מודלים חישוביים, חישוביות וסיבוכיות - ניצן ברזילי

מקרא: **הגדרה**, **משפט או טענה**, **צורת חשיבה או שיטה**, **ריכוז דוגמאות**

חלק I
הקדמה ורקע מתמטי

קבוצות, יחסים ועוצמות

- **קבוצה:** קבוצה A היא אוסף של איברים.
  - הקבוצה הריקה מסומנת $\emptyset$.
  - לכל איבר x נסמן $x \in A$ אם"ם x חבר בקבוצה A.
  - לכל איבר x בהכרח מתקיים אחד בדיוק משני המצבים הבאים: $x \in A$ או $x \notin A$.
- **איחוד של קבוצות:** איחוד של שתי קבוצות A, B הוא קבוצה המכילה את כל האיברים מ־A ואת כל האיברים מ־B, כלומר $A \cup B = \{x \mid x \in A \ or \ x \in B\}$.
- **חיתוך של קבוצות:** חיתוך של שתי קבוצות A, B הוא קבוצה המכילה את כל האיברים שנמצאים ב־A וגם ב־B, כלומר $A \cap B = \{x \mid x \in A \ and \ x \in B\}$.
- **משלים של קבוצה:** המשלים של קבוצה A הוא קבוצת כל האיברים שלא נמצאים ב־A.
- **הפרש בין קבוצות:** ההפרש בין קבוצות A, B הוא קבוצה המכילה את כל האיברים ב־A שלא נמצאים ב־B, כלומר $A \backslash B = \{x \mid x \in A \ and \ x \notin B\}$.
- **הכלה של קבוצות:** נאמר שקבוצה A מוכלת בקבוצה B ונסמן $A \subseteq B$ אם כל איבר ב־A קיים ב־B, כלומר $\forall x \in A, x \in B$.
- **שוויון בין קבוצות:** נאמר שקבוצות הן זהות ונסמן $A = B$ אם ורק אם $A \subseteq B$ וגם $B \subseteq A$.
- **קבוצת החזקה:** תהי A קבוצה. קבוצת החזקה של A היא קבוצת כל תתי הקבוצות של A, כלומר $\{C : C \subseteq A\}$ ומסומנת $P(A)$ או 2^A.
- **מכפלה קרטזית בין קבוצות:** יהיו A, B קבוצות. המכפלה הקרטזית של A, B היא קבוצת הזוגות הסדורים מ־A ומ־B, כלומר $A \times B = \{(a, b) \mid a \in A, b \in B\}$.
- **יחס:** יהיו שתי קבוצות S, T. יחס בין S ל־T הוא תת קבוצה $R \subseteq S \times T$.
  - **יחס רפלקסיבי:** יחס בין קבוצה לעצמה $R \subset A \times A$ מכונה רפלקסיבי אם לכל $a \in A$ מתקיים $(a, a) \in R$.
  - **יחס סימטרי:** יחס בין קבוצה לעצמה $R \subset A \times A$ מכונה סימטרי אם לכל $a, b \in A$ המקיימים $(a, b) \in R$, מתקיים $(b, a) \in R$.
  - **יחס טרנזיטיבי:** יחס בין קבוצה לעצמה $R \subset A \times A$ מכונה טרנזיטיבי אם לכל $a, b, c \in A$ מתקיים שאם $(a, b) \in R$ וגם $(b, c) \in R$, אז $(a, c) \in R$.
- **יחס שקילות:** יחס בין קבוצה לעצמה $R \subset A \times A$ מכונה יחס שקילות אם הוא רפלקסיבי, סימטרי וטרנזיטיבי.
- **מחלקת שקילות:** לכל יחס $R \subset A \times A$ ולכל איבר $a \in A$, מחלקת השקילות של a מעל R היא $\{b \in A \mid (a, b) \in R\}$, ומסומנת $[a]_R$.
  - אם R הוא יחס שקילות, כל מחלקות השקילות יוצרות חלוקה זרה של A, כלומר לכל $a, b \in A$ מתקיים $[a]_R = [b]_R$ או $[a]_R \cap [b]_R = \emptyset$, וכן מתקיים $\bigcup_{a \in A} [a] = A$.
- **עוצמה של קבוצה סופית:** תהי קבוצה סופית A, העוצמה של A היא כמות האיברים ב־A, ומסומנת $|A|$.
- **עוצמה של קבוצה שאינה בהכרח סופית:** יהיו שתי קבוצות A, B:

- נאמר שהעוצמה שלהן שווה ונסמן $|A|=|B|$ אם"ם קיימת פונקציה חח"ע ועל ביניהן.
- נאמר שמתקיים $|A|\leq|B|$ אם קיימת העתקה חח"ע $f:A\to B$.
- נאמר שמתקיים $|A|<|B|$ אם קיימת העתקה חח"ע $f:A\to B$ וגם <u>לא קיימת</u> העתקה על $g:A\to B$.

- **קבוצה בת מניה:** נסמן ב־$\aleph_0$ את העוצמה של $|\mathbb{N}|$. נאמר שקבוצה בת מנייה אם העוצמה שלה היא $\aleph_0$.
  - **טענה:** מתקיים $|\mathbb{Z}|=|\mathbb{Q}|=\aleph_0$, כלומר השלמים והרציונליים הם קבוצות בנות מניה.
  - **טענה:** $2^{\aleph_0}\neq\aleph_0$, ובפרט $2^{\aleph_0}>\aleph_0$.
  - **טענה:** מתקיים $|\mathbb{R}|=|[0,1]|=2^{\aleph_0}$ (הערה - $2^{\aleph_0}$ מכונה גם א), כלומר - הממשיים <u>אינם</u> קבוצה בת מניה.

שפות

- **אלפבית:** קבוצה סופית של סימנים (המכונים אותיות), מסומן בד"כ ע"י Σ.
- **מילה:** סדרה סופית של אותיות מ־Σ, מסומנת לעתים ע"י w. קבוצת כל המילים שניתן ליצור מהאותיות של Σ מסומנת Σ^*.
  - <u>אורך של מילה:</u> מספר האותיות במילה, מסומן ע"י $|w|$.
  - <u>המילה הריקה:</u> מילה יכולה להיות סדרה באורך 0 (כלומר מחרוזת ריקה), המילה הריקה מסומנת ע"י ε.
  - **טענה:** מתקיים $|\Sigma^*|=\aleph_0$ (כלומר קבוצת כל המילים האפשריות היא בת מניה).
- **שרשור של מילים:** יהיו w_1,w_2 מילים, השרשור שלהן יהיה הצמדה שלהן w_1w_2. כיוון ש־Σ^* מכילה רק מילים באורך סופי, אזי היא סגורה תחת שרשור.
- **שפה פורמלית:** שפה מעל אלפבית Σ היא קבוצה (סופית או אינסופית) של מילים אפשריות שנוצרו מהאותיות ב־Σ, כלומר תת קבוצה של Σ^*.
  - **הערה:** השפה הריקה $L=\emptyset$ היא שפה.
  - **הערה:** קבוצה המכילה רק את ε, כלומר $L=\{\varepsilon\}$ היא שפה, אך היא <u>אינה</u> השפה הריקה.

קבוצת כל השפות: נסמן את קבוצת כל השפות ב־2^{Σ^*}.

- **טענה:** מתקיים $|\Sigma^*|=\aleph_0<2^{\aleph_0}=|2^{\Sigma^*}|$, כלומר יש יותר שפות מעל Σ מאשר שיש מילים מעל Σ.
- **פעולות על שפות:** ניתן לבצע על שפות כל פעולה שניתן לבצע על קבוצות. בנוסף, נגדיר את הפעולות הבאות:
  - **שרשור שפות:** יהיו שפות L_1,L_2, נגדיר את השרשור שלהן להיות קבוצת כל המילים שנוצרו באמצעות שרשור של מילה מ־L_1 עם מילה מ־L_2, כלומר $L_1\cdot L_2=\{w_1\cdot w_2|w_1\in L_1,w_2\in L_2\}$.

חלק II
מודלים חישוביים

אוטומט סופי דטרמיניסטי DFA

- **אוטומט סופי דטרמיניסטי (DFA - Deterministic Finite Automaton):** חמישיה $\mathcal{A}=\langle Q,\Sigma,q_0,\delta,F\rangle$ כאשר:
  - Q היא קבוצה סופית של מצבים
  - Σ הוא אלפבית
  - $q_0\subseteq Q$ הוא מצב התחלתי
  - $\delta:Q\times\Sigma\to Q$ היא פונקציית מעברים
  - $F\subseteq Q$ היא קבוצת מצבים מקבלים / סופיים
- **ריצה של מילה על אוטומט דטרמיניסטי:** תהי מילה $w=w_1w_2..w_n$ (עם $\forall i\ w_i\in\Sigma$). ריצה של w על אוטומט היא סדרה של מצבים $q_0,q_1,..,q_n$, כאשר q_0הוא המצב ההתחלתי, ולכל i מתקיים $q_i=\delta(q_{i-1},w_i)$ (כלומר המעבר בין מצב מסוים למצב הבא מתבצע באמצעות פונקציית המעבר δ).
  - נגיד שריצה היא **מקבלת** אם $q_n\in F$.

- **שפה על אוטומט דטרמיניסטי:** יהי $\mathcal{A}$ אוטומט סופי דטרמיניסטי. השפה על $\mathcal{A}$ מכילה את כל המילים $w \in \Sigma^*$ כך שהריצה של $\mathcal{A}$ על w היא מקבלת (כלומר כל המילים שהמצב האחרון בריצה שלהן על האוטומט הוא מצב מקבל). מסומנת ע"י $L(\mathcal{A})$.

- **שפה משלימה:** תהי L שפה מעל אלפבית Σ, אזי השפה המשלימה של L היא $\Sigma^*\backslash L$, כלומר כל המילים האפשריות שלא נמצאות ב־L. מסומנת $\bar{L}$.

- **פעולות על שפות:** יהיו L_1, L_2 שפות, נגדיר את הפעולות הבאות:
  - **איחוד של שפות:** $L_1 \cup L_2 = \{w | w \in L_1 or\ w \in L_2\}$
  - **חיתוך של שפות:** $L_1 \cap L_2 = \{w | w \in L_1\ and\ w \in L_2\}$.
  - **שרשור של שפות:** $L_1 \cdot L_2 = \{w_1 w_2 | w_1 \in L_1, w_2 \in L_2\}$.
  - **פעולת כוכב על שפה:** $L^* = \{w_1 w_2 ... w_k | k \geq 0,\ \forall i \in [k]\ w_i \in L\}$.
    - **הערה:** L^* היא כמעט תמיד אינסופית. היא תהיה סופית (ויתקיים $L^* = \{\varepsilon\}$) רק באחד משני המקרים הבאים: $L = \emptyset$ או $L = \{\varepsilon\}$.

- **שפה רגולרית:** שפה L כך שקיים אוטומט סופי דטרמיניסטי $\mathcal{A}$ כך ש־$L(\mathcal{A}) = L$.
  - מחלקת כל השפות המתקבלות ע"י אוטומטים סופיים דטרמיניסטיים, כלומר הקבוצה המכילה כל שפה רגולרית, מסומנת REG.
  - **טענה:** לא כל שפה היא רגולרית - ישנן $\aleph_0$ שפות רגולריות, אבל $2^{\aleph_0}$ שפות.
  - **טענה:** כל שפה סופית היא רגולרית.
  - **מסקנה:** תהא שפה L, נאמר כי $L \in REG$ (כלומר שהיא רגולרית) אם מתקיימים אחד התנאים הבאים:
    - ל־L יש DFA
    - ל־L יש NFA
    - ל־L יש $Regex$

- **תכונות סגור של שפות רגולריות:** יהיו L_1, L_2 שפות רגולריות. אזי הרגולריות נשמרת תחת הפעולות הבאות:
  - **סגור לאיחוד:** $L_1 \cup L_2$ רגולרית.
  - **סגור לחיתוך:** $L_1 \cap L_2$ רגולרית.
  - **סגור להשלמה:** $L_1 \backslash L_2$ רגולרית.
  - **סגור לשרשור:** $L_1 \cdot L_2$ רגולרית.
  - **סגור תחת כוכב:** L_1^* רגולרית.

- **הפונקציה δ^* באוטומט דטרמיניסטי:** יהי אוטומט סופי דטרמיניסטי $\mathcal{A} = \langle Q, \Sigma, q_0, \delta, F \rangle$. נזהה כי התכונה של פונקצית המעברים δ היא שהיא מאפשרת לנו רק לשאול מה יקרה אחרי שנקרא <u>אות אחת</u>. נגדר את הפונקציה $\delta^* : Q \times \Sigma^* \to Q$ שמאפשרת לקרוא <u>מילה (ולא רק אותיות)</u> באופן הבא: $\delta^*(q,w) = \begin{cases} q, & \text{if } w = \varepsilon \\ \delta\left(\delta^*\left(q, w'\right), \sigma\right), & \text{if } w = w' \cdot \sigma \text{ where } w' \in \Sigma^* \text{ and } \sigma \in \Sigma \end{cases}$.
  - **הערה:** בהנתן Q, Σ, לא כל פונקציה $Q \times \Sigma^* \to Q$ היא δ^*.

- **טענה:** עבור אוטומט A, יתקיים כי $L(A)$ אינסופית אם"ם בגרף של A יש מעגל שאפשר להגיע אליו מ־Q_0 ושאפשר להגיע ממנו למצב מקבל.

אוטומט סופי לא דטרמיניסטי NFA

- **אוטומט סופי לא דטרמיניסטי (NFA - Nondeterministic Finite Automaton):** חמישיה $\mathcal{A} = \langle Q, \Sigma, Q_0, \delta, F \rangle$ כאשר:
  - Q היא קבוצה סופית של מצבים
  - Σ הוא אלפבית
  - $Q_0 \subseteq Q$ הוא <u>קבוצה</u> של מצבים התחלתיים (בשונה מ־DFA שבו יש רק מצב התחלתי q_0 יחיד, ב־NFA יתכנו מספר מצבים התחלתיים, ובפרט יתכן מצב שבו לאותה מילה יש מספר ריצות אפשריות, שיתכן שחלקן יסתיימו במצב מקבל אבל חלקן לא)

- $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \to 2^Q$ היא פונקציית מעברים (ב־NFA יתכנו "מעברי ε", כלומר מעברים שלא מתרחשת בהם קריאה של אות. כמו כן, יתכנו מצבים בהם למצב מסוים יש יותר ממעבר אחד המתייחס לאותה אות), שמקבלת מצב ואות ומחזירה קבוצת מצבים.
- $F \subseteq Q$ היא קבוצת מצבים מקבלים

- **ריצה של מילה על אוטומט לא דטרמיניסטי:** תהי מילה $w = w_1 w_2 .. w_n$ (עם $\forall i\ w_i \in \Sigma$). ריצה של w על אוטומט היא סדרה של מצבים $r_0, r_1, .., r_m$ (עם $m \geq n$, שכן יתכנו מעברי ε) כך שמתקיימים התנאים הבאים:
  - ניתן לכתוב את w בתור שרשור $w = y_1 y_2 .. y_m$ כאשר לכל $i \in [m]$ מתקיים $y_i \varepsilon \in \Sigma^* \cup \{\varepsilon\}$ (האותיות המקוריות של w כך שיתכנו εים ביניהן)
  - $r_0 \in Q_0$ (כלומר המצב הראשון הוא מצב התחלתי)
  - לכל $i \in [0, m)$ מתקיים $r_{i+1} \in \delta(r_i, y_{i+1})$

- נגיד שריצה היא **מקבלת** אם $r_m \in F$. נאמר ש־$\mathcal{A}$ מקבלת את w אם קיימת ריצה מקבלת של $\mathcal{A}$ על w.

- **שפה על אוטומט לא דטרמיניסטי:** יהי $\mathcal{A}$ אוטומט סופי לא דטרמיניסטי. השפה על $\mathcal{A}$ מכילה את כל המילים $w \in \Sigma^*$ כך ש־$\mathcal{A}$ מקבלת את w (כלומר שקיימת ריצה מקבלת של $\mathcal{A}$ על w). מסומנת ע"י $L(\mathcal{A})$.

- **משפט:** לכל NFA (בין אם הוא מכיל או לא מכיל מצבי ε) קיים DFA שקול (כלומר הם מתארים את אותה השפה).

- **הפונקציה ρ באוטומט לא דטרמיניסטי:** יהי אוטומט לא דטרמיניסטי $\mathcal{A} = \langle Q, \Sigma, q_0, \delta, F \rangle$, נסמן את $\mathcal{A}' = \langle Q', \Sigma, q_0, \delta, F' \rangle$ את ה־DFA השקול לו. נסמן את הפונקציה $\rho : Q' \times \Sigma \to Q'$ (כיוון ש־Q' היא קבוצה של קבוצות, זה שקול ל־ $\rho : 2^Q \times \Sigma \to 2^Q$) שהיא פונקצית המעברים הדטרמיניסטיים של $\mathcal{A}'$.

- **הפונקציה ρ^* באוטומט לא דטרמיניסטי:** יהי אוטומט לא דטרמיניסטי $\mathcal{A} = \langle Q, \Sigma, q_0, \delta, F \rangle$, נסמן את $\mathcal{A}' = \langle Q', \Sigma, q_0, \delta, F' \rangle$ את ה־DFA השקול לו. נסמן את הפונקציה $\rho^* : Q' \times \Sigma^* \to Q'$ שהיא הרחבה של הפונקציה ρ שמקבלת קבוצת מצבים ומילה (במקום קבוצת מצבים ואות בודדת ב־ρ המקורית). הפונקציה ρ^* מקיימת $\rho^*(S, w) = S'$ (כשנמצאים בקבוצת מצבים S וקוראים w מגיעים לקבוצת מצבים S').

- **הפונקציה δ^* באוטומט לא דטרמיניסטי:** יהי אוטומט לא דטרמיניסטי $\mathcal{A} = \langle Q, \Sigma, Q_0, \delta, F \rangle$. נזהה כי התכונה של פונקצית המעברים δ היא שהיא מאפשרת לנו רק לשאול מה יקרה אחרי שנקרא אות אחת. נגדר את הפונקציה $\delta^* : 2^Q \times (\Sigma^* \cup \{\varepsilon\}) \to 2^Q$ שהיא הרחבה של δ, שמאפשרת גם לקרוא מילה (ולא רק אותיות) וגם לטפל בקבוצות של מצבים (ולא רק במצב אחד כל פעם) באופן אינדוקטיבי על האורך של w: לכל קבוצה $S \subseteq 2^Q$, מתקיים:
  - עבור ε מתקיים $\delta^*(S, \varepsilon) \in S$
  - עבור מילים שמורכבות מאות אחת σ מתקיים $\delta^*(S, \sigma) = \bigcup_{t \in S} \delta(t, \sigma)$.
  - עבור מילה $w\sigma$ מתקיים $\delta^*(S, w\sigma) = \bigcup_{t \in \delta^*(S,w)} \delta(t, \sigma)$ (כלומר קודם כל מפעילים את δ^* על הרישא של המילה w, ואחר כך על האות האחרונה σ).

- **הקבוצה $E(q)$:** יהיו אוטומט לא דטרמיניסטי $\mathcal{A} = \langle Q, \Sigma, Q_0, \delta, F \rangle$ ומצב $q \in Q$, נגדיר את הקבוצה הבאה:

$$E(q) = \{q' \in Q \mid q' \text{ is reachable from q using only } \varepsilon \text{ transitions}\}$$

- **טענה:** לכל מילה $w \in \Sigma^*$ מתקיים $\delta^*(Q_0, w) = \rho^*(q_0, w)$ (האגף השמאלי הוא מצבים ש־$\mathcal{A}$ עשויה לבקר בהם לאחר קריאת w, האגף הימני הוא קבוצת מצבים ש־$\mathcal{A}'$ עשויה לבקר בהם לאחר קריאת w).

- **טענה - subset construction:** התנאים הבאים שקולים:
  - $w \in L(\mathcal{A})$
  - $\delta^*(Q_0, w) \cap F \neq \emptyset$ (קיימת קבוצה מקבלת, כלומר יש מצב מקבל שהגענו אליו עם δ^* אחרי שהתחלנו מ־Q_0 וקראנו את w).
  - $\rho^*(q_0, w) \in F'$
  - $w \in L(\mathcal{A}')$

ביטויים רגולריים

- **ביטוי רגולרי מעל אלפבית Σ:** הגדרה אינדוקטיבית:
  - בסיס: מתקיים כי $a \in \Sigma$ (אות בודדת), ε ו־$\emptyset$ הם ביטויים רגולריים
  - מקרה כללי: אם r_1, r_2 הם ביטויים רגולריים אז גם הביטויים הבאים הם רגולריים:
    - $r_1 + r_2$ (לעתים מסומן $r_1 \cup r_2$)
    - $r_1 \cdot r_2$
    - r_1^*
  - נסמן ב־r_1^+ את השרשור $r_1 \cdot r_1^*$ (שהוא בהכרח שרשור לא ריק)

- **שפה של ביטוי רגולרי:** יהי ביטוי רגולרי r, השפה של r המסומנת $L(r)$ מוגדרת ע"י:

$$\begin{array}{cc} L(\emptyset) = \emptyset & L(r_1 + r_2) = L(r_1) \cup L(r_2) \\ L(\varepsilon) = \{\varepsilon\} & L(r_1 \cdot r_2) = L(r_1) \cdot L(r_2) \\ L(a) = \{a\} & L(r_1^*) = L(r_1)^* \end{array}$$

- **טענה:** לא קיים פולינום $p : \mathbb{N} \to \mathbb{N}$ כך שבהנתן NFA עם n מצבים קיים DFA שקול עם $p(n)$ מצבים.

- **משפט:** לכל שפה $L \subseteq \Sigma^*$ מתקיים ש־L רגולרית אם"ם קיים ביטוי רגולרי r כך ש־$L(r) = L$.

- **למת הניפוח:** אם L רגולרית אז קיים **קבוע ניפוח** $p \geq 1$ כך שלכל מילה $w \in \Sigma^*$, אם $|w| \geq p$ (כלומר w יותר ארוכה מקבוע הניפוח), אז יש חלוקה של w ל־xyz כך שמתקיימים שלושת התנאים הבאים:
  1. $|y| > 0$ (כלומר $y \neq \varepsilon$)
  2. $|xy| \leq p$ (כלומר יתכן כי $x = \varepsilon$ או $z = \varepsilon$)
  3. לכל $i \geq 0$ מתקיים $xy^iz \in L$ (בפרט אם $i = 0$ זה אומר $xz \in L$, אם $i = 1$ זה אומר $w \in L$)

- **תכונה של למת הניפוח:** אפשר להשתמש בלמת הניפוח כדי להראות ששפה אינה רגולרית ־ מניחים בשלילה שהיא כן רגולרית ולכן קיים קבוע ניפוח, ומראים שלא ניתן לנפח עבור לפחות i כלשהו ועבור כל חלוקה רלוונטית. ההפך אינו נכון ־ אם שפה כן מקיימת את למת הניפוח, זה לא אומר שהיא רגולרית.

- **$GNFA$ (NFA מוכלל):** כמו NFA רגיל, רק שפונקצית המעברים לא עובדת על אותיות אלא על ביטויים רגולריים. נניח בה"כ שתמיד אם יש לשפה $GNFA$ אז קיים לה $GNFA$ שיש לו מצב התחלתי יחיד, מצב מקבל יחיד, והמצב ההתחלתי והמקבל שונים זה מזה.

מחלקות שקילות ומציאת DFA מינימלי

- **יחס מייהיל־נרוד / היחס $\sim_L$:** עבור שפה $L \subseteq \Sigma^*$, נגדיר יחס שקילות $\sim_L \subseteq \Sigma^* \times \Sigma^*$ המוגדר כך שעבור כל שתי מילים $x, y \in \Sigma^*$ נאמר ש־$x \sim_L y$ אם אין ל־x, y זנב מפריד (כלומר לא קיימת מילה שאם משרשרים אותה למילה אחת זה יהיה בשפה אבל אם משרשרים אותה למילה אחרת זה לא יהיה בשפה), כלומר מתקיים שלכל מילה $z \in \Sigma^*$, $xz \in L$ אם"ם $yz \in L$.
  - היחס $\sim_L$ הוא אכן יחס שקילות:
    - רפלקסיבי: לכל $x \in \Sigma^*$ מתקיים $x \sim_L x$.
    - סימטרי: לכל x, y המקיימים $x \sim_L y$ מתקיים $y \sim_L x$.
    - טרנזיטיבי: אם $x \sim_L y$ ו־$y \sim_L w$ אז $x \sim_L w$.
  - היחס הזה מחלק את L למחלקות שקילות שמפורדות באמצעות זנבות מפרידים שמקשרים ביניהן.

- **למה:** תהא $f : \mathbb{N} \to \mathbb{N}$ פונקציה מונוטונית עולה כך ש־$f(n) \in \omega(n)$ (עם $\omega(n) = \left\{g : \mathbb{N} \to \mathbb{N} \mid \lim_{n \to \infty} \frac{g(n)}{n} = \infty\right\}$). אזי לכל N, k קיים $n > N$ כך ש־$f(n+1) - f(n) > k$ (כלומר סדרת ההפרשים לא חסומה החל מ־k מסוים).

- **טענה:** תהא f כמו בלמה למעלה, אזי השפה $L_F = \left\{a^{f(n)} | n \in \mathbb{N}\right\}$ לא רגולרית.

- **משפט מייהיל נרוד MN:** המשפט אומר שלכל שפה $L \subseteq \Sigma^*$, L רגולרית שקול לזה שיש ליחס $\sim_L$ מספר סופי של מחלקות שקילות.
  - אם יש ל־$\sim_L$ n מחלקות שקילות אז יש DFA ל־L עם n מצבים.

- **טענה:** לכל $0 \leq i$, מתקיים $q =_i q'$ אם"ם לכל מילה w, אם $|w| \leq i$ אז $\delta^*(q', w) \in F \iff \delta^*(q, w) \in F$.

- $q =_0 q'$ אם"ם $\delta^*(q) \in F \iff \delta^*(q') \in F$ כלומר שני מצבים הם שקולים־0 אם שניהם מקבלים או ששניהם לא מקבלים
  - $q =_{i+1} q'$ אם"ם $q =_i q'$ וגם לכל $\sigma \in \Sigma$ מתקיים $\delta(q,\sigma) =_i \delta(q',\sigma)$.
- **טענה:** לסדרה $=_i$ יש נקודת שבת, כלומר נקודה $i \le |Q|$ שבה $=_i$ זהה ל־$=_{i+1}$ (בפועל זה אומר שקיימת נקודה שעבורה לא נפצל יותר, כי בכל איטרציה שאיננה נקודת שבת מתפצלת לפחות מחלקת שקילות אחת).
- **שקול A:** מתקיים $q =_A q'$ אם"ם $q =_i q'$ באיטרציה ה־i בה הגענו לנקודת שבת.

שפות חסרות הקשר

- **דקדוק חסר הקשר:**

$G =<$	$V,$	$\Sigma,$	$R,$	S	$>$
, מסומן CFG.	משתנים	א"ב / טרמינלים כך ש־$\Sigma \cap V \neq \emptyset$	חוקי גזירה מהצורה $V \to (V \cup \Sigma)^*$	$\in V$ משתנה התחלתי	

  - אם $w,u,v \in (V \cup \Sigma)^*$ מילים ו־$A \to w$ חוק בדקדוק, אז נאמר ש־$uAv \Rightarrow uwv$ (כאשר הסימון $\Rightarrow$ נקרא "מייצר את").
  - אם $u,v \in (V \cup \Sigma)^*$ מילים, אז נאמר ש־$u \overset{*}{\Rightarrow} v$ (כאשר הסימון $\overset{*}{\Rightarrow}$ נקרא "מייצר סטאר") אם יש סדרה $u = u_1 \Rightarrow u_2 \Rightarrow .. \Rightarrow u_k = v$ עם $k \ge 1$.
- **שפה חסרת הקשר:** שפה $L(G)$ היא כל המילים $w \in \Sigma^*$ כך ש־$S \overset{*}{\Rightarrow} w$, מסומנת CFL.
- **תכונות סגור של שפות חסרות הקשר:** יהיו $L(G_1), L(G_2)$שפות חסרות הקשר, אזי מתקיים:
  - סגור לאיחוד - $L(G_1) \cup L(G_2)$ היא שפה חסרת הקשר
  - סגור לשרשור - $L(G_1) \cdot L(G_2)$ היא שפה חסרת הקשר
  - <u>לא מתקיים</u> סגור לחיתוך - $L(G_1) \cap L(G_2)$ היא <u>לא בהכרח</u> שפה חסרת הקשר
- **צורה נורמלית של חומסקי:** נאמר ש־G CFG נמצא בצורה נורמלית של חומסקי אם כל אחד מהכללים שלו מהצורה:
  1. $S \to \varepsilon$
  2. $A \to BC$ עם $B,C \in V \backslash S$
  3. $A \to a$ כאשר a הוא טרמינל (אות בא"ב)
- **טענה:** לכל דקדוק חסר הקשר G קיים דקדוק חסק הקשר בצורה נורמלית של חומסקי G'.

חלק III
חישוביות

מכונות טיורינג

- **מכונת טיורינג (דטרמיניסטית):** מוגדרת באמצעות:

$M =<$	$Q,$	$\Sigma,$	$\Gamma,$	$\delta,$	$q_0,$	$q_{acc},$	q_{rej}	$>$
	קבוצת מצבים סופית	א"ב הקלט	א"ב העבודה	$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L,R\}$ $\delta(q,a) = \langle a,b,R \rangle$	$\in Q$ מצב התחלתי	$\in Q$ מצב מקבל	$\in Q$ מצב דוחה	

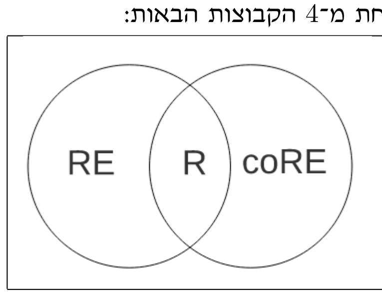
  - הביטוי $\delta(q,a) = \langle q',b,R \rangle$ אומר: כאשר M במצב q , הראש הקורא מצביע על תא שכתוב בו a, אז M עוברת למצב q', כותבת b במקום a, וזזה עם הראש הקורא תא אחד ימינה.
- **קונפיגורציה של מכונת טיורינג:** מוגדרת ע"י המצב הנוכחי, תוכן הסרט ומיקום הראש הקורא. עבור שתי מילים המופיעות ברציפות בסרט $v,u \in \Gamma^*$ ומצב $q \in Q$, נסמן ב־vqu את הקונפיגורציה שבה המצב הוא q, תוכן הסרט הוא $v \cdot u$ והראש הקורא מצביע על האות הראשונה ב־u.
  - **הקונפיגורציה ההתחלתית** של מכונת טיורינג על מילה $w \in \Sigma^*$ היא $q_0 w$ - הראש הקורא מצביע האות הראשונה ב־w והמצב הוא q_0.
  - **קונפיגורציות עוקבות** של מכונת טיורינג: יהיו $a,b,c \in \Gamma$, $u,v \in \Gamma^*$, $q,q' \in Q$.
    - אם q הוא מצב עצירה (מצב מקבל או דוחה), אז אין קונפיגורציה עוקבת והריצה מסתיימת.

* אם $\delta(q,b)=(q',c,L)$ אז הקונפיגורציה העוקבת של $uaqbv$ היא $uq'acv$.
* אם $\delta(q,b)=(q',c,R)$ אז הקונפיגורציה העוקבת של $uaqbv$ היא $uacq'v$.
* אם $\delta(q,a)=(q',b,L)$ אז הקונפיגורציה העוקבת של qav היא $q'bv$ (כלומר - לא "נופלים מהסרט מצד שמאל" אלא דורכים במקום).

- **ריצה סופית של מכונת טיורינג:** ריצה סופית של M על $w=w_1...w_n\in\Sigma^*$ מוגדרת ע"י סדרה של קונפיגורציות $R=C_0C_1..C_m$ כך שמתקיימים התנאים הבאים:
  - $C_0=q_0w$ (הקונפיגורציה ההתחלתית של M על w).
  - לכל $i\in[0,m]$ מתקיים C_{i+1} עוקבת ל-C_i .
  - C_m היא קונפיגורציה עוצרת (המצב שלה הוא q_{acc} או q_{rej}).
    * כל ריצה היא בהכרח עוצרת ומקבלת / עוצרת ודוחה / לא עוצרת.
    * יש קונפיגורציה שחוזרת על עצמה $\Leftarrow$ הריצה אינה עוצרת (הגרירה בכיוון ההפוך לא בהכרח נכונה).

שפות של מכונות טיורינג

- **שפה של מכונת טיורינג:** $L(M)$ היא קבוצת כל המילים w כך שקיימת ריצה סופית ומקבלת של M על w.
  - נאמר שמכונת טיורינג **מזהה** את השפה $L\subseteq\Sigma^*$ אם $L(M)=L$.
  - נאמר ששפה L היא **ניתנת למניה רקורסיבית** recursively enumerable ונסמן $L\in RE$ אם קיימת מכונת טיורינג **שמזהה** אותה.
    * המחלקה המשלימה של RE, המסומנת $coRE$, היא המחלקה $\Sigma^*\backslash RE$.
  - **טענה:** המחלקה המשלימה של RE, המסומנת $coRE$, מקיימת $L\in coRE$ אם"ם $\overline{L}\in RE$, כלומר שקיימת מכונת טיורינג M כך שלכל $w\in\Sigma^*$, אם $w\notin L$ אז M עוצרת ומקבלת (/דוחה), ואם $w\in L$ אז M דוחה (/מקבלת) או לא עוצרת.
  - נאמר שמכונת טיורינג **מכריעה** את L decides אם M מזהה את L ובנוסף עוצרת על כל קלט.
    * אם מכונה מכריעה שפה, היא בפרט גם מזהה אותה.
  - נאמר ששפה L היא **רקורסיבית** ונסמן $L\in R$ אם קיימת מכונת טיורינג **שמכריעה** אותה.
    * **טענה:** מתקיים $\overline{L}\in R$ (כי בהנתן מכונה M שמכריעה את L, המכונה $\overline{M}$ שמתקבלת מ-M ע"י החלפה בין q_{acc},q_{rej} מכריעה את $\overline{L}$)
    * **טענה:** מתקיים $L\in R \Leftarrow L\in RE$, כלומר $R\subset RE$ (הגרירה בכיוון ההפוך לא בהכרח נכונה).
    * **משפט:** מתקיים $R=RE\cap coRE$
  - כל שפה נמצאת באחת מ-4 הקבוצות הבאות:



$$\begin{array}{c} R \\ \hline RE\backslash R \\ \hline coRE\backslash R \\ \hline RE\cup coRE \end{array}$$

- **תכונות סגור של RE:** מתקיימות התכונות הבאות:
  - RE סגורה לאיחוד, כלומר לכל $L_1,L_2\in RE$ מתקיים $L_1\cup L_2\in RE$.
  - RE סגורה לשרשור, כלומר לכל $L_1,L_2\in RE$ מתקיים $L_1\cdot L_2\in RE$.
- **מכונות חישוב שקולות:** נאמר ששתי מכונות חישוב N,M הן שקולות אם לכל $w\in\Sigma^*$:
  - N מקבלת את w $\iff$ M מקבלת את w
  - N דוחה את w $\iff$ M דוחה את w
  - Nלא עוצרת על w $\iff$ M לא עוצרת על w
- **מודלים חישוביים שקולים:** נאמר ששני מודלים חישוביים x,y הם שקולים אם:

- לכל מכונה מסוג x יש מכונה מסוג y ששקולה לה
- לכל מכונה מסוג y יש מכונה מסוג x ששקולה לה

- **מודלים ששקולים למכונת טיורינג:**
  - מכונת טיורינג עם מצב סופי k של סרטים
    * בפרט, מכונת טיורינג עם שני סרטים - כמו מכונת טיורינג רגילה רק עם $\delta : Q \times \Gamma \times \Gamma \to Q \times \Gamma \times \Gamma \times \{L, R\} \times \{L, R\}$ ו-$\delta(q, \gamma_1, \gamma_2) = (q', \gamma_1', \gamma_2', L, R)$.
  - מכונת טיורינג עם סרט אינסופי בשני הכיוונים
  - מכונת טיורינג לא דטרמיניסטית
  - מכונת טיורינג שבמקום סרט יש לה מטריצה דו מימדית אינסופית
  - מכונת טיורינג שיכולה גם להשאר במקום (ולא רק להתקדם ימינה או שמאלה)
  - מכונת RAM

- **טענה:** לכל מכונת טיורינג עם שני סרטים יש מכונת טיורינג ששקולה לה (זה נכון גם בכיוון ההפוך - פשוט לא נשתמש בסרט אחד).

- **אנומרטור / ספרן:** מכונת טיורינג ללא קלט, עם מדפסת המדפיסה מילים מתוך Σ^*, מסומן E.

- **שפה של ספרן:** השפה של אנומרטור E היא המילים שהאנומרטור מדפיס, כלומר $L(E) = \{w \mid E\ prints\ w\}$.

- **משפט:** מתקיים $L \in RE \iff$ קיים ספרן E כך ש-$L(E) = L$.

כריעות ואי כריעות

- **התזה של צ'רץ' וטיורינג:** אלגוריתם שקול להכרעה ע"י מכונת טיורינג. יש שלוש רמות לתיאור אלגוריתם:
  1. פסודו קוד (שפה עילית)
     - איך עוברים מפסודו קוד לתיאור פעולה של מכונת טיורינג - עבור איבר A שהקוד רץ עליו (כאשר A יכול להיות מכל מיני סוגים - פולינום, גרף, מטריצה וכו'), נסמן ב-$< A > \in \Sigma^*$ את הקידוד (המרה של האיבר A למילה ב-Σ^* שמכונת טיורינג יודעת לרוץ עליה). נתאר את אופן הפעולה של מ"ט שבודקת שהקידוד נכון, כלומר עוצרת ומקבלת עבור כל הקידודים של איברים ששייכים לשפה שלה. אפשר לעשות את זה ע"י להגדיר Γ שמכילה סימנים מיוחדים שנחליף אותיות במילה בהם כדי לציין פעולה כלשהי בקוד.
  2. תיאור הפעולה של מכונת טיורינג
  3. הגדרה מפורשת של מכונת טיורינג $M =< Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{rej} >$

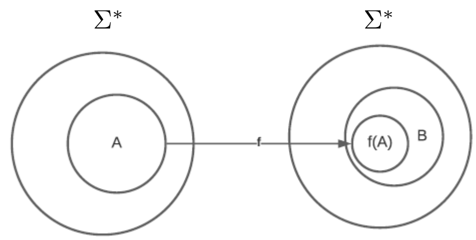
- **טענה:** השפה $A_{DFA} = \{\langle A, w\rangle | \quad w \in L(A)\ with\ DFA\ A\}$ מקיימת $A_{DFA} \in R$.

- **טענה:** לכל א"ב סופי Σ בגודל 2 או יותר קיימת שפה L כך שמתקיים $L \notin R$ (כלומר שפה שאינה כריעה - שלא קיימת מכונת טיורינג שמכריעה אותה)
  - **טענה:** עבור השפה $A_{TM} = \{\langle M, w\rangle |\ \text{w}\in\text{L}\ and, M\ is\ a\ Turing\ machine\}$ מתקיים $A_{TM} \notin R$, כלומר A_{TM} אינה כריעה
  - **טענה:** מתקיים $A_{TM} \in RE$ וכן $A_{TM} \notin coRE$
  - **טענה:** עבור השפה $HALT_{TM}$ המכילה כל קידוד $< M, w >$ כך שמ"ט M עוצרת על w, מקיימת $HALT_{TM} \in RE$ אך $HALT_{TM} \notin R$.

- **פונקציה ניתנת לחישוב:** עבור א"ב Σ, נאמר שהפונקציה $f : \Sigma^* \to \Sigma^*$ ניתנת לחישוב $computable$ אם קיימת מכונת טיורינג M_f שבהנתן קלט $x \in \Sigma^*$ תמיד עוצרת עם $f(x)$ על הסרט.

רדוקציה

- **רדוקצית מיפוי של שפה לשפה אחרת:** עבור שתי שפות $A,B\subseteq\Sigma^*$, נאמר ש־A ניתנת לרדוקצית מיפוי ל־B ונסמן $A\leq_m B$ ("A יותר קלה מ־B") אם קיימת פונקציה ניתנת לחישוב $f:\Sigma^*\to\Sigma^*$ (לא בהכרח חח"ע ועל) כך שלכל מילה $x\in\Sigma^*$ מתקיים $x\in A$ אם"ם $f(x)\in B$.
  - אינטואיציה - אם $A\leq_m B$ אז B "יותר קשה" מ־A



- **משפט הרדוקציה ב־R:** אם $A\leq_m B$ ו־$B\in R$ אז $A\in R$.
  - ניסוח אחר של משפט הרדוקציה - אם $A\leq_m B$ ו־$A\notin R$ אז $B\notin R$.
- **משפט הרדוקציה ב־RE:** אם $A\leq_m B$ ו־$B\in RE$ אז $A\in RE$.
- **משפט הרדוקציה ב־$coRE$:** אם $A\leq_m B$ ו־$B\in coRE$ אז $A\in coRE$.
- **משפט:** אם $A\leq_m B$ אם"ם $\overline{A}\leq_m\overline{B}$
- **משפט:** $\overline{A}\leq_m B$ אם"ם $A\leq_m \overline{B}$
- **משפט:** אם $A\leq_m B$ ו־$A\notin RE$ אז $B\notin RE$
- **משפט:** אם $A\leq_m B$ ו־$A\notin coRE$ אז $B\notin coRE$
- **תכונה סמנטית של מכונות טיורינג:** קבוצה p של מכונות טיורינג כך שלכל שתי מכונות טיורינג M_1,M_2שמקיימות $L(M_1)=L(M_2)$ מתקיים ש־$M_1\in p$ אם"ם $M_2\in p$ (כלומר או ששתיהן ב־p או ששתיהן לא ב־p).
  - **משפט רייס:** תהא p תכונה סמנטית לא טריויאלית של מכונות טיורינג, אזי השפה $L_p=\{<M>|M\in p\}$ לא כריעה.
  - **למה:** תהא p תכונה סמנטית לא טריויאלית של מכונות טיורינג, אזי אם $T\notin p$ אז $A_{TM}\leq_m L_p$.

חלק IV

סיבוכיות

מכונות טיורינג אי דטרמיניסטיות

- **מכונת טיורינג אי־דטרמיניסטית NTM:** זהה למכונת טיורינג רגילה (דטרמיניסטית), למעט פונקצית המעברים $\delta:Q\backslash\{q_{acc},q_{rej}\}\times\Gamma\to 2^{Q\times\Gamma\times\{L,R\}}\backslash\emptyset$ (כלומר δ לא דטרמיניסטית - המעבר בין אות ומצב לאות, מצב וכיוון הוא לא יחיד).
- **קבלת מילה ע"י NTM:** נאמר ש־NTM N מקבלת את $w\in\Sigma^*$ אם קיימת ריצה מקבלת של N על w.
- **קונפיגורציה עוקבת של NTM:** נאמר שקונפיגורציה d היא עוקבת לקונפיגורציה c אם קיים מעבר מ־c ל־d באמצעות δ.
- **מכונת NTM מכריעה:** נאמר ש־NTM מכריעה אם לכל $w\in\Sigma^*$, כל הריצות שלה על w עוצרות.
- **עץ ריצה של NTM:** יהיו N מכונת טיורינג לא דטרמיניסטית ו־$w\in\Sigma^*$. עץ הריצה של N ביחס ל־w המסומן $T_{N,w}=<V,E>$ מקודד את כל הריצות האפשריות של N מעל w. הוא אינו בהכרח מעומק סופי, והוא מוגדר ע"י V (קודקודים) ו־E (צלעות) באופן הבא:
  - V (קודקודים) הם זוגות $\langle c,i\rangle$ של c קונפיגורציות אפשריות בריצה של N על w ביחד עם אינדקס i (שמייצג את עומק הריצה בעץ).
    - בפרט השורש של העץ הוא $\langle q_0,0\rangle$.
  - E (צלעות) - קיימת צלע מ־$\langle c,i\rangle$ ל־$\langle d,i+1\rangle$ אם קיים מעבר לפי δ בין c ל־d בריצה על w.

- **הערות על עצי ריצה של NTM:**
  - עומק העץ לא בהכרח סופי.
  - מסלול בעץ $T_{N,w}$ מהשורש לקודקוד כלשהו מייצג ריצה חלקית של N על w
  - מסלול בעץ $T_{N,w}$ מהשורש לעלה מייצג ריצה עוצרת (מקבלת או דוחה) של N על w
  - הדרגה של $T_{N,w}$ תמיד חסומה ע"י $k \in \mathbb{N}$ כלשהו
- **למה:** מכונת טיורינג לא דטרמיניסטית N היא מכריעה אמ"מ כל עצי הריצה שלה סופיים.
- **הלמה של קניג:** בכל עץ אינסופי עם דרגת פיצול k סופית, קיים מסלול אינסופי.
- **משפט:** לכל NTM N מכריעה, קיימת מכונת טיורינג D שהיא מכריעה כך ש-$L(N) = L(D)$.
- **מכריע $Decider$:** מ"ט אי דטרמיניסטית שעוצרת בכל הריצות שלה.

בעיות לא כריעות

- **בעיית הריצוף $Tiling$:**
  - קלט: $<$ t_{init}, $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$, $\delta(q,a) = \langle a,b,R \rangle$; V, $\in T$ אריח התחלתי ; H, $H \subseteq T \times T$ תנאי שכנות במאוזן ; T, $\{t_0, t_1, .., t_n\}$ קבוצת אריחים סופית $>$
  - פלט: ריצוף חוקי בגודל $n \times n$.
    * נגדיר את הקבוצה $TILE = \{\langle T, V, H, t_{\text{int}} \rangle \ s.t. \ \forall \text{n} \geq 1 \text{ there is a tiling } n \times n \ \}$.
    * נגדיר ריצוף חוקי $n \times n$ ע"י $f : \{1, n\} \times \{1, n\} \to T$ המוגדרת באופן הבא:
      · $t(1,1) = t_{init}$
      · לכל $i \in [1,n]$ ולכל $j \in [1,n)$ מתקיים $H\left(f(i,j), f\left(i+1,j\right)\right)$,$V\left(f(i,j), f\left(i,j+1\right)\right)$
  - **טענה:** קיים ריצוף חוקי $n \times n$ לכל n אם"ם יש ריצוף חוקי (אינסופי) לרבע המישור החיובי.
- **בעיית אבני הדומינו (PCP) Post correspondence problem:**
  - קלט: אוסף סופי של אבני דומינו מהצורה $e_i = \boxed{\begin{array}{c} u_i \\ \hline d_i \end{array}}$ עם $u_i, d_i \in \Sigma^*$.
  - פלט: הכרעה האם קיימות אבני דומינו $e_{i_1}, e_{i_2}, e_{i_3}, .., e_{i_m}$ כך שהצמדה שלהן בזו אחר זו תיצור מצב בו שרשור כל המילים בחצאים העליונים של האבנים ושרשור כל המילים בחצאים התחתונים של האבנים יניבו את אותה המילה.
- **מציאת מסלול המילטון בגרף מכוון:**
  - קשה $(\in NP)$ לבדוק האם קיים מסלול המילטון (מסלול שעובר בכל הקודקודים בגרף, בכל קודקוד בדיוק פעם אחת) בגרף מכוון.
  - קל $(\in N)$ לבדוק האם מסלול נתון (סדרת קודקודים) בגרף מכוון הוא מסלול המילטון.

סיבוכיות זמן

- **משפט:** לכל מ"ט מרובת סרטים שעובדת בזמן $O(t(n))$ יש מ"ט שקולה בעלת סרט יחיד שעובדת בזמן $O\left(t^2(n)\right)$.
- **משפט:** אם שפה L ניתנת להכרעה בזמן $O(nlogn)$ אז L רגולרית.
- **המחלקה P:** נגדיר את המחלקה P (או $PTIME$) להיות $P = \bigcup\limits_k TIME(n^k)$, כלומר שפות שניתנות להכרעה בזמן פולינומיאלי באמצעות מ"ט <u>דטרמיניסטית</u>.
- **המחלקה NP:** נגדיר את המחלקה NP (או $NPTIME$) להיות $NP = \bigcup\limits_k NTIME(n^k)$, כלומר שפות שניתנות להכרעה בזמן פולינומיאלי באמצעות מ"ט <u>אי־דטרמיניסטית</u>.
- **המחלקה $EXPTIME$:** נגדיר את המחלקה $EXPTIME$ להיות $EXPTIME = \bigcup\limits_k TIME(2^{n^k})$, כלומר שפות שניתנות להכרעה בזמן אקספוננציאלי באמצעות מ"ט <u>דטרמיניסטית</u>.

- **הערה:** כאשר יש לנו אלגוריתם שרץ על מספר, אם האלגוריתם תלוי בערך של המספר ולא באורך שלו, הסיבוכיות של האלגוריתם תהיה תמיד אקספוננציאלית באורך הקלט.

- **המחלקה $co-NP$:** עבור שפה L, מתקיים $L\in co-NP$ אם"ם $\overline{L}\in NP$.

- **אבחנה:** מתקיים $P\subseteq NP\subseteq EXPTIME$.

- **משפט:** אם L ניתנת להכרעה ע"י מ"ט א"ד בזמן $t(n)$, אז L ניתנת להכרעה ע"י מ"ט דטרמיניסטית בזמן $2^{O(t(n))}$.

- **מאמת / מוודא / $Verifier$:** מוודא עבור שפה L הוא מ"ט דטרמיניסטית V כך ש-$L=\{w\mid \exists c\in\Sigma^{*}\ s.t.\ V\ accepts\ \langle w,c\rangle\}$ (כלומר מ"ט שהקלט שלה הוא זוגות $\langle w,c\rangle$ עם $w\in L$ ו**עֵד** $c\in\Sigma^{*}$ כלשהו, כך שהמכונה מקבלת זוג שכזה רק עבור עד ספציפי אחד לפחות).

- **סיבוכיות של מוודא** נקבעת ביחס למילה w:
  - **מוודא פולינומיאלי** הוא מוודא V שרץ על $\langle w,c\rangle$ בזמן פולינומיאלי ב-$|w|$.
    * הגדרה נוספת: נאמר ש-V מוודא פולינומיאלי עבור L אם L היא השפה שמכילה את כל המילים w כך שקיים c פולינומיאלי ב-w כך ש-V מקבלת בזמן פולינומיאלי את $<w,c>$.

- **משפט:** מתקיים $L\in NP$ אם"ם יש ל-L מוודא פולינומיאלי.

- **פונקציה ניתנת לחישוב בזמן פולינומיאלי:** פונקציה f כך שקיימת מכונת טיורינג דטרמיניסטית M_f שעבור קלט $w\in\Sigma^{*}$, עוצרת אחרי $t(w)$ צעדים (עם t פולינום כלשהו) עם $f(w)$ על הסרט.

- **רדוקציה פולינומיאלית:** נאמר ש-A ניתנת לרדוקציה פולינומיאלית ל-B ונסמן $A\leq_p B$ אם קיימת פונקציה f ניתנת לחישוב בזמן פולינומיאלי כך שלכל $w\in\Sigma^{*}$ מתקיים $w\in A$ אם"ם $f(w)\in B$.

- **משפט:** אם $A\leq_p B$ וגם $B\in P$ אז מתקיים כי $A\in P$.

- **שפה NP-קשה:** שפה L כך שלכל $L'\in NP$ מתקיים $L'\leq_p L$ (אינטואיציה - "אם $L\in P$, אז $P=NP$").
  - **טענה:** שפה L היא NP-קשה אם קיימת שפה NP-קשה L'' ומתקיים $L''\leq_p L$.

- **שפה NP-שלמה:** שפה L כך שמתקיימים שני התנאים הבאים:
  - $L\in NP$
  - L היא NP-קשה

- **משפט קוק-לוין:** מתקיים $SAT\in P$ אם"ם $P=NP$.

- **שפה $co-NP$-קשה:** שפה L כך שלכל $L'\in co-NP$ מתקיים $L'\leq_p L$ (אינטואיציה - "אם $L\in P$, אז $P=NP$").
  - **טענה:** שפה L היא NP-קשה אם קיימת שפה NP-קשה L'' ומתקיים $L''\leq_p L$.

- **שפה $c-NP$-שלמה:** שפה L כך שמתקיימים שני התנאים הבאים:
  - $L\in co-NP$
  - L היא $co-NP$-קשה

- **מסקנה:** שפה L היא NP-קשה אם"ם $\overline{L}$ היא $coNP$-קשה

- **מסקנה:** שפה L היא NP-שלמה אם"ם $\overline{L}$ היא $coNP$-שלמה

סיבוכיות זכרון (שטח) לינארית

- **סיבוכיות זכרון של מכונת טיורינג:** בהנתן מכונת טיורינג חד סרטית M שעוצרת על כל קלט, סיבוכיות הזכרון של M היא פונקציה $s:\mathbb{N}\to\mathbb{N}$ כך ש-$s(n)$ הוא מספר התאים ש-M משתמשת בהם בריצתה על מילה באורך n (אם M לא דטרמיניסטית, התנאי הזה צריך להתקיים בכל ריצה).

- **המחלקה $SPACE(s(n))$:** עבור $n\leq s(n)$ נגדיר את $SPACE\left(s(n)\right)$ להיות כל השפות L כך שקיימת מ"ט דטרמיניסטית M שמכריעה את L בסיבוכיות זכרון $O\left(s(n)\right)$.

- **המחלקה $NSPACE(s(n))$:** עבור $n\leq s(n)$ נגדיר את $NSPACE\left(s(n)\right)$ להיות כל השפות L כך שקיימת מ"ט אי דטרמיניסטית M שמכריעה את L בסיבוכיות זכרון $O\left(s(n)\right)$.

- **המחלקה $PSPACE$:** נגדיר את $PSPACE$ להיות כל השפות L כל שקיימת מ"ט דטרמיניסטית M שמכריעה את L - $PSPACE=\cup_k SPACE\left(n^k\right)$.
  - **שפה $PSPACE$-קשה:** נאמר ששפה L היא $PSAPCE-HARD$ אם לכל $L'\in PSAPCE$ קיימת רדוקציה פולינומיאלית $L'\leq_p L$.
  - **שפה $PSPACE$-שלמה:** נאמר ששפה L היא $PSAPCE-COMPLETE$ אם מתקיים:
    * $L\in PSPACE$
    * L היא $PSPACE$-קשה
- **המחלקה $NPSPACE$:** נגדיר את $NPSPACE$ להיות כל השפות L כל שקיימת מ"ט אי דטרמיניסטית M שמכריעה את L - $NPSPACE=\cup_k NSPACE\left(n^k\right)$.
- **קידוד של ריצה של מכונת טיורינג:** תהא M מכונת טיורינג שפועלת במקום S. המשתנים של M יהיו:
  - לכל $i\in S$ ולכל $a\in\Gamma$ נגדיר משתנה $x_{i,a}$ שמקבל 1 אם"ם כתוב בתא ה-i את האות a באותו הרגע.
  - לכל $i\in S$ נגדיר את המשתנה y_i שמקבל 1 אם"ם הראש הקורא נמצא כעת ב-i.
  - לכל $q\in Q$ נגדיר משתנה z_q שמקבל 1 אם"ם המכונה נמצאת במצב q.
- **דברים שידועים על היחס בין מחלקות:**
  - מתקיים $PSPACE=NPSPACE$ (בשונה מסיבוכיות זמן, כאן המחלקות זהות - סמלוץ של מכונה לא דטרמיניסטית על מכונה דטרמיניסטית תדרוש אותו סדר גודל של זכרון).
  - **טענה:** לכל $f(n)$ מתקיים $TIME(f(n))\subseteq SPACE(f(n))$ (כי מכונה שעוצרת תוך $f(n)$ צעדים לא יכולה להשתמש ביותר מ-$f(n)$ תאים).
  - **טענה:** לכל $f(n)$ מתקיים $SPACE(f(n))\subseteq TIME\left(2^{O(f(n))}\right)$ (כי אם המכונה ב-$SPACE\left(f(n)\right)$ היא טבוח עוצרת, ולכן לא יכול להיות שנחזור על אותה קונפיגורציה פעמיים, ולכן נעצור לאחר מספר צעדים אקספוננציאלי ב-$f(n)$).
  - **חסם לכמות הקונפיגורציות של מ"ט:** למ"ט עם סיבוכיות זכרון $s(n)$ יש לכל היותר $|Q|\cdot|\Gamma|^{s(n)}\cdot s(n)$ קונפיגורציות. זה חסם על זמן הריצה של המכונה כי ידוע שהיא עוצרת ולכן לא חוזרת על אותה קונפיגורציה פעמיים.
  - **טענה:** מתקיים $L\subseteq NL\subseteq PTIME\subseteq NP/coNP\subseteq PSPACE=NPSPACE\subseteq EXPTIME$.
  - **טענה:** מתקיים $PTIME\neq EXPTIME$.
- **משפט $Savitch$:** לכל פונקציה $S(n)\geq n$ מתקיים $NPSPACE(S(n))\subseteq PSPACE\left(S^2(n)\right)$. כלומר, בהנתן מכונת טיורינג שעובדת בזכרון $S(n)$, קיימת מכונת טיורינג דטרמיניסטית שעובדת בזכרון $S^2(n)$.
  - **מסקנה:** זה גורר ש-$PSPACE=NPSPACE$.
  - **מסקנה:** זה גורר ש-$NPSPACE=\overline{NPSPACE}$.
- **פונקציה חשיבה בזמן:** נאמר כי t היא ניתנת לחישוב בזמן אם יש מ"ט שמקבלת מתור קלט את 1^n (הייצוג האונרי של n) ומחשבת את $t(n)$ בבינארי בזמן $O(t(n))$.
- **משפט ההירכיה בזמן:** תהי $t:\mathbb{N}\to\mathbb{N}$ חשיבה בזמן, אזי יש שפה L שניתנת להכרעה בזמן $O\left(t\left(n\right)\right)$ אבל לא ניתנת להכרעה בזמן $O\left(\frac{t(n)}{log(t(n))}\right)$.
- **מסקנה:** לכל $c_1>c_2\geq 2$, מתקיים $TIME(n^{c_2})\subsetneqq TIME(n^{c_1})$.
- **מסקנה:** לכל $c_1>c_2\geq 2$, מתקיים $P\subsetneqq EXPTIME$.
- **למה:** יש מ"ט S כך שבהנתן קלט $\langle M,w,t\rangle$, המכונה S יכולה לחשב את הקונפיגורציה ה-t-ית של ריצת M על w, בזמן $O\left(tlog(t)\cdot p(\langle M\rangle)\right)$ עבור p מסוים.
- **פונקציה חשיבה במקום:** נאמר כי t היא ניתנת לחישוב במקום אם יש מ"ט שמקבלת מתור קלט את 1^n (הייצוג האונרי של n) ומחשבת את $t(n)$ בבינארי במקום $O(t(n))$.
- **משפט ההירכיה במקום:** תהי $t:\mathbb{N}\to\mathbb{N}$ כך ש-$t=\Omega(log(n))$ וגם t חשיבה במקום, אזי יש שפה L שניתנת להכרעה במקום $O\left(t\left(n\right)\right)$ אבל לא ניתנת להכרעה בזמן $o(t(n))$.

סיבוכיות זכרון (שטח) תת־לינארית

- **מודל החישוב:** נגדיר מודל חישוב ע"י מכונה עם שני סרטים: סרט קלט שהוא $read\ only$ וסרט עבודה קטן שהוא $write\ only$, כאשר סיבוכיות הזכורן של המודל תוגדר לפי סרט העבודה.
- **המחלקה $LOGPSPACE$ (L):** נגדיר את $LOGSPACE$ להיות כל השפות L כל שקיימת מ"ט <u>דטרמיניסטית</u> M שמכריעה את L עם סרט עבודה שמשתמש ב־$logn$ תאים לכל מילה באורך n. המחלקה מסומנת גם L.
  - בפרט, ישנן מכונות טיורינג שעובדות בשטח קבוע.
- **המחלקה $NLOGPSPACE$ (NL):** נגדיר את $LOGSPACE$ להיות כל השפות L כל שקיימת מ"ט <u>אי־דטרמיניסטית</u> M שמכריעה את L עם סרט עבודה שמשתמש ב־$logn$ תאים לכל מילה באורך n. המחלקה מסומנת גם NL.
- **מסקנות ממשפט $Savitch$:** ממשפט זה ניתן להסיק שמתקיים $NLOGSPACE \subseteq SPAVE(log^2 n) \neq LOGSPACE$, כלומר לא ניתן להזיק ש־$NLOGSPACE \subseteq LOGSPACE$.
- **המחלקה $co-NL$:** נאמר ששפה $L \in co-NL$ אם מתקיים $\overline{L} \in NL$.
- **משרן בשטח לוגריתמי log-space transducer:** מכונת טיורינג דטרמיניסטית M עם שלושה סרטים: סרט קלט $Read\ only$, סרט עיבוד קטן $Read/write$, וסרט פלט $Write\ only$. עבור מילת קלט w, המכונה M משתמשת ב־$O\left(log(|w|)\right)$ תאים בסרט העיבוד כדי לחשב את הפלט (אותו היא כותבת על סרט הפלט).
- **פונקציה ניתנת לחישוב בשטח לוגריתמי:** נאמר שפונקציה $f:\Sigma^* \to \Sigma^*$ ניתנת לחישוב בשטח לוגריתמי אם קיים משרן בשטח לוגריתמי כך שלכל מילת קלט $w \in \Sigma^*$, f עוצרת עם $f(w)$ על סרט הפלט.
- **רדוקציה בשטח לוגריתמי:** עבור $A, B \subseteq \Sigma^*$ נאמר ש־B $A \leq_{logspace}$ אם קיימת פונקציה f ניתנת לחישוב בשטח לוגריתמי כך שמתקיים $w \in A$ אם"ם $f(w) \in B$.
- **משפט הרדוקציה בשטח לוגריתמי:** אם $A \leq_{logspace} B$ ו־$B \in LOGSPACE$ אז $A \in LOGSPACE$.
- **שפה NL־קשה:** שפה A תקרא NL־קשה אם לכל שפה $B \in NL$, קיימת רדוקציה בשטח לוגריתמי $B \leq_{logspace} A$.
- **שפה NL־שלמה:** שפה A תקרא NL־שלמה אם $A \in NL$ וגם A היא NL־קשה.
- **משפט אימרמן:** מתקיים $NL = coNL$.
  - **למה:** אם $\overline{PATH} \in NL$ ינבע מכך ש־$NL \subseteq coNL$.

הירכית מחלקות הסיבוכיות

- **דברים שיודעים:**
  - $L \subseteq \text{NL} \subseteq \text{PTIME} \subseteq \text{NP} \subseteq \text{PSPACE} = \text{NPSPACE} \subseteq \text{EXPTIME}$.
  - $L \subseteq \text{NL} \subseteq \text{PTIME} \subseteq \text{co}-\text{NP} \subseteq \text{PSPACE} = \text{NPSPACE} \subseteq \text{EXPTIME}$.
  - $$\begin{array}{ccc} PSPACE & \overset{savitch}{=} & NPSPACE \\ \| & & \| \\ \overline{PSPACE} & \overset{savitch}{=} & \overline{NPSPACE} \end{array}$$
  - $PTIME \neq EXPTIME$
  - $NL = coNL$
- **דברים שלא יודעים:**
  - לא יודעים אם $P \overset{?}{=} NP$
  - לא יודעים אם $P \overset{?}{=} co-NP$
  - לא יודעים אם $NL \overset{?}{=} NP$
  - לא יודעים אם $L \overset{?}{=} NL$
  - לא יודעים אם $NL \overset{?}{=} P$

חלק V

ריכוז דוגמאות ושיטות הוכחה

תכונות סגור של מחלקות

מחלקה	איחוד	חיתוך	חיסור	שרשור	השלמה	*
REG	✓	✓	✓	✓	✓	✓
CFL	✓	✗	✗	✓	✗	✓
RE	✓	✓	✗	✓	✗	✓
R	✓	✓	✓	✓	✓	✓
$coRE$	✓	✓	✗	✓	✗	✓

- ראשי תיבות שעוזרים לזכור תכונות סגור - יותר קל לזכור מה לא מתקיים. עבור CFL לא מתקיים **חחה** (חיבור, חיסור והשלמה) ועבור RE, $coRE$ לא מתקיים **חה** (חיסור והשלמה).

אוטומטים

- **איך מוכיחים פורמלית מהי השפה של אוטומט:**
  - משרטטים את הגרף של האוטומט, מסתכלים על כל מצב ומנסים לזהות בצורה איטואיטיבית מה הוא "זוכר" (כלומר מה החוקיות של המילים שנכנסות אליו).
  - מנסחים מתוך החוקיות שפענחנו לכל מצב q_i הגדרה של תנאי שמתקיים אם"ם $\delta^*(q_0,w)=q_i$ (לדוגמא - $\delta^*(q_0,w)=q_i$ אם"ם w מכילה את האות a מספר זוגי של פעמים). עושים את זה לכל i מ־0 ועד האינדקס של המצב המקבל.
  - מנסחים מתוך המקרה שמסתיים במצב המקבל את ההגדרה של הקבוצה של $L(\mathcal{A})$.
  - מוכיחים באינדוקציה על $|w|$ (הגודל של מילה), כשמקרה הבסיס הוא המילה הריקה $|w|=0$, שעבורו מהגדרת δ^* נקבל $\delta^*(q_0,w)=\delta^*(q,\varepsilon)=q_0$. הנחת האינדוקציה תהיה ש־$|w|>0$, כלומר אפשר לכתוב את w בתור שרשור של u,σ כש־u היא מילה סופית מתוך Σ^* ו־σ היא אות אפשרית שיכולה לסיים את המילה.
  - מחלקים למקרים לפי האפשרויות ש־σ יכולה להיות, ולכל מקרה מוכיחים באינדוקציה את נכונות התנאי שהגדרנו על δ^*.
- **הוכחת סגור תחת הפעלת פעולות על שפות רגולריות:**
  - בונים אוטומט מכפלה מתאים (כלומר מגדירים את האוטומט A_1 של L_1 ואת האוטומט A_2 של L_2, ואז בונים באמצעותם אוטומט מכפלה A - אוטומט שעבור מילה w, מבקר במצב q_1,q_2 אם A_1 ביקרה ב־q_1 אחרי קריאת w ו־A_2 ביקרה ב־q_2 אחרי קריאת w). מגדירים את F, קבוצת המצבים המקבלים של אוטומט המכפלה A, באופן שתואם את ההגדרה של הפעולה שאנחנו רוצים להראות סגור תחתיה.
  - מסבירים למה הרצה של אוטומט המכפלה על מילה שקולה להרצה שלו על האוטומטים A_1,A_2.
  - מראים שמגיעים למצב מקבל באוטומט המכפלה אם"ם מגיעים למצב מקבל לאחר הפעלת הפעולה שרוצים להוכיח את הסגור תחתיה (לדוגמא עבור פעולת האיחוד שהוכחנו בכיתה - מראים שמגיעים למצב מקבל באוטומט המכפלה אם"ם מגיעים למצב מקבל באוטומט של L_1 או באוטומט של L_2, ומסיקים שאוטומט המכפלה מקבל מילה אם היא נמצאת ב־L_1 או ב־L_2, כלומר ב־$L_1\cup L_2$).
- **אלגוריתם למזעור אוטומט דטרמיניסטי (סיבוכיות פולינומיאלית):** עבור אוטומט $A=\langle Q,\Sigma,q_0,\delta,F\rangle$, נגדיר את ה־$DFA$ המינימלי של A בתור $\langle Q',\Sigma,q_0,\delta',F'\rangle$ המוגדרים באופן הבא:
  - Q' היא מחלקות השקילות של $=_A$
  - $\delta'([q],\sigma)=\delta([q],\sigma)$ לכל $\sigma\in\Sigma$
  - F' היא $\{[q]\mid q\in F\}$

שפות רגולריות

- **כל הדרכים להראות ששפה היא רגולרית:**
  - מראים DFA
  - מראים NFA
  - מראים שהשפה סופית

- מראים שקיים לה ביטוי רגולרי
- מראים שיש לה מספר סופי של מחלקות מייהיל־נרוד
- מראים שהרוורס שלה רגולרית
- מראים ששפת החצי שלה $L_{\frac{1}{2}} = \{w \in \Sigma^* \mid w \cdot w \in L\}$ רגולרית
- מראים שהיא ניתנת להכרעה בזמן $O(nlogn)$

- **שפות שהוכחנו שרגולריות:**
  - $\emptyset$
  - Σ^*
  - $\Sigma = \{a,b\}, L_2 = \{a \cdot w \cdot a : w \in \{a,b\}^*\}$
  - $\Sigma = \{0,1,\ldots,9\}, L_4 = \{w \in \Sigma^* : w \text{ is divisible by } 3 \text{ and } \epsilon \notin L_4\}$
  - $\Sigma = \{0,1\}, L_5 = \left\{0^{i_1}10^{i_2}10^{i_3}10^{i_4}10^{i_5}10^{i_6}1 : i_1 > i_2 > i_3 > i_4 > i_5 > i_6 \text{ and } i_1 < 100\right\}$
  - $\Sigma = \{0,1\}, L_6 = \left\{0^{i_1}10^{i_2}10^{i_3}10^{i_4}10^{i_5}10^{i_6}1 : i_1 > i_2 > i_3 > i_4 > i_5 > i_6 \text{ and } i_2 < 100\right\}$
  - $\left\{1^k : k \equiv 0(\text{mod} 3)\right\} \text{ over } \Sigma = \{1\}$

- **כל הדרכים להראות ששפה לא רגולרית:**
  - מניחים בשלילה שהיא רגולרית ומגיעים לסתירה באמצעות תכונות סגור (נגיד ע"י להראות שהיא חיתוך / איחוד וכו' של שתי שפות שאחת מהן לא רגולרית)
  - מראים שקיים קבוע ניפוח p עבורו לכל חלוק xyz לא מתקיימים תנאי למת הניפוח

- **שפות שהוכחנו שלא רגולריות:**
  - $\Sigma = \{a,b\}, L_1 = \{v \cdot u \cdot u : v,u \in \{a,b\}^*, u \neq \epsilon\}$
  - $\Sigma = \{1\}, L_3 = \{1^p : p \text{ is a prime number }\}$
  - $\Sigma = \{0,1\}, L_7 = \left\{0^{i_1}10^{i_2}10^{i_3}10^{i_4}10^{i_5}10^{i_6}1 : i_1 > i_2 > i_3 > i_4 > i_5 > i_6 \text{ and } i_3 < 100\right\}$
  - $\left\{a^ib^jc^k : i+j=k\right\} \text{ over } \Sigma = \{a,b,c\}$
  - $\left\{0^i1^j : i > j\right\} \text{ over } \Sigma = \{0,1\}$
  - $\{a^nb^n \mid n \in \mathbb{N}\}$
  - $L_F = \left\{a^{f(n)} | n \in \mathbb{N}\right\}$ עם $f : \mathbb{N} \to \mathbb{N}$ מונוטונית עולה כך ש־$f(n) \in \omega(n) = \left\{g : \mathbb{N} \to \mathbb{N} \mid \lim_{n\to\infty} \frac{g(n)}{n} = \infty\right\}$.

ביטויים רגולריים

- **אלגוריתם שמתרגם DFAים לביטוי רגולרי:**
  - מתחילים עם DFA עבור השפה.
  - מתרגמים אותו ל־$GNFA$ ששקול לו ומכיל 2 מצבים או יותר (מוודאים שיש מצב התחלתי יחיד, מצב מסיים יחיד, ושהם שונים זה מזה).
  - מתחילים להסיר מצבים מה־$GNFA$ (כלומר להסיר מצב ולהחליף אותו במעבר עם ביטוי רגולרי) עד שאנחנו נשארים עם 2 מצבים (איך להסיר זה החלק המסובך, יש דוגמא בתרגול 3).
  - בסיום (כלומר כשנשארנו עם שני מצבים בלבד ב־$GNFA$) הקשת שמחברת בין שני המצבים שנותרו מיוצגת ע"י ביטוי רגולרי r ששקול ל־A.

שפות חסרות הקשר

- **דרכים להראות ששפה היא CFL:**
  - מראים מפורשות את הדקדוק שלה (אפשר להתבסס על דקדוק של CFL אחרת)
  - מראים שהיא איחוד / חיסור / שרשור של CFL אחרות

- **אלגוריתם שמתרגם CFG כללי לצורה נורמלית של חומסקי:** חשוב לבצע את השלבים לפי הסדר:

1. נוסיף משתנה התחלתי חדש S_0 (שלא בדקדוק המקורי) ונוסיף את הכלל $S_0 \rightarrow S$ (ולעולם לא מוסיפים כלל שגורר את S_0).
2. <u>מוחקים כללי ε</u> - מוחקים כל כלל מהצורה $A \rightarrow \varepsilon$ ומחליפים אותו בכלל חדש (למשל אם $A \rightarrow \varepsilon$ ו-$R \rightarrow AB$, נסיר את $A \rightarrow \varepsilon$ ונוסיף את $R \rightarrow B$). אף פעם לא יוצרים כלל $A \rightarrow \varepsilon$ חדש ל-A שכבר "טיפלנו" בו).
3. <u>מוחקים טרמינלים</u> - מוסיפים לכל $\sigma \in \Sigma$ משתנה חדש $X_\sigma \rightarrow \sigma$, ומחליפים כל מופע של σ בכללים המקוריים ב-X_σ.
4. <u>מוחקים כללים קצרים מידי ("כללי יחידה")</u> - מוחקים כל כלל מהצורה $A \rightarrow B$. בכל פעם ש-$B \rightarrow A$ מופיע, מוסיפים כלל עבור מה ש-B יכול לגרור.
5. <u>מוחקים כללים ארוכים מידי</u> - עבור גרירות מהצורה $A \rightarrow V_1 \rightarrow \cdots \rightarrow V_k$ עם $k \geq 3$, נוסיף משתנים חדשים $u_2 \rightarrow u_{k-1}$ ואת הכללים
$$\begin{array}{c} A \rightarrow V_1 u_2 \\ u_2 \rightarrow V_2 u_3 \\ \vdots \\ u_{k-1} \rightarrow V_{k-1} u_k \end{array}.$$

מכונות טיורינג

- **דרכים להראות ששפה היא $\in \overline{RE \cup coRE}$:**
  - מראים רדוקציה מ-REG_{TM}, INF_{TM} או כל שפה אחרת ב-$\overline{RE \cup coRE}$.
  - מראים שתי רדוקציות - אחת שמראה שהיא לא ב-RE ואחת שמראה שהיא לא ב-$coRE$. אפשר להשתמש ברדוקציה "רגילה" (נגיד לקחת שפה שלא ב-$coRE$ כמו A_{TM} ולהראות רדוקציה ממנה) או בוריאציה "$\overline{A} \leq_m B$ אם"ם $A \leq_m \overline{B}$" (לדוגמא להראות רדוקציה מ-$\overline{A_{TM}} \notin RE$.
- **שפות שהוכחנו שכריעות ($L \in R$):**
  - Σ^*
  - $A_{DFA} = \{\langle A, w\rangle | \quad w \in L(A)\ with\ DFA\ A\}$ קידוד של אוטומט ומילה בשפה שלו.
  - $L_{nice} = \{\langle x, y\rangle : x, y \in \{0,1\}^* \text{and there exists a nice TM that accepts x and rejects y}\}$
  - $INF_{DFA} = \{\langle A\rangle | A\ is\ a\ DFA\ s.t.\ L(A)\ is\ infinite\}$ אוטומטים עם שפה סופית.
- **שפות שהוכחנו ב-$RE \backslash R$:**
  - $A_{TM} = \{\langle M, w\rangle | \text{ w}\in\text{L}\ and, M\ is\ a\ Turing\ machine\}$ מכונות טיורינג.
  - $HALT_{TM} = \{\langle M, w\rangle | \text{ w}\in\text{L}\ s.t.\ M\ is\ a\ TM\ that\ stops\ on\ w\}$ זוגות של M, w כך ש-M עוצרת על w.
  - $HALT^{\varepsilon}_{TM} = \{\langle M\rangle | M\ is\ a\ TM\ that\ stops\ on\ \varepsilon\}$ מכונות טיורינג שעוצרות על ε
  - $HALT^{aba}_{TM} = \{\langle M\rangle | M\ is\ a\ TM\ that\ stops\ on\ aba\}$ מכונות טיורינג שעוצרות על aba
  - $PCP = \{\langle e_1, e_2, .., e_n\rangle\ \ s.t. \text{ there is a match between } e_1, .., e_n\}$
  - $REPEAT_{TM} = \{\langle M, w\rangle | \text{ w}\in\text{L}\ s.t.\ M\ doesnt\ HALT\ on\ w\ and\ repeats\ a\ configuration\}$ זוגות M, w כך שהריצה של M על w עוברת באותה קונפיגורציה פעמיים
  - $\{\langle M, w\rangle | \exists \text{w}\in\text{L}\ s.t.\ M\ accepts\ w\ after\ |w|\ steps\}$ - זוגות M, w כך ש-M מקבלת את w אחרי $|w|$ צעדים.
  - $L_{\geq n} = \{\langle M\rangle | s.t.\ |L(M)| \geq n \in \mathbb{N}\}$ - מכונות טיורינג עם שפה בגודל n או יותר.
- **שפות שהוכחנו ב-$coRE \backslash R$:**
  - $TILE = \{\langle T, V, H, t_{\text{int}}\ \rangle\ s.t.\ \forall \text{n} \geq 1 \text{ there is a tiling } n \times n\ \}$ ריצופים בגודל $n \times n$.
  - $USELESS_{TM} = \{\langle M\rangle\ s.t.\ \exists q \notin \{q_{acc}, a_{rej}\}\ s.t\ \forall w\ M\ doesn't\ pass\ through\ q\}$ מכונות טיורינג כך שיש בהן מצב שלא עוברים בו אף פעם.
  - $L_{\leq n} = \{\langle M\rangle | s.t.\ |L(M)| \leq n \in \mathbb{N}\}$ - מכונות טיורינג עם שפה בגודל n או פחות.
  - $E_{TM} = \{\langle M\rangle | L(M) = \emptyset\}$ מכונות טיורינג שהשפה שלהן ריקה (שלא מקבלות שום מילה).
  - $\{\langle M\rangle | there\ isn't\ w \in \Sigma^*\ s.t.\ M\ rejects\ w\}$ מכונות טיורינג שלכל מילה בשפה, מקבלות אותה או לא עוצרות עליה.
  - $REACH_{TM} = \{\langle M, q\rangle : q \neq q_{\text{acc}} \text{ and } M \text{ reaches the state } q \text{ on every input }\}$ - מכונות טיורינג שלא משנה איזה קלט יקבלו, יגיעו תמיד למצב q במהלך הריצה.
- **שפות שהוכחנו ב-$\overline{RE \cup coRE}$:**

- $REG_{TM}=\{\langle M\rangle|M\ is\ a\ TM\ s.t.\ L(M)\in REG\}$ מכונות טיורינג שהשפה שלהן רגולרית.
- $INF_{TM}=\{\langle M\rangle|M\ is\ a\ TM\ s.t.\ L(M)\ is\ infinite\}$ מכונות טיורינג שהשפה שלהן אינסופית.
- $L_{A_{TM}}=\{\langle M\rangle\,|L(M)=A_{TM}\}$ מכונות טיורינג שהשפה שלהן היא כל הקידודים של מכונת טיורינג עם מילה שהיא מקבלת.
- $NONTRIVIAL_{TM}=\{\langle M\rangle\,|L(M)\neq\emptyset\wedge L(M)\neq\Sigma^*\}$ מכונות טיורינג שהשפה שלהן לא טריויאלית (כלומר לא השפה הריקה ולא כל Σ^*).
- $SUB_{TM}=\{\langle M_1,M_2\rangle\,|L(M_1)\subseteq L(M_2)\}$ שתי מכונות טיורינג כך שהשפה של הראשונה מוכלת בשפה של השניה.
- $\{\langle M_1,M_2,w\rangle\,|M_1\ and\ M_2\ agree\ on\ w\}$ 2 מ"ט ומילה, כך שהמ"ט מסכימות על המילה.
- $ALL_{TM}=\{\langle M\rangle|L(M)=\Sigma^*\}$ מכונות טיורינג שהשפה שלהן היא כל Σ^*.
- $FINITE_{TM}=\{\langle M\rangle|L(M)\ is\ finite\}$ מכונות טיורינג שהשפה שלהן סופית.
- $L_{=n}=\{\langle M\rangle|s.t.\ |L(M)|=n\in\mathbb{N}\}$ - מכונות טיורינג עם שפה בגודל n בדיוק.

- **קבוצות של מכונות טיורינג שהראינו שמקיימות תכונה סמנטית:**
  - כל מכונות הטיורינג M כך שלכל $w\in L(M)$ מתקיים $ww^{rev}\in L(M)$.

סיבוכיות זמן

- **דרכים להראות ששפה היא $\in NP$:**
  - מראים מוודא פולינומיאלי לשפה
  - מראים רדוקציה פולינומיאלית משפה אחרת ב-NP
- **בעיות שהוכחנו שב-$EXPTIME$:**
  - הכרעה שלא קיים מסלול המילטון בגרף מכוון $\overline{D-ST-HAMPATH}=\{<G,s,t>\ \mid \text{G is directed}\ \&\ \neg\exists\ \text{Hamilton path s}\rightarrow t\}$.
  - בדיקת אוניברסיליות של שפה של אוטומט - השפה ALL_{NFA} - כל האוטומטים הא"ד שהשפה שלהם <u>היא</u> Σ^*, כלומר $ALL_{NFA}=\{\langle A\rangle : A \text{ is } NFA \text{ and } L(A)=\Sigma^*\}$.
  - בדיקת אי-אוניברסיליות של שפה של אוטומט - השפה $\overline{ALL_{NFA}}$ - כל האוטומטים הא"ד שהשפה שלהם <u>אינה</u> Σ^*, כלומר $\overline{ALL_{NFA}}=\{\langle A\rangle : A \text{ is } NFA \text{ and } L(A)\neq\Sigma^*\}$.
- **בעיות שהוכחנו שב-P:**
  - הכרעה האם סדרת קודקודים נתונה בגרף מכוון היא מסלול המילטון
  - הכרעה האם מספר נתון הוא פריק $COMPOSITE=\{x\ (binary)\in\mathbb{N}|\ \exists p,q\in\mathbb{N}\ s.t.\ p,q\neq 1\wedge p\cdot q=x\}$
  - בהנתן מספרים x,p, בדיקה האם p מחלק את x ללא שארית
  - בדיקת ריקנות של שפה של אוטומט - השפה $EMPTY_{NFA}$ - כל האוטומטים הא"ד שהשפה שלהם <u>היא</u> השפה הריקה $EMPTY_{NFA}=\{\langle A\rangle :\ \text{A is } NFA \text{ and } L(A)=\emptyset\}$.
  - בדיקת אי-ריקנות של שפה של אוטומט - השפה $\overline{EMPTY_{NFA}}$ - כל האוטומטים הא"ד שהשפה שלהם <u>אינה</u> השפה הריקה $\overline{EMPTY_{NFA}}=\{\langle A\rangle :\ \text{A is } NFA \text{ and } L(A)\neq\emptyset\}$.
  - BAR (*bounded above reachability*) - רביעיות $\langle G,s,t,b\rangle$ כך ש-G גרף עם משקלות חיוביים ושלמים בייצוג אונארי או בינארי, ויש מסלול בגרף בין s ל-t שהמשקל שלו קטן או שווה ל-$b\in\mathbb{N}$.
- **בעיות שהוכחנו שהן NP-שלמות:**
  - השפה SAT - כל הנוסחאות הבוליאניות φ כך ש-φ היא ספיקה.
    - **נוסחא פסוקית בוליאנית CNF:**
      - T,F הן נוסחאות.
      - משתנה בוליאני Var הוא נוסחא.
      - אם φ_1 ו-φ_2 הן נוסחאות, אז גם $\varphi_1\wedge\varphi_2$ ו-$\varphi_1\vee\varphi_2$ הן נוסחאות.
    - **נוסחא ספיקה:** נאמר שנוסחא בוליאנית φ היא ספיקה אם קיימת השמה $f:Var\rightarrow\{T,F\}$ כך שהשערוך של φ לפי f הוא T.
  - השפה $3SAT$ - כל הנוסחאות $3CNF$ הספיקות.
  - השפה $MAX2SAT$ - כל הזוגות $\langle\varphi,k\rangle$ כך ש-φ היא נוסחת $2CNF$ כך שיש השמה מספקת ללפחות k פסוקיות בה.

- IS - מציאת קבוצה בלתי תלויה של קודקודים בגודל k בגרף לא מכוון $IS=\{\langle G,k\rangle \mid \text{ G is undirected with independet set of size } k\leq 1\}$.
  - * **קבוצה בלתי תלויה של קודקודים independent set:** עבור גרף לא מכוון $G=\langle V,E\rangle$, קבוצת קודקודים $S\subseteq V$ נקראת קבוצה בלתי תלויה אם לכל שני קודקודים בקבוצה $v_1,v_2\in S$ אין ביניהם צלע E $\{v_1,v_2\}\notin E$.
- השפה $D-ST-HAMPATH$ - שלשות $\langle G,s,t\rangle$ כך ש-G הוא גרף מכוון עם מסילה המילטונית מ-s ל-t. $D-ST-HAMPATH=\{<G,s,t> \mid \text{G is directed \& } \exists \text{ Hamilton path s} \rightarrow t\}$.
- השפה $U-ST-HAMPATH$ - שלשות $\langle G,s,t\rangle$ כך ש-G הוא גרף לא מכוון עם מסילה המילטונית מ-s ל-t.
- השפה $SUBSETSUM$ - זוגות $\langle A,s\rangle$ של קבוצת מספרים A ומספר יעד s, כך שקיימת קבוצה $B\subseteq A$ כך ש-$\Sigma B=s$.
- השפה $CLIQUE=\{\langle G,k\rangle\,|G\ has\ a\ clique\ of\ size\ k\}$ גרפים עם קליקה בגודל k
- השפה $VC=\{\langle G,k\rangle\,|G\ has\ a\ vertex\ cover\ of\ size\ k\}$ גרפים עם כיסוי קודקודים בגודל k
- השפה $3COLOR=\{\langle 3\rangle\,|G\ could\ be\ colored\ using\ 3\ colors\}$ גרפים $3-$צביעים.
- $SBBR$ (*simple bounded below reachability*) - רביעיות $\langle G,s,t,b\rangle$ כך ש-G גרף עם משקלות חיוביים ושלמים בייצוג אונארי או בינארי, ויש מסלול פשוט בגרף בין s ל-t שהמשקל שלו גדול או שווה ל-$b\in\mathbb{N}$.

- **בעיות שהוכחנו ש$co-NP$ שלמות:**
  - השפה $CONTRADICTION=\overline{SAT}$ - כל הסתירות, כלומר כל הפסוקיות φ כך שאין ל-φ שום השמה מספקת.
  - השפה VAL - כל הטאוטולוגיות, כלומר כל הפסוקיות φ כך שכל השמה מספקת את φ.

סיבוכיות זכרון

- **בעיות שהוכחנו שהן ב-$PSPACE$:**
  - השפה SAT - כל הנוסחאות הבוליאניות φ כך ש-φ היא ספיקה.
- **בעיות שהוכחנו שהן $PSPACE$-שלמות:**
  - בדיקת אוניברסליות - $ALL_{NFA}=\{\langle A\rangle : A \text{ is } NFA \text{ and } L(A)=\Sigma^*\}$.
  - בעיית ההכלה - $CONT_{NFA}=\{\langle A_1,A_2\rangle : L(A_1)\subseteq L(A_2)\, A_1 \text{ and } A_2 \text{ are } NFAs\}$ כאשר נגדיר $L(A_1)\subseteq L(A_2)\iff L(A_1)\cap\overline{L(A_2)}=\emptyset$.
  - השפה $TQBF$ - כל הנוסחאות הבוליאניות שהן מכומתות לחלוטין ונכונות
    - * **נוסחא בוליאנית מכומתת לחלוטין:** נוסחא בוליאנית שכל משתנה בה קשור בכמת $\exists$ או $\forall$ (ולכן מקבלת ערך אמת או שקר).
  - $MIN_{NFA}=\{\langle A,k\rangle \mid A\ has\ an\ equivalent\ DFA\ with\ k\ states\}$ אוטומטים א"ד שיש להם אוטומט דטר' שקול עם k מצבים.
- **בעיות שהוכחנו שהן ב-$NPSPACE$:**
  - השפה $\overline{ALL_{NFA}}=\{\langle A\rangle : A \text{ is } NFA \text{ and } L(A)\neq\Sigma^*\}$.
  - $\overline{PATH}$- שלשות $\langle G,s,t\rangle$ כך ש-G גרף מכוון ולא קיים מסלול בין s ל-t.
- **בעיות שהוכחנו שהן ב-$LOGPSPACE$:**
  - $EQ=\left\{0^k1^k : k\geq 0\right\}\in L$
- **בעיות שהוכחנו שהן NL-שלמות:**
  - $PATH$- שלשות $\langle G,s,t\rangle$ כך ש-G גרף מכוון וקיים מסלול בין s ל-t.
  - $2SAT$ - כל הנוסחאות $2CNF$ הספיקות.
  - SC - כל הגרפים המכוונים הקשירים-חזק (כלומר ניתן להגיע מכל קודקוד לכל קודקוד בגרף - לכל זוג קודקודים בגרף x,y יש מסלול $y\rightarrow x$ וגם מסלול $x\rightarrow y$).
  - $2SC$ - כל הגרפים כך שיש בהם בדיוק 2 רכיבי קשירות חזקה (קודקודים x,y נמצאים באותו רכיב קשירות חזקה אם יש ב-G מסלול $y\rightarrow x$ וגם מסלול $x\rightarrow y$).
  - BAR (*bounded above reachability*) - רביעיות $\langle G,s,t,b\rangle$ כך ש-G גרף עם משקלות חיוביים ושלמים בייצוג אונארי או בינארי, ויש מסלול בגרף בין s ל-t שהמשקל שלו קטן או שווה ל-$b\in\mathbb{N}$ שנתון באונרית.
  - BBR (*bounded below reachability*) - רביעיות $\langle G,s,t,b\rangle$ כך ש-G גרף עם משקלות חיוביים ושלמים בייצוג אונארי או בינארי, ויש מסלול בגרף בין s ל-t שהמשקל שלו גדול או שווה ל-$b\in\mathbb{N}$ שנתון באונרית.