab)$ הדקדוק שיוצר אותה. נדגים ריצת א"מ מתאים:

$$\delta(q_0,\epsilon,S) = \{(q_0,aSb),(q_0,ab)\}$$
$$\delta(q_0,a,a) = (q_0,\epsilon) = \delta(q_0,b,b)$$

נתבונן בגזירה: $S \to aSb \to aaSbb \to aaabbb$. להלן החישוב של האוטומט:

$$\begin{aligned}(q_0,aaabbb,S) &\vdash (q_0,aaabbb,aSb) \vdash (q_0,aabbb,Sb)\\ &\vdash (q_0,aabbb,aSbb) \vdash (q_0,abbb,Sbb)\\ &\vdash (q_0,abbb,abbb) \vdash_M^4 (q_0,\epsilon,\epsilon)\end{aligned}$$

כאשר ארבעת הצעדים האחרונים הם $(q_0,\sigma w,\sigma w) \vdash (q_0,w,w)$

4.4 בנייה פורמלית

יהי $G = (V,T,S,P)$ כך ש $L = L(G)$. נגדיר א"מ כך:

$$M = (\{q_0\},T,V \cup T,\delta,q_0,S,\phi)$$

כאשר δ מוגדרת לפי כללי P:

לכל $A \in V$: $\delta(q_0,\epsilon,A) = \{(q_0,\alpha) : A \to \alpha \in P\}$.

לכל $a \in T$, $\delta(q_0,a,a) = (q_0,\epsilon)$.

טענה: לכל $x \in T^*$, ולכל $\alpha \in (V \cup T)^*$ (שאינה מתחילה בטרמינל):

$S \Rightarrow^* x\alpha$ בגזירה שמאלית ביותר **אמ"מ** $(q_0,x,S) \vdash^* (q_0,\epsilon,\alpha)$.

כלומר: אם בדקדוק הצלחנו לשכתב מ-S את $x\alpha$, אז גם באוטומט: אם התחלנו מהמצב ההתחלתי, עם S במחסנית, אחרי קריאת x יהיה לנו α במחסנית.

למה זה עוזר? כי עבור $\alpha = \epsilon$ נקבל $x \in L(G) \Leftrightarrow x \in L(M)$. נוכיח בהכלה דו כיוונית:

4.5 כיוון ראשון

טענה: לכל $x \in T^*$ ולכל $\alpha \in (V \cup T)^*$, אם $(q_0,x,S) \vdash^* (q_0,\epsilon,\alpha)$ אזי $S \Rightarrow^* x\alpha$ בגזירה שמאלית ביותר.

הוכחה באינדוקציה על i: מספר צעדי החישוב.

בסיס: עבור $i = 0$, בהכרח $x = \epsilon, \alpha = S$. ואכן $S \Rightarrow^0 S = x\alpha$.

צעד: נניח כי הטענה נכונה לכל חישוב שאורכו קטן מ-i, וכי $(q_0,x,S) \vdash_M^i (q_0,\epsilon,\alpha)$.

כלומר אחרי i צעדים, סיימנו את x ונשאר α במחסנית. מה היה הצעד האחרון?

יש שני סוגים של צעדים באוטומט: מסע אפסילון שמחליף משתנה (הפעלת כלל גזירה), או מסע של ריקון טרמינל מהקלט.

מקרה א – המסע האחרון מחליף משתנה: היה במחסנית $A\alpha'$, קראנו את A, אולי הכנסנו משהו למחסנית ועכשיו יש במחסנית α. מהגדרה, זה מסע אפסילון. וזה הצעד האחרון, כלומר כבר סיימנו לקרוא את x: $(q_0,x,S) \vdash_M^{i-1} (q_0,\epsilon,A\alpha') \vdash_M (q_0,\epsilon,\alpha)$.

מהנ"א, $S \Rightarrow^* xA\alpha'$ בגזירה שמאלית ביותר, ומהגדרת M מתקיים $A\alpha' \Rightarrow \kappa\alpha' = \alpha$, כאשר $A \to \kappa \in P$. מכאן $S \Rightarrow^* x\alpha$ בגזירה שמאלית ביותר, כנדרש.

מקרה ב – המסע האחרון הוא ריקון טרמינל מהקלט: אחרי $i-1$ צעדים נשאר לקרוא σ, ובראש המחסנית יש σ ומתחתיו α.

הצעד האחרון (בקריאת σ) מוחק את σ מראש המחסנית: $(q_0,x,S)\vdash_M^{i-1}(q_0,\sigma,\sigma\alpha)\vdash_M(q_0,\epsilon,\alpha)$ כאשר $x=x'\sigma$.

לכן נוכל לרשום $(q_0,x',S)\vdash_M^{i-1}(q_0,\epsilon,\sigma\alpha)$. מהנ"א נקבל $S\Rightarrow^* x'\sigma\alpha$ בגזירה שמאלית ביותר, (וכמובן $x=x'\sigma$), כנדרש.

4.6 כיוון שני

טענה: לכל $x\in T^*$ ולכל $\alpha\in(V\cup T)^*$ שאינה מתחילה בטרמינל, אם $S\Rightarrow^* x\alpha$ בגזירה שמאלית ביותר אזי $(q_0,x,S)\vdash^*(q_0,\epsilon,\alpha)$. הוכחה באינדוקציה על אורך הגזירה, i.

בסיס: עבור $i=0$, בהכרח $x=\epsilon,\alpha=S$. בוודאי מתקיים $(q_0,\epsilon,S)\vdash_M^0(q_0,\epsilon,S)$.

צעד: נניח כי הטענה נכונה לכל גזירה עד אורך $i-1$, ויהי $S\Rightarrow^i x\alpha$ בגזירה שמאלית ביותר, כאשר $x\in T^*$ ו- $\alpha\in(V\cup T)^*$ שאינה מתחילה בטרמינל.

היות והגזירה שמאלית ביותר (בכלל פעם נטפל במשתנה הכי שמאלי), ניתן לרשום $S\Rightarrow^{i-1}x'Aw\alpha'\Rightarrow x\alpha$, כאשר α' אינה מתחילה בטרמינל (הנ"א) וכלל הגזירה האחרון שהופעל הוא $A\to\ddot{x}\ddot{\alpha}$. ($x'$ מורכבת רק מטרמינלים).

נחלק בין מקרים: אם A שכתבה רק טרמינלים, אז $x'\ddot{x}\ddot{\alpha}w\alpha'$ היא תת מחרוזת של x. כלומר $\ddot{\alpha}=\epsilon$.

אחרת, $\ddot{\alpha}\neq\epsilon$. כלומר $\ddot{x}\ddot{\alpha}$ מכילה טרמינלים. ואז $x'\ddot{x}$ היא תת מחרוזת של x, $\ddot{\alpha}w\alpha'$ היא תת מחרוזת של α.

מקרה א: $\ddot{\alpha}=\epsilon$:

במקרה זה נקבל כי $\alpha=\alpha'$ וכי $x=x'\ddot{x}w$. מהנ"א נקבל $(q_0,x',S)\vdash_M^*(q_0,\epsilon,Aw\alpha')$ ולכן:

$$(q_0,x,S)=(q_0,x'\ddot{x}w,S)\vdash_M^*(q_0,\ddot{x}w,Aw\alpha')$$

שוב, היות והכלל המופעל הוא $A\to\ddot{x}$, ניתן להמשיך כך:

$$(q_0,\ddot{x}w,Aw\alpha')\vdash_M(q_0,\ddot{x}w,\ddot{x}w\alpha')\vdash(q_0,\epsilon,\alpha')$$

ולכן קיים החישוב:

$$(q_0,x,S)\vdash_M^*(q_0,\epsilon,\alpha)$$

כנדרש.

מקרה ב: $\ddot{\alpha}\neq\epsilon$:

במקרה זה נקבל כי $\alpha=\ddot{\alpha}w\alpha'$ וכי $x=x'\ddot{x}$. בנוסף, מהנ"א נקבל $(q_0,x',S)\vdash_M^*(q_0,\epsilon,Aw\alpha')$ ולכן:

$$(q_0,x,S)=(q_0,x'\ddot{x},S)\vdash_M^*(q_0,\ddot{x},Aw\alpha')$$

היות והכלל המופעל הוא $A\to\ddot{x}\ddot{\alpha}$, ניתן להמשיך כך:

$$(q_0,\ddot{x},Aw\alpha')\vdash_M(q_0,\ddot{x},\ddot{x}\ddot{\alpha}w\alpha')\vdash_M^*(q_0,\epsilon,\ddot{\alpha}w\alpha')$$

ולכן קיים החישוב:

$$(q_0,x,S)\vdash_M^*(q_0,\epsilon,\alpha)$$

כנדרש.

עד כאן אוטומטים, ברוך שפטרנו. ■