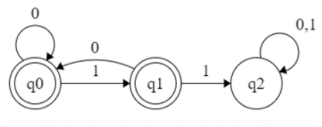
הרצאה 2

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 $\epsilon \notin 11$).

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:

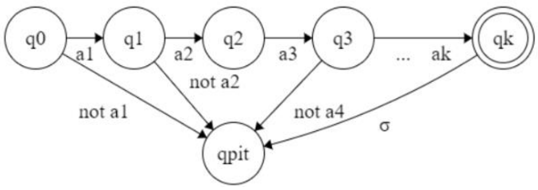
- האם קיימת שפה שאס"ד אינו מסוגל לזהות?
- אילו תכונות יש לקבוצת השפות שאותן יכול אס"ד לקבל?
- האם קיימים מודלים נוספים המקבלים את אותה קבוצת שפות?

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 $w = a_1 a_2 \dots a_3$, ותהי $L = \{w\}$. כיצד נבנה אס"ד לשפה?



רעיון הבנייה: נרצה לראות את רצף התווים $a_1 a_2 \dots a_3$ 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:

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

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

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

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

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	
q_1	$* \rightarrow q_0$
q_2	q_1
q_3	q_2
q_0	q_3

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:

$$Q=Q_1\times Q_2,\qquad F=F_1\times F_2$$
$$q_0=(q_{01},q_{02})$$
$$\forall q_1\in Q_1,q_2\in Q_2:\delta\big((q_1,q_2),a\big)=\big(\delta_1(q_1,a),\delta_2(q_2,a)\big)$$

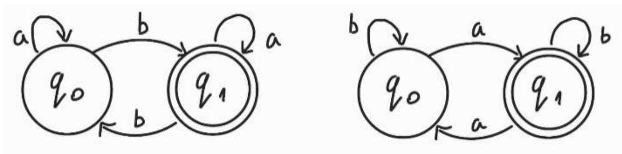
(נשים לב ש- δ מוגדר עבור תו בודד).

דוגמה: נבנה אס"ד לשפה הבאה:

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

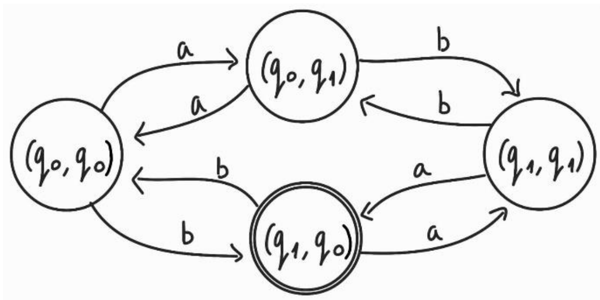
(שפת כל המילים מעל a,b שיש בהן מספר זוגי של a ומספר אי-זוגי של b).

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:

$$\begin{aligned}\hat{\delta}\big((q_1,q_2),w\big)&\overset{\text{א}}{=}\hat{\delta}\big((q_1,q_2),ua\big)\overset{\text{ב}}{=}\delta\big(\hat{\delta}\big((q_1,q_2),u\big),a\big)\overset{\text{ג}}{=}\delta\big(\hat{\delta}\big((q_1,q_2),u\big),a\big)\overset{\text{ד}}{=}\\&=\delta\left(\left(\hat{\delta}_1(q_1,u),\hat{\delta}_2(q_2,u)\right),a\right)\overset{\text{ה}}{=}\left(\delta_1\big(\hat{\delta}_1(q_1,u),a\big),\delta_2\big(\hat{\delta}_2(q_2,u),a\big)\right)\overset{\text{ו}}{=}\\&=\big(\delta_1(q_1,w),\delta_2(q_2,w)\big)\end{aligned}$$

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

כעת, נוכיח ש $L(A)=L_1\cap L_2$:

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

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

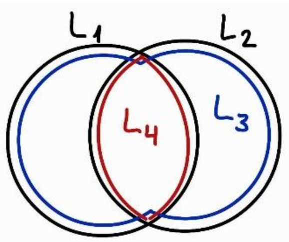
שפת החיסור $L = L_1 \setminus L_2$ מתקבלת את כל המילים שנמצאות גם ב-L_1 ולא ב-L_2. אם L_1, L_2 רגולריות, בהכרח גם L רגולרית.
הוכחה: מתקיים:

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

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: $L_1 \cap L_2 = L_4$, $L_1 \cup L_2 = L_3$. 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:

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