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

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: $L_1 \subseteq L_1 \cdot L_2 \Longleftrightarrow \epsilon \in L_2$. (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 $\epsilon \neq v$, 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.

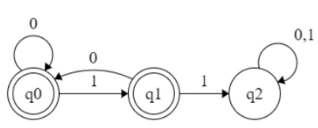
הצגה על ידי גרף:

- 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
$\to. * q_0$	q_0	q_1
$* q_1$	q_0	q_2
q_2	q_2	q_2



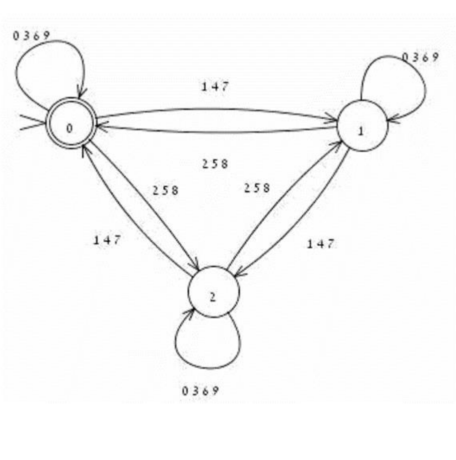
דוגמה 2: אוטומט המקבל את כל המחרוזות המסתיימות ב- "abc".

Q	a	b	c	$else$
$\to.\ q_0$	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 מחרוזת. אינטואיציה: δ מורחבת מחושבת ממצב q ומחרוזת $a_1 a_2 \ldots a_n$ ע"י כך שנעקוב אחרי מסלול המתחיל ב-q ועובר דרך קשתות עם תיוגים $a_1, a_2, \ldots, a_n$.

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.