בס"ד

אוניברסיטת אריאל
המחלקה למדעי המחשב

שם הקרוס: אוטומטים ושפות פורמליות 1

מס' קורס: 7029010
מרצה: ד"ר מירה גונן
העורכים: יצחק שרון, משה חנוקוגלו ,אביב רובשיץ, הדר אלון, מירב בוים, יובל מזרחי, נעמה שטאובר, דביר ברזילי
תאריך: תשע"ח סמ' א

קובץ זה הינו דף נוסחאות בקורס אוטומטים ושפות פורמליות 1.
דף נוסחאות זה כולל הגדרות, כללים ודוגמאות של תרגילים.

הגדרות

א"ב– קבוצה **סופית** נתונה של אותיות (=סימנים, תווים).
א"ב יסומן: $\Sigma = \{a, b, 0, @\}$.
מילה– היא סדרה סופית של אותיות מעל הא"ב.
המילה הריקה– היא סידרה ריקה של תווים. ותסומן ε.
אורך מילה– מספר אותיות המילה. נסמן $|w| =$ אורך המילה w.
קבוצת כל המילים- מעל הא"ב Σ תסומן Σ^*.
שפה L- היא תת קבוצה של Σ^* כלומר $L \subseteq \Sigma^*$ יתכן ש L היא קבוצה ריקה או ש $L = \Sigma^*$.
לדוגמה:

$$\Sigma = \{a\} \Rightarrow \Sigma^* = \{\varepsilon, a, aa, aaa, \dots\} \Rightarrow |\Sigma^*| = \aleph$$

פעולות על מילים

שרשור של מילים – תהינה u,v מילים מעל א"ב Σ השרשור של u ואח"כ v שנסמנו u·v היא המילה שמתקבלת מכתיבת v אחרי u (קוראים מילה משמאל לימין).
הערה: $\mathrm{u} \cdot \mathrm{v} \neq \mathrm{v} \cdot \mathrm{u}$,
$\varepsilon \cdot \mathrm{u} = \mathrm{u} \cdot \varepsilon$,
$\mathrm{w} \cdot (\mathrm{u} \cdot \mathrm{v}) = (\mathrm{w} \cdot \mathrm{u}) \cdot \mathrm{v} = \mathrm{w} \cdot \mathrm{u} \cdot \mathrm{v}$
חזקה – תסומן $n \in \mathbb{N}, w^n = \underbrace{w \cdot w \cdots w}_{n \text{ פעמים}}$. לכן גם אפשר לומר $(ab)^n = \underbrace{(ab) \cdot (ab) \cdots (ab)}_{n \text{ פעמים}} = a \cdot (ba)^{n-1} b = a(ba)^{n-1}b$ $= (baba \cdots ba) \cdot a$.
הגדרה רקורסיבית לחזקה - $\begin{cases} w^0 = \varepsilon, \ n = 0 \\ w^n = w \cdot w^{n-1}, \ \mathrm{n} \geq 1 \end{cases}$

פעולות על שפות

איחוד - $L_1 \cup L_2 = \{\omega \in \Sigma^* \mid \omega \in L_1 \text{ או } \omega \in L_2\}$.
חיתוך - $L_1 \cap L_2 = \{\omega \in \Sigma^* \mid \omega \in L_1 \text{ וגם } \omega \in L_2\}$.
משלים - $\bar{L} = \{\omega \in \Sigma^* \mid \omega \notin L\}$.
הפרש – $L_1 \backslash L_2 = L_1 \cap \overline{L_2}$
$= \{\omega \in \Sigma^* \mid \omega \in L_1 \text{ וגם } \omega \notin L_2\}$.
שרשור – $L_1 \cdot L_2 =$
$\{\omega \in \Sigma^* \mid \exists u \in L_1, \exists\, v \in L_2, \omega = u \cdot v\}$.
לדוגמה:
$\mathrm{L}_1 = \{\mathrm{aab}, \mathrm{abb}\}, \mathrm{L}_2 = \{\mathrm{ba}, \mathrm{baa}\} \Rightarrow L_1 \cdot L_2 = \{aabba, aabbaa, abbba, abbbaa\}$.
הערה: $\begin{cases} L \cdot \emptyset = \emptyset \cdot L = \emptyset \\ L \cdot \{\varepsilon\} = L \end{cases} = \emptyset \neq \{\varepsilon\}$
הפעולות $L \cdot \varepsilon$ וגם $w \cdot \emptyset$ אינן חוקיות, כיוון שא"א לשרשר קבוצה ומילה.

$$L_1 \cdot L_2 \neq L_2 \cdot L_1$$
$$(L_1 \cdot L_2) \cdot L_3 = L_1 \cdot (L_2 \cdot L_3)$$

איטרציה - $L^* = \bigcup_{i=0}^{\infty} L^i \, ; L^0 = \{\varepsilon\}$

דוגמאות:

1. שפת כל המילים המתחילות בab מעל $\Sigma = \{a, b\}$ $\Leftarrow \{ab\}\Sigma^*$.
2. שפת כל המילים שמכילות ab $\Leftarrow \Sigma^*\{ab\}\Sigma^*$.
3. שפת כל המילים באורך זוגי מעל $\Sigma = \{a\}$ $\Leftarrow \{aa\}^*$ או $\bigcup_{k \in \mathbb{N}} \{a\}^{2k}$ או $\{\Sigma\Sigma\}^*$
4. שפת כל המילים מעל $\Sigma = \{a, b\}$ שמכילות a אחת בלבד $\Leftarrow \{b\}^*\{a\}\{b\}^*$
5. שפת כל המילים מעל $\Sigma = \{a, b\}$ שמכילות ab ולא bb $\Leftarrow \left(\Sigma^*\{ab\}\Sigma^*\right) - \left(\Sigma^*\{bb\}\Sigma^*\right)$

אוטומט הגדרות:

אוטומט הוא חמישיה של איברים:
$A = (\Sigma_A, Q_A, q_{0A}, F_A, \delta_A)$
Σ_A – א"ב הקלט.
Q_A – קבוצה סופית לא ריקה של מצבים.
q_{0A} – המצב שממנו מתחילים לבדוק את הקלט. ($q_{0A} \in Q_A$)
F_A – הקבוצה של המצבים שמקבלים. ($F_A \subseteq Q_A$)
δ_A – פונקציית מעברים. הפונקציה מקבלת מצב ואות מהא"ב ומחזירה את המצב הבא בתור.
אס"ד (להבדיל מאסל"ד) צריך לאפשר יציאה מכל מצב עבור כל אות בא"ב.
אוטומט אס"ד שקול לאוטומט אסל"ד שקול לאסל"ד עם מסעי אפסילון

המרה מאסל"ד לאס"ד אלגוריתם:

1. התחל מq_0
2. עבור כל אות קלט עבורה מוגדרת $\delta(q_0, \sigma)$ בדוק לאלו מצבים ניתן להגיע באסל"ד.
3. הקצה מצב חדש באס"ד, שיהווה "אוסף המצבים האפשריים עבור אות זו" ואז חבר חץ מq_0 למצב זה.
4. חזור על התהליך לכל מצב ולכל אות קלט.
5. כל מצב חדש כזה שלפחות אחת האופציות שלו היא מצב מקבל ב NFA, יהיה מצב מקבל ב DFA.
6. הקצה מצב בור לכל המצבים מהם לא הוגדרה $\delta(q, \sigma)$ לאות σ.

רגולריות:

הגדרה: שפה L היא רגולרית אם היא מתקבלת ע"י אוטומט סופי דטרמיניסטי.
איך מזהים שפה רגולרית:
$\sum^*$, $\emptyset$ -רגולריות
כל שפה סופית רגולרית
"גודל" של שפה אינו מצביע על רגולריות, או לא.
כל שפה שהמשלים שלה רגולרי- רגולרית.
שפה רגולרית היא שפה שלא צריך "לזכור" או לספור יותר מכמות סופית.
$\{a^{f(n)} | n \in N\}$ – לא רגולרית אלא אם כן f(n) פונקציה לינארית.

תכונות סגור:

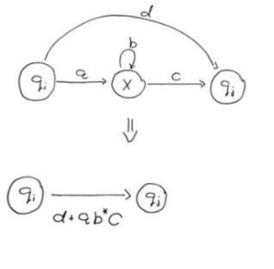
משלים (בבנית אוטומט,מצבים מקבלים ולא מקבלים מתהפכים),
איחוד-סופי (נוכיח ע"י פעמיים משלים ודמורגן), **חיתוך-סופי**(אוטומט מכפלה), **איטרציה, היפוך, שירשור, חיסור**.
(שימושי להוכחה: דמורגן) חיתוך או איחוד אין סופי וכן הכלה אינן תכונת סגור!
דוגמא שהכלה לא שומרת רגולריות
לא רגולרי – $\{0^n 1^n | n \in \mathbb{N}\} \subseteq \{01\}$ – רגולרי,
רגולרי – $\Sigma^* \subseteq \{0^n 1^n | n \in \mathbb{N}\}$ – לא רגולרי.
כל שפה לא רגולרית אז גם המשלימה שלה לא רגולרית וגם ההיפוך שלה לא רגולרי.
השפה $\mathrm{L} = \{\mathrm{a^n b^n} \mid \mathrm{n} \geq 1\}$ לא רגולרית!
השפה $\mathrm{L} = \{\mathrm{a^p} | \text{ראשוני } \mathrm{p}\}$ לא רגולרית!
$\mathrm{L} = \{a^n b^m c^k | n \leq m + k\}$ לא רגולרית!
$\mathrm{L} = \{(01)^n (10)^n | n \in \mathbb{N}\}$ לא רגולרית!
$\mathrm{L} = \{a^{2p} | p \textit{ is prime}\}$ לא רגולרית!
$\mathrm{L} = \{\ \mathrm{L} = \{ww^R | w \in \sum*\}$ לא רגולרית!
$\mathrm{L} = \{a^{2p} | p \textit{ is odd}\}$ רגולרית!
הוכחת רגולריות ע"י בניית אס"ד או מציאת ביטוי רגולרי.

תבנית להפרכה רגולריות עם למת הנפוח:

"נניח בשלילה כי L רגולרית, לכן היא מקיימת את למת הניפוח, לכן לפי הלמה קיים קבוע $\mathrm{n} \in \mathbb{N}$ כך שלכל $|z|$≥n קיים פירוק uvw כך ש $|uv|$≤n , 1≤$|v|$≤n שמקיים עבור כל $i \in \mathbb{N}$, $\mathrm{uv}^i w \in L$.
יהא $L \in$__=z , ≥__ = $|z|$ n , נראה שקיים $i = \_\_ \in \mathbb{N}$ עבורו המילה $\mathrm{uv}^i w$ אינה ב L- (לדוגמא)
$\mathrm{uv}^i w = \cdots$ (לדוגמא ב $a^n b^n$ היא $a^{n+(i-1)*|v|} b^n$) "

ביטויים רגולרים:

בביטוי המבטא שפה רגולרית ניתן להשתמש בתווים $\{\Sigma, \square^*, \cdot, (,), +, \square^+\}$
אלגורתם לבנית ביטוי רגולרי:
1)בונים לשפה אוטומט כמה שיותר אי דטרמיניסטי
2) מיצרים מצב מקבל יחיד (ע"י מסעי אפסילון)
3)האלגוריתם עצמו:
*בכל שלב מוחקים מצב שאינו ההתחלתי ואינו המקבל ובודקים מה המסלול באורך 2 שמגיע מכל מצב לכל מצב (ששונה ממה שמחקנו) דרך אותו מצב שמחקנו. שאר המצבים אותו הדבר.



דוגמאות להוכחות

תהא $L = \{w \in \{0,1\} | \#_0 \equiv \#_1 (mod 3)\}$ נוכיח ש L מקיימת את למת הניפוח.
נבחר n=3 (בד"כ כמספר מצבי האוטומט)
תהי $z \in L$ מילה בגודל $|z| \geq 3$
אם z מתחילה ב 111 או 000:
נבחר $u = \varepsilon$, v נבחר כשלושת התווים הראשונים, ו w תיהיה שאר המילה. נראה שהפירוק מקיים את התנאים:
1) $|uv| = 3 \leq 3$
2) $|v| = 3 \geq 1$
3) יהא $i \in N$ $\mathrm{z}' = uv^i w$
בה"כ (נוכיח על z מתחילה ב 000 והוכחה תיהיה אותו הדבר על 111)
$\#_0(z') \equiv \#_0(z) + 3(i-1)$
לכן מכיוון ש $z \in L$ אז $\#_0(z) \equiv \#_1(z)(mod 3)$
ולכן $\#_0(z') \equiv \#_1(z')(mod 3)$
מכאן $z' \in L$
אחרת z מכילה ב-3 תווים הראשונים רצף של 01 או 10:
נבחר u- מה שלפני הרצף, v –הרצף, w– מה שאחרי הרצף.
1) $|uv| \leq 3$
2) $|v| \geq 2 \geq 1$
3) יהא $i \in N$ $\mathrm{z}' = uv^i w$
$\#_0(z') \equiv \#_0(z) + (i-1)$
$\#_1(z') \equiv \#_1(z) + (i-1)$
לכן מכיוון ש $z \in L$ אז $\#_0(z) \equiv \#_1(z)(mod 3)$
ולכן $\#_0(z') \equiv \#_1(z')(mod 3)$
מכאן $z' \in L$

דוגמה להפרכת למת הניפוח לשפה

L=$\{a^{n^2+15n+7} \ |n \in N\}$
נניח בשלילה ש-L רגולרי, ויהי n הקבוע המובטח מהלמה.
נבחר z=$a^{n^2+15n+7}$.
$|z| = n^2 + 15n + 7 > n$ ולכם קיים פירוק z=uvw∈ L כך ש: $|uv| \leq n$, $|v| \geq 1$. נראה שקיים i∈ N עבורו המילה המנופחת אינה בשפה .
נבחר i=2 ונקבל:

$$uv^i w = a^{n^2+15n+7+(i-1)|v|} = a^{n^2+15n+7+|v|}$$
$$n^2 + 15n + 7 < n^2 + 15n + 7 + |v|$$
$$\leq n^2 + 16n + 7$$
$$< n^2 + 2n + 1 + 15n + 15 + 7$$
$$= (n+1)^2 + 15(n+1) + 7$$

ולכן המילה המנופחת אינה בשפה. בסתירה ללמת הניפוח. לכן השפה אינה רגולרית.

הוכיחו את הגרסה הבאה של למת הניפוח

תהי L שפה רגולרית. אזי קיים n ב-N כך שלכל מילה z ב-L שאורכה לפחות n ,קיים פירוק מהצורה z=uvw כאשר:
- |uv|≤n
- 2≤|u|,1≤|v|
- $uv^i w \in L$ לכל 0≤i טבעי

נוכיח את הגרסה הזו של הלמה. ההבדל מהגרסה המקורית, הוא שאם ניקח את n להיות כמות המצבים, יכולה להיות לנו לולאה במצבים הראשוניים, ואז לא נצליח להוכיח, בגלל ש |u|≥2.לכן ניקח קבוע אחר:
תהי L שפה רגולרית. לכן, קיימים לה DFA $A = (\Sigma_A, Q_A, q_{0_A}, F_A, \delta_A)$ נסמן $|\mathrm{Q}| = \mathrm{n}$ תהי Z∈Lמילה המקיימת $|\mathrm{z}| \geq \mathrm{n} + 4$ (אם אין כזו, הלמה מקיימת באופן ריק וסיימנו).נתבונן ב n+4 הרישות הראשונות של z:

$$a_0 = \varepsilon \, , a_1 = z_1 \, , a_2 = z_1 z_2 \, ,$$
$$a_3 = z_1 z_2 z_3 \dots \, a_{n+3} = z_{1\dots} z_{n+2} z_{n+3}$$

היות ויש n+3 רישות ורק n מצבים, יש לפחות 4 רישות שבניהן יש הגעה לאותו מצב במסלול החישוב לפי עקרון שובך היונים (4 זוגות של רישות שמגיעות לאותו מצב, או שלישיית רישות שמגיעה לאותו מצב ועוד 2 זוגות רישות שמגיעות לאותו מצב, או 4 רישות שמגיעות לאותו מצב, או שתי שלישיות).נבחר שתיים מתוכן המקיימות $i, j \geq 2$ (במקרה הכי גרוע ,הרישות האלו הן a_0, a_1, a_2, a_3 ולכן a_2, a_3 עונות על הדרישה.)
יהי q' מצב באוטומט המקיים $(q_0, a_i) = (q_0, a_j) = q'$ בלי הגבלת הכלליות, נניח i<j בבירור, a_i היא רישא של a_j , a_j היא רישא של z. נגדיר $u = a_i$, $v = a_j / a_i$ ולכן מתקיים:

$$\delta\,(q_0, a_i) = \delta\left(\delta\,(q_0, a_i), a_j/a_i\right) =$$
$$\delta\left(q', a_j/a_i\right) = q'$$

אכן מתקיים 1≤|v| כנדרש, שהרי i<j. כלומר נוכל לחזור על קריאת v כמה פעמים שנרצה ולכן הלמה מתקיימת:

$\delta(q_0,uvw) = \delta(\delta(\delta(q_0,u),v),w)$
$= \delta(\delta(q',v),w) = \delta(q',w) = q_F$

לסיכום: הראינו שעבור כל $Z \in L$ המקיימת $|Z| \geq n+4$ הלמה מתקיימת, ולכן הוכחנו את הגרסה הזו של למת הניפוח.

<u>הוכיחו/ הפריכו L={ a^i b^j| i>j>0}</u>

הפרכה: נניח בשלילה שL רגולרית. יהי n הקבוע המובטח בלמה. נבחר $Z=a^n b^n$ נבחר i=0 ונקבל $a^{|v|-1}b^n$ שהוא לא שייך לL.

<u>הוכיחו או הפריכו L={ a^n|n פריק} ניתנת לניפוח אך לא רגולרית.</u>

הוכחה: קיים n טבעי כך שלכל |z|<n קיים פירוק z=uvw עבורו מתקיים |uv|<n וגם |v|>1. נבחר u=ε נבחר n=6: יהי |z| >6 וגם z שייך לL. במקרה ו|Z| זוגי אזי: נבחר $w=a^{|z|-2}$, $v=a^2$, u=ε, וגם עבור כל i>1 נקבל שמילה מוכלת בשפה $a^{2i+|v|-2}$ כי הוא תמיד זוגי.
עבור |z| אי זוגי: |z|=m*k עבור k>2 אזי m פריק ולכן נבחר u=ε , $v=a^m$, ואז $w=a^{m(k-1)}$ בנוסף עבור כל i מתקיים: $w=a^{m(k-1+i)}$.לעומת זאת L אינה רגולרית ולכן גם L לא רגולרית.

<u>הוכיחו שהשפה ניתנת לניפוח אך לא רגולרית.</u>

<u>$L = \{b^k a^{4n+n^2} | k,n > 0\} \cup \{b,b^2,b^3\} \cup \{a^*\}$</u>

הוכחה: יהי *n* הקבוע המובטח בלמה, נבחר *k>n* ואת המילה $z = b^k a^{n^2} \in L$, $n \leq |z|$ ולכן קיים פירוק *z=uvw*, היות ו*k>n* מתקיים ש $v=b^{|V|}$ ואז לכל *i* מתקיים:
$z=uv^iw=b^{k+|v|(i+)1}a^{4n+n^2} \in L$
נראה כי *L* אינה רגולרית: נניח בשלילה ש*L* שפה רגולרית ולכן גם L^R רגולרית (מתכונות סגור). יהי *n* הקבוע המובטח בלמה נבחר במילה $z = a^{4n+n^2} * b \in L$ ונבחר *i=2* ונקבל:
$z = uv^2w = a^{4n+n^2+|v|}b$ ואז נקבל $n^2+4n < n^2 +$ $4n + |v| < (n+1)^2 + 4(n+1)$ ולכן המילה המנופחת לא נמצאת בשפה ולכן L^R אינה רגולרית ולכן גם L אינה רגולרית.

<u>הוכיחו או הפריכו $\mathbf{L = \{wa^{|w|} | w \in \{a,b\}^*\}}$</u>

הפרכה: נניח בשלילה..(תבנית למת הניפוח) נבחר $z=a^nb^na^n$ והניפוח הוא: $z=a^{n-|v|}b^na^n$ *המילה לא שייכת* לL.

<u>אם r1 מופיע כתת מחרוזת של r2 אזי בהכרח מתקיים $L[r1] \subseteq L[r2]$.</u>

תשובה: לא, דוגמא נגדית: $r1=(a+b)^*$ *ונבחר* $r2=b(a+b)^*$
r1מופיע כתת מחרוזת של r2 אבל $\varepsilon \in L[r1]$ *אבל*
$\varepsilon \notin L[r2]$

<u>הוכיחו או הפריכו רגולריות $\mathbf{L = \{a^nb^m | n \neq m\}}$</u>

תשובה*: כדי להפריך נצטרך לנפח בצורה כזאת שיהיה שוויון בין כמות ה a-ים לבין כמות הb-ים. זה ידרוש קצת עבודה, נעזר בפונקציית עצרת. נניח בשלילה שהשפה L רגולרית, ויהי n הקבוע המובטח מלמת הניפוח. נבחר* $z = a^{n!}b^{2(n!)}$ *כך ש*$z \in L$, $|z| = 2(n!) + n! > n$ *ולכן קיים פירוק z=uvw כך ש* $|uv| \leq n$, $|v| \geq 1$ *נראה שקיים* $i \in \mathbb{N}$ *עבורו* $uv^iw \notin L$. $uv^iw = a^{n!+(i-1)|v|}\, b^{2(n!)}$
כדי שהמילה המנופחת לא תהיה בשפה נרצה לקיים:
$n! + |v|(i-1) = 2(n!)$ *ולכן* $|v|(i-1) = 2(n!) - n! = n!$ *ולכן* $i = \frac{n!}{|V|} + 1$. *היות ו*$1 \leq |v| \leq n$ *הגודל של V תמיד מחלק את !n ולכן נקבל* $i \in \mathbb{N}$ *עבורו המילה המנופחת לא בשפה. ולכן L אינה רגולרית.*

<u>תהיינה L₁L₂ שפות. נגדיר Shuffle באופן הבא:</u>
<u>$Shuffle(L_1, L_2) = \{u_1v_1u_2v_2 \dots u_nv_n | n \geq 1, u_1..u_n \in L_1\ v_1 \dots v_n \in L_2\}$</u>

הוכיחו או הפריכו:

א. אם L_1L_2 רגולרית אז גם $Shuffle(L_1,L_2)$ רגולריות.

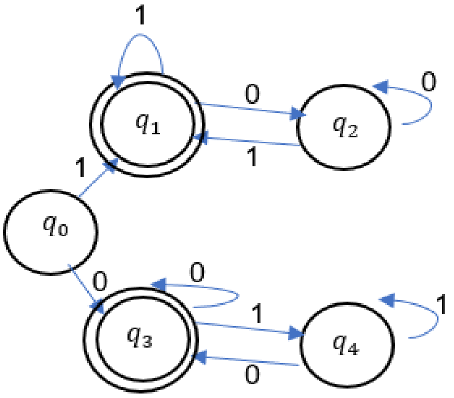
ב. אם $Shuffle(L_1,L_2)$ רגולרית אז בהכרח L_1L_2 רגולרית.

א. הטענה נכונה השפה $Shuffle(L_1,L_2) =$ $(L_1 \cdot L_2)^n$ *ולכן , אם* $L_1\ L_2$ *רגולרית גם האיחוד שלהם רגולרי, ואיטרציה n פעמים נשארת שפה רגולרית, שכן n מספר סופי.*

ב. אם $Shuffle(L_1,L_2)$ *רגולרית, לא בהכרח ש*$L_1\ L_2$ *רגולריות . נראה דוגמא נגדית:*
$L_1 = \{a^p | p\ is\ prime\}$ ו
$L_2 = \{a^c | c\ isn't\ prime\}$ **אז**
$Shuffle(L_1,L_2) = \{a\}^*)$ *אך 2 השפות אינן רגולריות .*

<u>הוכיחו או הפריכו רגולריות</u>

L(A) = {w∈ ∑*| מתחילה ומסתיימת באותה אות w}



הוכחה: תהי
L = {w∈ {0,1}*| מתחילה ומסתיימת באותה אות}
נראה כי לכל n מתקיים $\forall x \in \{0,1\}^n\ x \in L(A) \leftrightarrow x \in L$
בנוסף נוכיח טענה חזקה יותר, שלכל x המתחיל ב '0' $\delta(q_0,x) \in \{q_0,q_4\}$ ולכל x המתחיל ב '1' $\delta(q_0,x) \in \{q_1,q_3\}$
באינדוקציה על n:
<u>בסיס:</u> ניקח n=0 ← $x = \varepsilon$ ולכן $q_0=\delta(q_0,x) \notin F$, $\varepsilon \notin L$
נניח כי הטענה נכונה עבור n ונוכיח עבור n+1 ($x \in \{0,1\}^{n+1}$).
נניח $x \in L(A)$:
מקרה 1: $\delta(q_0,x) = q_1$: נרשום x = wb כאשר |w|=n
נראה כי x מתחיל ומסתיים ב1:
נניח בשלילה כי b=0. היות ומתקיים $\delta(q_0,x) = q_1$ נובע ש: $\delta(q_0,w) \in \{q_0,q_1,q_3\}$ ובשלושת המקרים לא נגיע לq_1 שכן $\delta(q_0,w0) \in \{q_2,q_3\}$ נניח בשלילה כי w מתחיל ב '0':
לפי הנחת האינדוקציה $\delta(q_0,w) = \{q_2,q_4\}$ ולכן $\delta(q_2,wb) \neq \{q_2,q_4\}$ בסתירה להנחה ש $\delta(q_0,x) = q_1$.
לכן x מתחיל ומסתיים ב '1', לכן x∋L.
מקרה 2: $q_3 = \delta(q_0,x)$ נשנה מספרים ממקרה 1ומספיק דומה.

<u>הוכיחו $(L_1 \cup L_2)^*$=$(L_1{}^*L_2{}^*)^*$</u>

נוכיח ע"י הכלה דו כיוונית:
כיוון ראשון: תהי $w \in (L_1 \cup L_2)^*$ מכאן:
קיים $n \in N$ כך ש:
$w=u_1u_2u_3 \dots u_n$ כאשר $u_i \in L_1 \cup L_2$ לכל $1 \leq i \leq n$.
בלי הגבלת הכלליות אם $u_i \in L_1$ נבחר $\varepsilon \in L_2$, $u_i \in L_1{}^*$
ולכן: $u_i \in L_1{}^*L_2{}^*$.
מכאן, לפי הגדרת איטרציה: $w \in (L_1{}^*L_2{}^*)^*$.
כיוון שני: תהי $w \in (L_1{}^*L_2{}^*)^*$ מכאן:
קיים $n \in N$ כך ש:
$w=u_1u_2u_3 \dots u_n$ כאשר $u_i \in (L_1{}^*L_2{}^*)$
לכן, קיימים $k_1,k_2 \in N$ כך ש:
$u_i = v_{11}v_{12} \dots v_{1k1}.v_{21}v_{22} \dots v_{2k2}$
כאשר $v_{1j} \in L_1$ $v_{2j} \in L_2$
ולכן: w מורכבת מכל תתי המילים האלו כאשר כל אחת נמצאת ב- L_1 או ב-L_2
לכן לפי הגדרת איטרציה: $w \in (L_1 \cup L_2)^*$

<u>עבור שפה L נגדיר $Subs(L) = \{y \in \sum^*, z \in \sum^*\ s.t\ xyz \in L\}$ כלומר, *subs(L)* היא שפה המכילה את כל תתי המילים של המילים בשפה *L*. הוכח\ הפרך: אם *L* שפה לא רגולרית אזי *subs(L)* בהכרח לא רגולרית.</u>

תשובה: תהי $L = \{w = a^nb^n | n \in \mathbb{N}\}$ השפה *L* אינה רגולרית, לכן $subs(L) = \{a\}^*\{b\}^*$ ולכן רגולרית.

<u>הוכח את השוויון $(L^*)^* = L^*$</u>

כיוון ראשון:
תהי $w \in L^*$ נראה כי $w \in (L^*)^*$
הטענה מתקיימת באופן טריוויאלי, כי נחבר להתייחס לאיטרציה החיצונית בתור "פעם אחת", ואז קיבלנו $L^* = L^*$
כיוון שני: תהי
$w \in (L^*)^*$ נראה כי $w \in L^*$
$w = b_1b_2 \dots b_n$ כאשר $\forall i \in [n] : b_i \in L$
לפי הגדרה, מתקיים שלכל *i* ניתן לפרק את b_i בצורה הבאה :
ולכן : $\forall i \in [n]\ c_i \in L$ $\forall i \in [n]: b_i = c_1^ic_2^i \dots c_n^i$
$w=c_1^1c_2^1 \dots c_n^1c_1^2c_2^2 \dots c_n^2 \dots c_n^n$
ולכן $w \in L^*$

<u>הוכח או הפרך: לכל שפה לא רגולרית L ,כל תת שפה אינסופית שלה היא גם לא רגולרית.</u>

תשובה: נפריך ע"י טענה נגדית, עבור השפה: $L = \{a^nb^n | n \leq 1\} \cup \{a\}^*$ היא שפה לא רגולרית, אך תת שפה אינסופית שלה היא $L = \{a\}^*$ היא שפה רגולרית.
נוכיח על ידי דוגמא נגדית:

<u>הוכח או הפרך: קיימת שפה לא רגולרית *L*, כך שלכל תת שפה אינסופית שלה היא גם לא רגולרית.</u>

תשובה: עבור השפה $L = \{a^p | p\ is\ prime\}$ כל תת שפה אינסופית שלה היא גם כן לא רגולרית (שכן כל תתי השפות של *L* היא *L* עצמה).

<u>הוכח או הפרך: תהי השפה L לא רגולרית, השפה $\mathbf{L^2\Sigma^*}$ היא בהכרח רגולרית.</u>

תשובה: הטענה אינה נכונה, נבחר $L = \{a^nb^nc | n \in \mathbb{N}\}$ ולכן $L^2 = \{a^nb^nca^mb^mc | m,n \in \mathbb{N}\}$ כעת בשרשור עם Σ^* עדיין נצטרך לשמור על התבנית a^nb^nc ברישא של המילה , ולכן לא נוכל לאפס את L^2 בעזרת אפסילון (ולכן א"א לקבל את Σ^* מהשרשור).

<u>הוכח או הפרך: תהי השפה L לא רגולרית, השפה $\mathbf{\Sigma^*\ (L/\{\varepsilon\})}$ היא בהכרח לא רגולרית.</u>

תשובה: נסתור את הטענה על ידי דוגמא נגדית: נבחר $L = \{a^p | p\ \ is\ a\ prime\ number\}$ השפה אינה רגולרית . כעת ε אינו בשפה ולמרות זאת נוכל לבנות ביטוי רגולרי לשפה $L\Sigma^*$: $L = a^2(a+b+c)^*$

<u>הוכח או הפרך: קיימת שפה רגולרית L כך שהשפה $L' = \{ww : w \in L\}$ רגולרית.</u>

תשובה: הטענה נכונה, בניגוד להוכחות הרגילות , כאן התבקשנו להראות "קיימת שפה", ולכן דוגמא בודדת תספיק: ניקח את השפה $L = \{a^n | n \in \mathbb{N}\}$ השפה רגולרית. נסתכל על $L' = \{a^na^n | n \in \mathbb{N}\}$ כלומר זוהי השפה $L = \{a^{2n} | n \in \mathbb{N}\}$, קל לבנות לשפה הזאת אוטומט הסופר זוגיות של מספר ה-*a*ים

<u>הוכח או הפרך: קיימת שפה רגולרית L כך שהשפה $L' = \{ww : w \in L\}$ לא רגולרית.</u>

תשובה: הטענה נכונה, גם כאן מספיק דוגמא אחת שמקיימת את הטענה. ניקח את השפה $L = \{0^n1^m | n - m(mod5) = 0\}$ כעת על מנת לקרוא מילה כפולה נזדקק לזיכרון שיגיד מה גודל המילה הקודמת. כלומר $ww = 0^n1^m0^n1^m$.
נניח בשלילה שהשפה 'L היא שפה רגולרית. ולכן קיים קבוע n מלמת הניפוח . נבחר את המילה $z = 0^{5n+1}10^{5n+1}1 \in L$ $|z| \geq n$ ולכן קיים פירוק *z=uvw* העונה על תנאי הלמה. נבחר *i=0*:
$uv^iw = 0^{5n+1-|v|}10^{5n+1}1 \notin L$ המילה שנוצרה לאחר הניפוח אינה בשפה משום ש
$5n+1-|v| < 5n+1$ לכן, לכל פירוק אפשרי הראינו ש *i* טבעי עבורו המילה הנופחת אינה בשפה, ולכן השפה *L'* אינה רגולרית.

**<u>הוכח או הפרך: תהי *L* שפה הבאה:
$L = \{w \in \Sigma^* | |w| \geq 1000\}$ ותהי *P* שפה לא רגולרית כלשהי. אזי בהכרח: $L \cup P$ רגולרית.</u>**

מנוסחת דה מורגן נקבל:
השפה $L \cup P = (L \cup P)' = (L' \cap P')$ $L' =$ $\{w \in \Sigma^* | |w| < 1000\}$ ולכן היא סופית. בחיתוך עם *P'* נשאר עם כמות סופית של מילים . ולכן $(L' \cap P')$ רגולרית מסגירות למשלים נקבל $L \cup P$ רגולרית.

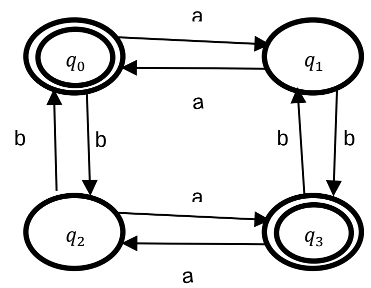
הוכח או הפרך: השפות הלא רגולריות סגורות לפעולות חיתוך.

תשובה: נגרום לחיתוך של השפות להיות קבוצה ריקה או רק אפסילון.

דוגמא נגדית: $L_1=\{a^nb^n|n\in\mathbb{N}\}$, $L_2=\{c^nd^n|n\in\mathbb{N}\}$ השפות אינן רגולריות, אבלם $L_1\cap L_2=\emptyset$ והשפה הריקה גם היא רגולרית.

האוטומט לשפה:

$L=\{w\epsilon\{a,b\}^*|\#_a(w)\equiv\#_b(w)(mod2)\}$ הוא:



הוכחה שהינה רגולרית:

$L=\{a^nb^mc^kd^l|1<n<3,\ m=2n,\ k\ mod3=1,\ l\ mod2=0\}$

נבנה אס"ד שיכריע את הספה: נשים לב כי n=2, m=4 .

$$A=(\{q_0,q_1,q_2,q_3,q_4,q_5,q_6,q_7,q_8,q_9,q_{10},q_{pit}\},\sum=\{a,b,c,d\},q_0,\delta,F)$$

$\delta(q_0,a)=q_1$, $\delta(q_6,c)=q_7$, $\delta(q_1,a)=q_2$,
$\delta(q_7,c)=q_8$, $\delta(q_2,b)=q_3$, $\delta(q_8,c)=q_6$
$\delta(q_3,b)=q_4$, $\delta(q_7,d)=q_9$, $\delta(q_4,b)=q_5$,
$\delta(q_9,d)=q_{10}$, $\delta(q_5,b)=q_6$, $\delta(q_{10},d)=q_9$

$$\delta(q_0,b)=\delta(q_0,c)=\delta(q_0,d)=\delta(q_1,a)=\delta(q_1,c)=\delta(q_1,d)=\delta(q_2,a)=\delta(q_2,c)=\delta(q_2,d)=\delta(q_3,a)=\delta(q_3,c)=\delta(q_3,d)=\delta(q_4,a)=\delta(q_4,c)=\delta(q_4,d)=\delta(q_5,a)=\delta(q_5,c)=\delta(q_5,d)=\delta(q_6,a)=\delta(q_6,b)=\delta(q_6,d)=\delta(q_7,a)=\delta(q_7,b)=\delta(q_8,a)=\delta(q_8,b)=\delta(q_8,d)=\delta(q_9,a)=\delta(q_9,b)=\delta(q_9,c)=\delta(q_{10},a)=\delta(q_{10},b)=\delta(q_{10},c)=q_{pit,}$$

הסבר : מסלול חישוב פשוט על aabbbb, ואז מעגל באורך 3 עבור אותיות c (שארית 1 במודולו 3) ומעגל באורך 2 עבור אותיות d (שארית 0 במודולו 2). נצטרך להקצות את q_{11} ולא לחזור אל q_7 כי ממנו ניתן להמשיך לראות c-ים ואילו לפי השפה, ברגע שהתחלנו לראות d לא ניתן לראות c מחדש.

פתרון ע"י ביטוי רגולרי: $L=a^2b^4c((c^3)^*)((d^2)^*)$

$L=\{a^qb^p|p,q\ are\ prime, and\ p+q<1000\}$

הוכחה שהנה רגולרית:

השפה היא כמות ראשונים של a-ים , ולאחריה כמות ראשונית של b -ים , כך שסכום הכמויות אינו מעל 1000. לכן, השפה סופית, ולכן רגולרית. נבנה ביטוי רגולרי המכריע את השפה:

$$L=\ a^2b^2+a^2b^3+a^2b^5+\ ...\ +a^2b^{997}+a^3a^2+a^3b^3+\ ...\ +\ a^3a^{991}+\ ...\ +a^{991}b^2+a^{991}b^3+a^{991}b^5+a^{991}b^7$$

$L=\{a^nb^m|n\ divides\ m, and\ m\ divides\ some\ 0\le i\le 100\}$

כדי ש n יחלק את m , נדרוש $n\le m$. התנאי $m\ divides\ some\ 0\le i\le 100$ אומר ש $m\le 100$, ולכן גם $n\le m$. קיבלנו שפה סופית ולכן רגולרית . נבנה לה ביטוי רגולרי :

$$L=a(b)^{1...100}+a^2b^{2i(1\le i\le 50)}+a^3b^{3i(1\le i\le 33)}+\ ...\ +a^{50}b^{50}+a^{50}b^{100}+a^ib^{i(51\le i\le 100)}$$

היות ובנינו אוטומט רגולרי, השפה רגולרית.

$L=\{(01)^n0(10)^n|n\in N\}$

נבנה אס"ד שיכריע את השפה:

$$A=(\{q_0,q_1,q_2,q_3,q_4,q_5,q_{pit}\},\sum=\{0,1\},q_0,\delta,F)$$

$\delta(q_0,0)=q_1,\delta(q_4,1)=q_{pit},\delta(q_2,0)=q_3$
$\delta(q_2,1)=q_{pit},\delta(q_5,0)=q_{pit},\delta(q_0,1)=q_{pit}$
$\delta(q_3,0)=q_{pit},\delta(q_5,1)=q_2,\delta(q_1,0)=q_{pit}$
$\delta(q_3,1)=q_4,\delta(q_{pit},0)=\delta(q_{pit},1)=q_{pit}$
$\delta(q_1,1)=q_2,\delta(q_4,0)=q_5$
$F: q_1\in F,\quad q_5\in F$

הסבר: כאשר נסיים רצף של 01010 ונראה עוד 1, הרי שאנחנו מצפים לראות 010 אחרי ולכן נחזור ל q_2. המצב q_1 הוא מצב מקבל שכן המילה '0' שייכת לשפה. בכל נקודה, אם נקבל את אות הקלט הלא נכונה, נלך למצב בור.

$L=\{a^nb^mc^p|\exists g\in N\ so\ that\ p=g^2\ and\ p\le 100, and\ n=m(mod\ p)\}$

במילים, זוהי שפת כל המילים מהצורה לעיל שבה n-m מתחלק ב p וגם p הוא ריבוע שלם שאינו גדול מ100. המשתנה p יכול לקבל את כל הערכים הריבועיים עד 100, כלומר: P={1,4,9,16,25,36,49,64,81,100}. המשתנים n,m שקולים מודולו p, ולכן ניתן להכריע אותם.

משום כך, נוכל לבנות ביטוי רגולרי עבור כל p כזה:

P=1: L= (a)*(b)*c

P=4: $L=((a)^4)^*((b)^4)^*c^4+((a)^4)^*a((b)^4)^*bc^4+((a)^4)^*a^2((b)^4)^*b^2c^4+((a)^4)^*a^3((b)^4)^*b^3c^4$

…

וכן הלאה. לכל p נוכל לבנות ביטוי רגולרי. לכן, הביטוי הרגולרי שיכריע את השפה יהיה איחוד של כולם, כלומר, לשים + בין כל הביטויים שנקבל. היות ובנינו ביטוי רגולרי עבור השפה – L הינה שפה רגולרית.

$-L=\{a^nb^m|\ n<m, and\ n+m<500\}$

חיתוך של אינו רגולרי עם שפה סופית ולכן נשאר שפה סופית ולכן השפה רגולרית.

השפה היא כמות של a-ים, ואחריה כמות גדולה יותר של b-ים, אך שתיהן ביחד לא מעל 500. לכן, השפה סופית, ולכן נוכל לבנות לה אס"ד (ענק) או ביטוי רגולרי (גם ארוך) שיכריע אותה:

$$L=ab^2+ab^3+\ldots ab^{498}+a^2b^3+a^2b^4+\ldots a^2b^{497}+\ldots+a^{248}b^{249}+a^{248}b^{250}+a^{248}b^{251}+a^{249}b^{250}$$

בנינו ביטוי רגולרי עבור השפה, ולכן היא רגולרית.

$L=\{a^n(a+b)^m|\ m\ge 0, n\ is\ a\ prime\ number\}$

נבנה ביטוי רגולרי שמכריע אותה: $L=a^2(a+b)^*$

הסבר: נרצה לראות כמות ראשונית של a – ים בהתחלה, ואז לראות איזשהו המשך של a -ים ו b – ים, שכן זוהי השפה. האופציה לקחת $(a+b)^m$ כאשר m לא מוגבל, "נוטלת את העוקץ" של הראשוניות, והופכת את השפה לרגולרית.

הוכחה ששפות רגולריות לא סגורות לאיחוד אינסופי:

שרשור של שני שפות רגולריות שנותן שפה שאינה רגולרית $L_1=\{a^n|n\in\mathbb{N}\}$ עם השפה $L_2=\{b^n|n\in\mathbb{N}\}$ נקבל את השפה $L=\{a^nb^n|n\in\mathbb{N}\}$

השפה $L=\{a^nb^m|n<m\ and\ n+m<500\}$. רגולרית. נבנה ביטוי רגולרי המוכיח את השפה:

$$r=ab^2+ab^3+\cdots.ab^{498}+a^2b^3+a^2b^4\ ....+a^2b^{497}\ ...+a^{248}b^{249}+..a^{248}b^{251}+a^{249}b^{250}+\cdots+a^{249}b^{250}$$

הוכח או הפרך: אם L רגולרית גם $L_{broken-subs}$ רגולרית

$L_{broken-sub}=\{x=x_1x_2\dots x_n\in\Sigma\ |n>0,\forall i:x_i\in\Sigma,\exists u_1u_2..,u_{n+1}\in\Sigma^*,u_1x_1u_2x_2\dots u_nx_nu_{n+1}\}$

תשובה: נוכיח את הטענה. השפה L רגולרית, ולכן קיים לה אוטומט NFA – ε המכריע אותה, עם מצב מקבל יחיד. (הוכח בהרצאות שמודל זה שקול לאס"ד רגיל) נסתכל על מסלול החישוב מq_0 ל q_f בין אותיות השפה $L_{broken-subs}$ ניתן להכניס מילים שלמות, ולכן נוכל לקחת כל תת קבוצה של מצבים במסלול החישוב בתור אוטומט המכריע את $L_{broken-subs}$ היות וקיים לה NFA – ε השפה רגולרית. נהפוך את המצבים שלקחנו למצבים מקבלים עבור $L_{broken-subs}$.

תהי $I\subseteq\mathbb{N}$ קבוצת אינדקסים. עבור שפה L נגדיר $L^I=\cup_{i\in I}L^i$.

א. תהי L שפה רגולרית ותהי I קבוצת האינדקסים כך ש $I'=N\backslash I$ שהיא בגודל 2 . נרשום $I'=\{i_1,i_2\}$ בטא את L^I במונחי L I' והוכח שהיא רגולרית.

ב. הוכח או הפרך: יהיו $I_1\subseteq I_2\subseteq\mathbb{N}$ ואזי לכל שפה מתקיים שאם L^{I_2} רגולרית אזי גם L^{I_1} רגולרית.

תשובה: א. אינטואציה: עבור $I=N$ אזי מתקיים ש$L^*=L^I$ ולכן יכול להיות שהשפה רגולרית. פתרון: תהי I כך שמתקיים N/I קבוצה סופית. ולכן נתאר את השפה כך $L^I=L^*/(\cup_{i\in N\backslash I}\ L^i)$. L^* - היא שפה רגולרית(סגירות לשרשור של שפות רגולריות). ומכיוון שN\I היא קבוצה סופית אזי נקבל איחוד של שפות רגולריות ולכן מסגירות לאיחוד סופי של שפוץ רגולריות השפה סופית. ולכן נקבל שיש כאן שני שפות רגולריות עם פעולת חיתוך ביניהם ולכן השפה רגולרית.

ב. נפריך את הטענה על ידי דוגמא נגדית : תהי $L=\{a^i|i\in\mathbb{N}\}$ תהי הקבוצה $I_2=\{2i+1|i\in\mathbb{N}\}$ (כלומר כל המספרים האי זוגיים) ו $I_1=\{all\ primes\ except\ 2\}$ באופן טרוויאלי מתקיים: $I_1\subseteq I_2\subseteq\mathbb{N}$ נראה כלי למרות שL^{i_2}רגולרית אזי L^{i_1} אינה רגולרית (ההוכחה עבור השפה a^P.)

ב. בנה ביטוי רגולרי עבור השפה:

L={x∈{a,b,c}* | every substring g of x of the form g=auc or g=cua contains at least two b's}

במילים, זוהי שפה כל המילים, כך שבין כל זוג a ו c במחרוזת (בסדר כלשהו), b חייבת להופיע פעמיים לפחות.

$$r=[(b+(c+bc)^*bb)+a(a+b(a+b^+a))^*b^+c(c+bc)^*bb]^*+(\varepsilon+c(b+c)^*(b+\varepsilon)+a(a+b(a+b^*a))^*bb^+c(c+bc)^*(b+\varepsilon)$$

הוכח או הפרך:

לכל ביטוי רגולרי r,s מתקיים $(r+s)^*=(r^*s)^*+(rs^*)^*$. נפריך את השקילות: $sr\in(r+s)^*$ אך $sr\notin(r^*s)^*+(rs^*)^*$ כי כדי להתחיל ב s נלך על החלק של $(r^*s)^*$ כאשר באיטרציה הראשונה לא ניקח אף r. אך במצב זה, הביטוי הרגולרי תמיד מסתיים ב-s ולכן $sr\notin(r^*s)^*+(rs^*)^*$.

הוכח או הפרך: $(rs)^*r=r(sr)^*$

הוכחה: שני הביטויים דומים. נוכיח כי $L[(rs)^*r]\subseteq L[r(sr)^*]$ באינדוקציה על $|w|$.

בסיס: $w=r$ כאשר $(|w|=1)$. לפי ההגדרה $r\in L[r(sr)^*]$

נניח כי הטענה נכונה לכל wr כאשר $|w|=n$ ונוכיח עבור xr כאשר $|x|=n+2$ (האיטרציה מוסיפה שני איברים בכל פעם).

נרשום $xr=wrsr$ לפי ההנחה $w=r(sr)^k$ $(k\ge 0)$ לכן $xr=wrsr=r(sr)^ksr=r(sr)^{k+1}$ ואכן $r(sr)^{k+1}\in L[r(sr)^*]$.

כיוון שני זהה לכיוון ראשון.

נוכיח שהשפה $L=\{a^{2p}|p\ is\ odd\}$ הינה שפה רגולרית.

נבנה אוטומט לשפה:

$A=(\Sigma_A,Q_A=\{q_0,q_1,q_2,q_3\},q_0,F_A=\{q_2\},\delta_A)$ כך ש

$\delta(q_0,a)=q_1$ $\delta(q_1,a)=q_2$ $\delta(q_2,a)=q_3$

$\delta(q_3,a)=q_0$ מספר המצב מהווה את השארית של חלוקת אורך הקלט ב-4, ולכן המצב q_2 הוא המצב המקבל.

נוכיח שהשפה $L=\{a^nb^mc^p|n+m+p=100\}$ הינה שפה רגולרית.

השפה סופית, ולכן רגולרית.

נבחר לראות כמות של a-ים (בין אפס ל-100), אח"כ כמות של b-ים (בין אפס ל 100-a), ואח"כ הכמות של ה-c-ים תהיה קבועה: $L=\sum_{i=0}^{100}\left(a^i(\sum_{j=0}^{100-i}(b^jc^{100-i-j}))\right)$

לחילופין, ניתן לבנות אוטומט לשפות הבאות: $L_1=\{abc^{98}\},L_2=\{a^2bc^{97}\},L_3=\{ac^{99}\},\dots$

ולכחל שאר האופציות האפשריות, ואז לבנות אוטומט איחוד בין כל השפות הרגולריות האלו. היות ואיחוד הוא תכונת סגור על כמות סופית של שפות, הרי שנקבל שפה רגולרית.

נוכיח שהשפה L המוגדרת להיות שפת כל המילים $w\in\sum^*$ כך שקיימת מילה $w'\in\sum^*$ עם $|w'|=2$ המופיעה לפחות פעמיים ב-w (ללא חפיפה). הערה: לכל w יכולה להיות 'w אחרת.

נבנה ביטוי רגולרי לשפה:

$$L=(a+b)^*aa(a+b)^*aa(a+b)^*+(a+b)^*bb(a+b)^*bb(a+b)^*$$

$\mathbf{L} = \{xy | x,y \in \Sigma^* \; and \; |x| = |y|\}$ $(a+b)^*ba(a+b)^*$
$+(a+b)^*ab(a+b)^*ab(a+b)^*$
מכיוון שהצלחנו לבנות ביטוי רגולרי לשפה השפה היא רגולרית.

<u>נוכיח שהשפה $\mathbf{L} = \{0^n1^m | (n-m) mod 5 = 0\}$ *רגולרית.*</u>

השפה היא כמות האפסים פחות כמות האחדות שקולה לאפס מודולו 5.

פתרון א': נוכל לבנות אוטומט המכריע את השפה. תחילה מעגל של חמישה מצבים עבור השקילות של כמות מופעי 0 במילה, ואז חמישה מעגלית שונים עבור השקילות של מופעי 1 במילה, כל אחד יוצא ממעגל אחר במעגל של אפס. זה יוצא אוטומט של 30 מצבים.

פתרון ב': נבנה ביטוי רגולרי לשפה:
$L = (0^5)^*(1^5)^* + (0^5)^*01(1^5)^* + (0^5)^*0^21^2(1^5)^* + (0^5)^*0^31^3(1^5)^* + (0^5)^*0^41^4(1^5)^*$

יהיו a ו- b שני ביטויים רגולריים מעל הא"ב $\{\Sigma, \cup, {}^*, {}^+, *, (,)\}$, כלומר ביטויים רגולרים מ- R_Σ. הוכיחו או הפריכו את הטענות הבאות: א. $L(a) = L(a^+)$ אזי מתקיים $a = \emptyset$
ב. אם $a = b^*$ אזי מתקיים: $L(b) \subseteq L(a)$.

תשובה: א. נפריך על ידי דוגמא נגדית: $a = \varepsilon$ ולכן מתקיים $L(a) = L(a^+)$ אך $a \neq \emptyset$.

ב. הטענה נכונה. באופן טריוויאלי, אם נשתמש באיטרציה בדיוק פעם אחת, נקבל *a=b* , ולכן כל מילה שנוכל לבנות מהביטוי הרגולרי *b* נוכל לבנות גם מהביטוי הרגולרי *a* . היות וניתן להשתמש באיטרציה כמה פעמים שנרצה, השפה של *a* גדולה יותר מהשפה של *b* ונוכל לרשום ש- $L(b) \subseteq L(a)$

נגדיר שפה יחידת מצב ויחידת מצב הפוכה: שפה *L* רגולרית הינה יחידת מצב אא"ם קיים אס"ד בעל מצב יחיד כך ש- $L = L(A)$. שפה רגולרית הינה יחידת מצב הפוכה הפוכה אא"ם קיים אס"ד בעל מצב יחיד שאינו מקבל כך ש- *L=L(A)*.

א. תנו דוגמא לשפה רגולרית שאיננה יחידת מצב . הסבירו את תשובתכם.

ב. תנו דוגמא לשפה רגולרית שאיננה יחידת מצב הפוכה. הסבירו את תשובתכם.

תשובה: א. כל שפה שחייבת לפחות שני מצבים מקבלים לא תהיה יחידת מצב. $L = \{w \in \{a\}^* | |w| = 1,2 mod 3\}$ (חייבת שני מצבים מקבלים) .

ב. כל שפה עם לפחות שני מצבים שאינם מקבלים איננה יחידת מצב הפוכה. דוגמא $L = \{w \in \{a\}^* | |w| = 1,2 \; mod \; 4\}$ השפה דורשת שני מצבים מקבלים ושני מצבים שאינם מקבלים.

הוכיחו או הפריכו:

א. **אם L היא שפת כל המילים w מעל {a,b} המקיימות: בכל רישא 'w של w מתקיים: $||w|_a - |w|_b| \leq 10$ כלומר מספר ה-a-ים ברישא פחות ממספר ה-b -ים ברישא קטן או שווה בערכו המוחלט מ10 , אז L רגולרית.**(תשובה: אוטומט עם 20 מצבים מ-10- עד 10)

ב. **אם L היא שפת כל המילים w המקיימות: $||w|_a - |w|_b| \leq 10$ כלומר מספר ה-a-ים במילה פחות ממספר הb-ים במילה קטן או שווה בערכו המוחלט מ10 , אז L שפה רגולרית.**

הטענה איננה נכונה. נגדיר את השפה הבאה: $L = \{w = (a^nb^n) \,|\}$ כלומר לא נרצה לראות את הביטוי ababab… סה"כ יותר מ11 פעמים. ההבדל בשאלה שכאן אין תנאי על הרישות, אלא על המילה הכללית. כאן נוכל להשתמש גם $L = \{a^nb^n\}$ המשך כמו סעיף א ..

L תקרא שפה קו – סופית אם המשלימה שלה היא שפה סופית.הוכיחו או הפריכו:

1. תהי L שפה קו – סופית כלשהי, ותהי P שפה לא רגולרית כלשהי. אז בהכרח $L \cup P$ רגולרית.

2. תהי L שפה קו – סופית כלשהי, ותהי P שפה לא רגולרית כלשהי. אז בהכרח $L \cap P$ רגולרית.

1. היות ו- L קו – סופית, הרי ש- 'L שפה סופית. לכן, ניעזר בדה- מורגן ונקבל (הסימן ' משמעו משלים):
$L \cup P = (L' \cap P')'$
היות ו- 'L סופי, החיתוך שלה עם שפה לא רגולרית נשאר סופי, ולכן רגולרי. לכן, מסגירות למשלים גם $L \cup P$ יהיה רגולרי.

2. עפ"י הדיאגרמה נקבל: $P' = (L' \cup P') \setminus (L' \setminus P')$
היות ו- L'\P' סופי, הוא רגולרי. חיסור הוא תכונת סגור ולכן שומר על רגולריות. אך 'P אינו רגולרי ולכן $(L' \cup P')$ לא רגולרי (אחרת 'P היה רגולרי). לפי דה- מורגן מתקיים: $L' \cup P' = (L \cap P)'$ ולכן $L \cap P$ לא רגולרי, מסגירות למשלים.

לכל אחת מהטענות הבאות קבעו אם היא נכונה או שגויה ונמקו בקצרה.

1. לכל שפה לא רגולרית L קיימת סדרה אינסופית של שפות $L_0 \subseteq L_1 \subseteq L_2 \subseteq L_3 \subseteq \cdots$ כך שלכל - i ≠ j , $L_i \neq L_j$, כך שלכל שפה L_i בסדרה היא תת קבוצה של L.

2. נסמן $L = \{a^nb^n \,| n \in \mathbb{N}\}$. קיימות שפות רגולריות L_1, L_2 כך ש- L_1LL_2 רגולרית.

1. כדי ששפה תהיה לא רגולרית, היא חייבת להיות אינסופית, ועוד משהו שמונע את הרגולריות שלה.
נוכיח את נכונות הטענה: תהי שפה L לא רגולרית. היות והשפה אינסופית, קיימת סדרה אינסופית של אורכי מילים בשפה. נגדיר: באופן טריוויאלי לכל $j \in \mathbb{N}$ מתקיים $L_j \subseteq L_{j+1}$, שהרי כל המילים עד אורך j + 1 מכילות את כל המילים עד אורך j. ואכן, כל שפה בסדרה L_j היא תת קבוצה של L. (כל שפה L_j היא שפה רגולרית, שכן יש כמות סופית של מילים עד אורך j אך השפה L אינה רגולרית, שכן j יכול להיות אינסופי).
2. הטענה נכונה. ניתן דוגמא המקיימת את הטענה: תהי $L_1 = \Sigma^*$ ותהי $L_2 = a$. כעת נסתכל על השרשור: $\Sigma^* a^nb^na = \Sigma^*$ ולכן רגולרי. זה יעבוד גם אם ניקח את אחת השפות להיות ריקה ואז השרשור יהיה שפה ריקה ולכן רגולרי.

הוכיחו או הפריכו: תהינה L_1, L_2 שפות לא רגולריות כך ש- $L_1 \subseteq L_2$, ו- L שפה המקיימת $L_1 \subseteq L \subseteq L_2$. אז L בהכרח אינה רגולרית.

פתרונות:
נגדיר את L_1, L_2 בצורה כזו שנוכל להכיל אותך אחת בשנייה, אך עדיין יישארו לא רגולריות, ואת L כך שהיא תוכל בתוך L_2 אך עדיין תישאר רגולרית. נפריך את הטענה ע"י דוגמא נגדית:
תהי $L_2 = \{a^*b^*c^md^m \,| m \in \mathbb{N}$ ותהי $L_1 = \{a^nb^n \,| n \in \mathbb{N}\}$ ותהי $L = \{a^*b^*\}$.
כעת מתקיים $L_1 \subseteq L_2$ עבור m = 0, ומתקיים $L_1 \subseteq L$ (קל לראות), וגם מתקיים $L \subseteq L_2$, אך L_1, L_2 אינן רגולריות (כבר הוכח ש-a^nb^n אינה רגולרית, וההוכחה עבור c^md^m זהה לחלוטין).

תהי L ו $I \subseteq \mathbb{N}$ קבוצת אינדקסים עבור שפה *L*. נגדיר $L^I = U_{i \in I} L^i$ הוכיחו או הפריכו:

א. **תהי *L* שפה רגולרית, ותהי קבוצת האידנקסים *I* כך ש- $\mathbb{N} \setminus I$ קבוצה סופית – אזי L^2 רגולרית.**

ב. **יהיו $I_1 \subseteq I_2 \subseteq \mathbb{N}$ אזי לכל שפה *L* מתקיים שאם L^{I_2} רגולרית אזי גם L^{I_1} רגולרית.**

א. עבור $I = \mathbb{N}$ אזי מתקיים ש- $L^I = L^*$ ולכן יכול להיות מצב שהשפה רגולרית. אנחנו יודעים ש*I* היא קבוצה אינסופית של אינדקסים, ולכן מסגירות לחיבור\ שרשור של שפה עבודת רק על מספר סופי של אינדקסים.
פתרון: תהי *I* קבוצת האינדקסים כך שמתקיים ש- $\mathbb{N} \setminus I$ קבוצה סופית ולכן נוכל לתאר את השפה כך: $L^2 = L^* \setminus (\cup_{i \in I} L^i)$ מכיוון ש- $N \setminus I$ קבוצה סופית אזי יש כאן איחוד של מספר סופי של שפות רגולריות, ולכן מסגירות לאיחוד סופי השפה רגולרית.

ב. דוגמא נגדית: $I_2 = \{n \in \mathbb{N}\}$ ואילו $I_1 = \{a^p | p \; is \; prime\}$ ואז השפה של l_2 רגולרית ואילו השפה של l_1 אינה רגולרית.

תהי *L* שפה לא רגולרית, הוכיחו או הפריכו:

א. **$\Sigma^* * L$ היא בהכרח רגולרית.**

ב. **$\Sigma^* * L$ היא בהכרח לא רגולרית.**

נשתמש בשפה $L = \{a^nb^n\}$ ונשרשר אותה עם Σ^* ונקבל : $L * \Sigma^* = \Sigma^*$ אם יתקיים *n=0* אזי נקבל $L = \varepsilon$. אבל עבור n>1 אזי מתקבל שפה לא רגולרית.

$$\mathbf{L} = \{a^{2p} | p \; is \; prime \; number\}$$

השפה לא רגולרית, הניפוח הוא בחירת i=2p+1

הוכחה: קיים ביטוי רגולרי: *((a+b)(a+b))
דרישות השפה הן שגודל ה x וגודל ה y יהיו זהים. לכאורה זה לא רגולרי, אך x הוא כל המחרוזות האפשריות, y הוא כל המחרוזות האפשריות. שרשור שלהם יוצר גם הוא את כל המחרוזות האפשריות. כל מחרוזת זוגית ניתן לחלקה בדיוק באמצע ולהגדיר את החצי השמאלי כ x והימני כ y. יש להוסיף הוכחה שהשפה המתקבלת ע"י הביטוי הרגולרי אכן השפה הנדרשת בשאלה.

<u>תהי L שפה רגולרית. אז השפה: $extend(L) = \{v \in \Sigma^* \,| \exists u \in L, w \in \Sigma^* \; s.t. \; v = uw\}$ גם היא בהכרח רגולרית.</u> השפה רגולרית כי מדובר בשרשור של שפה רגולרית עם Σ^* (רגולרית בשל סגירות לשרשור)

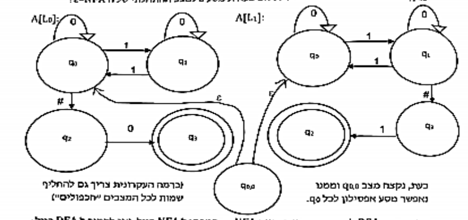
פרק ז': אוטומט סופי לא דטרמיניסטי עם מסעי ε (ε-NFA)

מודל נוסף של אוטומט לא דטרמיניסטי, החידוש הוא שעכשיו ניתן לבצע מעברים בין המצבים באוטומט ע"י קריאת המילה הריקה, כלומר, בלי לקרוא כלל, עבור חלק מהמצבים, בהם מוגדר "מסע ε".

דוגמה: בנו ε-NFA לשפה הבאה:

$L=\{\sigma_1\sigma_2...\sigma_n\#\sigma: i: \sigma_i \in \{0,1\}, \sigma = xor(\sigma_1\sigma_2...\sigma_n)\}$

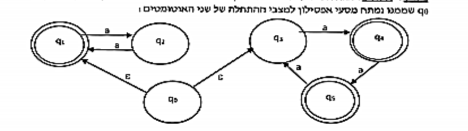
<u>פתרון</u>: נפצל את השפה לשתיים: $L_0=\{\sigma_1\sigma_2...\sigma_n\#0: i: \sigma_i \in \{0,1\}, 0 = xor(\sigma_1\sigma_2...\sigma_n)\}$, $L_1=\{\sigma_1\sigma_2...\sigma_n\#1: i: \sigma_i \in \{0,1\}, 1 = xor(\sigma_1\sigma_2...\sigma_n)\}$. כעת $L=L_0 \cup L_1$. נבנה אס"ד לכל אחד מהן, ונחבר את מצבי ההתחלה של שניהם בעזרת מסע ε למצב התחלתי של ה ε-NFA:



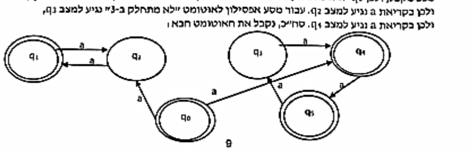
כעת, נקצה מצב $q_{0,0}$ וממנו נאפשר מסע אפסילון לכל q_0.
לכרמה העקרונית צריך גם להחליף שמות לכל המצבים "חכפוליים"

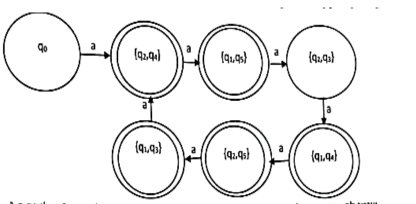
דוגמה: בנו DFA לשפה הבאה, ע"י בניית ε-NFA, המרתו ל NFA רגיל, ואז לתמיר ל DFA רגיל: $L=\{w \in \{a\}^* \,|\, |w|=0 \; (mod \; 2) \; or \; |w| \neq 0 \; (mod \; 3)\}$. במילים- כל המילים המורכבות רק מתאות a, כך שגודל המילה זוגי, או לא מתחלק ב-3.

<u>פתרון</u>: <u>ε-NFA</u>: נבנה אס"ד עבור כמות זוגית, ואס"ד עבור כמות שלא מתחלקת ב-3, ואז נקצה q_0 שממנו נמתח מסעי אפסילון למצבי ההתחלת של שני האוטומטים:



כעת, נמיר את האוטומט ל <u>NFA רגיל</u>: מה שמפריע לנו זה מסע האפסילון, ולכן "נעקוף" אותו ע"י בחינת השאלה "לאיפה מגיעים אחרי שקוראים a, בכל אחת מהאופציות של מסע האפסילון". q_1 מצב מקבל, ולכן q_0 יהיה מצב מקבל. עבור מסע אפסילון לאוטומט "מתחלק ב-2" נגיע למצב q_1, ולכן בקריאת a נגיע למצב q_2. עבור מסע אפסילון לאוטומט "לא מתחלק ב-3" נגיע למצב q_3, ולכן בקריאת a נגיע למצב q_4. סה"כ, נקבל את האוטומט הבא:





שימו לב: היה ניתן לפתור ישירות ע"י DFA בעזרת שימוש בעובדה שחלוקה ב-2 וחלוקה ב-3 מתפרש כחלוקה ב-6: $|w|=0,1,2,4,5 \; mod \; (6)$ ולבנות לזה אס"ד.

פרק ח': שקילות מודלים

המשמעות של שקילות היא שניתן להכריע אותן שפות בשני מודלים שונים.

בהרצאות ראינו את השקילות בין DFA ל NFA, ובין NFA ל ε-NFA. לכן כולם מכריעים את אותן שפות (רגולריות). שאלה לגיטימית היא לקבל שני מודלים, ולהחליט האם הם שקולים:

דוגמה: האם ε-NFA עם מצב מקבל יחיד, שקול ל DFA רגיל?

<u>פתרון</u>: כן. כדי להוכיח שקילות, יש להראות הכלה דו-כיוונית.

כיוון ראשון: בהינתן ε-NFA עם מצב מקבל יחיד, ניתן להמיר אותו ל NFA, ומשם להמיר ל DFA.

כיוון שני: בהינתן DFA, נבנה ε-NFA שקול:

נקצה מצב q_f, שהוא מצב מקבל יחיד, ומכל מצב שהיה מקבל ב DFA נמתח מסע אפסילון אל q_f. נבטל את הקבלה של המצבים המקוריים לאחר שנמתח מהם מסעי אפסילון, וקיבלנו את הנדרש.

דוגמה: נגדיר מודל 2-reg-NFA כך: בקריאת אות σ, ניתן לעבור בדיוק לשני מצבים. באופן פורמלי: $\forall \sigma \in \Sigma, \forall q \in Q, \exists i \neq j: : \delta(q,\sigma)=\{q_i,q_j\}$. האם המודל שקול ל NFA רגיל?

<u>פתרון</u>: נראה כי המודל שקול ל DFA רגיל, ואנחנו כבר יודעים שקיימת שקילות מ DFA ל NFA.

כיוון ראשון: בהינתן 2-reg-NFA, נבנה DFA שקול ע"י האלגוריתם שתואר בפרק ו'.

כיוון שני: בהינתן $DFA=(Q,\Sigma,q_0,\delta,F)$, נבנה 2-reg-NFA שקול:

מה שמפריע הוא שבאס"ד רגיל, עבור כל אות יש אופציה להתקדם למקום אחד בלבד. לכן, נקצה עוד מצב שאליו ניתן להגיע בקריאה של כל אות, וממנו נאפשר להגיע רק לעצמו. כעת, לכל המצבים יש בדיוק שני מצבים להגיע אליהם בכל קריאה, פרט למצב החדש הזה, ולכן נקצה עוד מצב כזה, שמהקודם מגיעים אליו ולעצמו, וממנו אל הקודם וחזרה. בצורה פורמלית:

$2\text{-reg-NFA}=(Q \cup \{q_{c1},q_{c2}\},\Sigma,q_0,\delta',F)$. נגדיר את δ':

$\delta'(q,\sigma) := \{\delta(q,\sigma),q_{c1}\}$ $\quad \forall \sigma \in \Sigma, \forall q \in Q$
$\delta'(q,\sigma) := \{q_{c1},q_{c2}\}$ $\quad \forall \sigma \in \Sigma, \forall q \in \{q_{c1},q_{c2}\}$

היות ולא השפענו על מסלול החישוב באף צורה (לא שינינו את F בכלל), המודלים שקולים.