תורת החישוביות – תרגול 4
רדוקציות חישוביות

מבוא

רדוקציה היא הכלי שבו אנו משתמשים כדי לתרגם בעיה חישובית אחת לבעיה חישובית אחרת.
פורמלית, רדוקציה של השפה L_1 לשפה L_2 היא פונקציה $f:\Sigma^*\to\Sigma^*$ שהיא:

1. מלאה.
2. ניתנת לחישוב.
3. תקפה: $x\in L_1 \iff f(x)\in L_2$.

שימו לב: f מוגדרת **לכל** x, ולא רק למילים ששייכות ל־L_1. כמו כן אין שום הכרח ש־f "תתפוס" את כל L_2 – ייתכן מאוד שתמונת f על L_1 תהיה קבוצה מצומצמת בהרבה מכל L_2.

אם קיימת רדוקציה של L_1 ל־L_2 אומרים ש־L_1 ניתנת לרדוקציה אל L_2 ומסמנים זאת $L_1\le L_2$.

חשיבותן של רדוקציות נובעת ממשפט הרדוקציה:

- אם $L_2\in \mathrm{R}$ ו־$L_1\le L_2$ אז גם $L_1\in\mathrm{R}$. בדומה, אם $L_2\in\mathrm{RE}$ ו־$L_1\le L_2$ אז גם $L_1\in\mathrm{RE}$.

הוכחת המשפט פשוטה: בהינתן מכונה M_2 עבור L_2 ניתן לבנות מכונה M_1 עבור L_1 שעל קלט x ראשית מחשבת את $f(x)$ (כאן נדרש ש־f תהיה מלאה אחרת שלב זה עשוי שלא להסתיים, וכמובן נדרש ש־f תהיה ניתנת לחישוב) ושנית מריצה את M_2 על $f(x)$ ועונה כמוה (כאן נדרש ש־f תהיה תקפה).

נציג ללא הוכחה מספר תכונות של רדוקציות (מומלץ כתרגיל למצוא את ההוכחה):

1. לכל שפה L מתקיים $L\le L$ (רדוקציה היא יחס רפלקסיבי).
2. אם $L_1\le L_2$ וגם $L_2\le L_3$ אז $L_1\le L_3$ (רדוקציה היא יחס טרנזיטיבי).
3. רדוקציה איננה יחס סימטרי: כך למשל $\mathrm{HP}\le\Sigma^*$ אך לא $\Sigma^*\le\mathrm{HP}$ (אחרת ממשפט הרדוקציה היינו מקבלים ש־$\mathrm{HP}\in\mathrm{R}$).
4. אם $L_1\le L_2$ אז $\overline{L_1}\le\overline{L_2}$.

השימוש העיקרי שלנו ברדוקציות הוא על מנת להראות ששפות מסוימות **אינן** ב־R או RE. לשם כך אנו הופכים את משפט הרדוקציה על פיו:

- אם $L_1\notin\mathrm{R}$ ו־$L_1\le L_2$ אז $L_2\notin\mathrm{R}$. בדומה, אם $L_1\notin\mathrm{RE}$ ו־$L_1\le L_2$ אז $L_2\notin\mathrm{RE}$.

תרגילים

תרגיל 1

יש להוכיח כי השפה $L_\varepsilon = \{\langle M\rangle \mid \varepsilon \in L(M)\}$ אינה ב־R.

פתרון: נוכיח כי $\mathrm{HP} \le L_\varepsilon$. ראינו כי $\mathrm{HP} \notin \mathrm{R}$, ובעזרת משפט הרדוקציה נסיק כי $L_\epsilon \notin \mathrm{R}$, כנדרש.

הרדוקציה תוגדר בתור $f(\langle M\rangle, x) = \langle M_x\rangle$ כך ש־M_x היא מכונה שעל כל קלט w:

1. מריצה את M על x (**ומתעלמת** מהקלט w שלה).
2. אם M עצרה, מקבלת.

באופן זה גרמנו לכך שהתשובה לשאלה "מהי השפה ש־M_x מקבלת?" תהיה תלויה בתשובה לשאלה "האם M עוצרת על x?" בפרט:

$$L(M_x) = \begin{cases} \Sigma^* & (\langle M\rangle, x) \in \mathrm{HP} \\ \emptyset & (\langle M\rangle, x) \notin \mathrm{HP} \end{cases}$$

בבירור הרדוקציה שהגדרנו מלאה וניתנת לחישוב – ייצור M_x מתוך $(\langle M\rangle, x)$ הוא פעולת קומפילציה פשוטה. הנקודה המרכזית כאן היא שייצור M_x מתוך $(\langle M\rangle, x)$ רק דורש מניפולציה סינטקטית של המחרוזות $\langle M\rangle$ ו־x – איננו צריכים, למשל, להריץ את M על x או כל פעולה חישובית דומה שעשויה שלא להסתיים.

תקפות הרדוקציה פשוטה להוכחה. למעשה, כל שנחוץ לנו הוא ש־$\epsilon \in \Sigma^*$ ו־$\epsilon \notin \emptyset$:

$$\begin{gathered} (\langle M\rangle, x) \in \mathrm{HP} \\ \Downarrow \\ L(M_x) = \Sigma^* \\ \Downarrow \\ \varepsilon \in L(M_x) \\ \Downarrow \\ \langle M_x\rangle \in L_\varepsilon \\ \Downarrow \\ f(\langle M\rangle, x) \in L_\varepsilon \end{gathered}$$

ובכיוון השני:

$$\begin{gathered} (\langle M\rangle, x) \notin \mathrm{HP} \\ \Downarrow \\ L(M_x) = \emptyset \\ \Downarrow \\ \varepsilon \notin L(M_x) \\ \Downarrow \\ \langle M_x\rangle \notin L_\varepsilon \\ \Downarrow \\ f(\langle M\rangle, x) \notin L_\varepsilon \end{gathered}$$

תרגיל 2

נעבור כעת לרדוקציה דומה באופיה אך מחוכמת יותר, שלרוב משתמשים בה כדי להראות שמשהו אינו שייך אפילו ל־RE:
$L_{\infty}=\{\langle M\rangle \mid |L(M)|=\infty\}$.

ניתן להשתמש בדיוק באותה רדוקציה כמו בתרגיל 1 כדי להוכיח ש־$L_{\infty}\notin \mathrm{R}$, אך אנו רוצים להוכיח תוצאה חזקה יותר – ש־$L_{\infty}\notin \mathrm{RE}$ – ולשם כך עלינו להראות רדוקציה משפה שאיננה ב־RE, כדוגמת $\overline{\mathrm{HP}}$.

נראה אם כן $\overline{\mathrm{HP}}\le L_{\infty}$. כאן עלינו לבנות מכונה כך שאם M **אינה** עוצרת על x, אז שפת המכונה אינסופית. לצורך כך המכונה שלנו כן תתייחס לקלט w שלה באופן מחוכם.

הרדוקציה תוגדר בתור $f(\langle M\rangle ,x)=\langle M_x\rangle$ כך ש־M_x היא מכונה שעל כל קלט w:

1. מריצה את M על x למשך $|w|$ צעדים.
2. אם M עצרה על x – דוחה. אחרת – מקבלת.

האבחנה הבסיסית כאן היא שאם M עוצרת על x, היא עוצרת אחרי מספר קבוע k של צעדים שאינו תלוי ב־w, ולכן M_x תדחה כל קלט w המקיים $|w|\ge k$.

מכאן שאם $(\langle M\rangle ,x)\in\overline{\mathrm{HP}}$ אז $|L(M_x)|=|\Sigma^*|=\infty$ ואחרת $|L(M_x)|<\infty$, מה שמראה את תקפות הרדוקציה.

תרגיל 3

עד כה עסקנו בשפות מהצורה "כל המכונות M כך ש־$L(M)$ מקיימת תכונה מסוימת". כעת אנו רוצים לעבור לשפה מהצורה "כל המכונות M כך שהמכונה M מקיימת תכונה מסוימת", שהן לרוב מאתגרות יותר (ולעתים שפה שמוגדרת באמצעות תכונה שנראית מסובכת היא למעשה ב־R).

נגדיר $L=\{\langle M\rangle \mid$ המכונה M בריצתה על ε מבצעת שלושה צעדים רצופים ימינה$\}$

לא קשה להראות כי $L\in \mathrm{RE}$. נראה כי $L\notin \mathrm{R}$ באמצעות רדוקציה $L_{\varepsilon}\le L$.

הרדוקציה תוגדר כך $f(\langle M\rangle)=\langle M'\rangle$ כך ש־M' זהה ל־M פרט לשינויים הבאים:

1. לכל $q\in Q$ נוסיף מצב q^R ומעברים $\delta\left(q^R,\sigma\right)=(q,\sigma,R)$ לכל $\sigma\in\Gamma$.
2. לכל מעבר $\delta(q,\sigma)=(p,\tau,R)$, נחליף אותו במעבר $\delta(q,\sigma)=\left(p^R,\tau,S\right)$.
3. q_{acc} יוצא מקבוצת המצבים הסופיים, ונגדיר $\delta(q_{acc},\sigma)=(q_{acc},\sigma,R)$.

פשר השינויים הללו: כל עוד M' מריצה את M היא אינה מסוגלת לבצע שלושה צעדים רצופים ימינה (שכן לפני כל צעד ימינה היא נעמדת לרגע במקום). אם ריצת M הסתיימה במצב מקבל, אז M' הולכת אינסוף צעדים ימינה, ובפרט שלושה.

אם $\langle M\rangle\in L_{\varepsilon}$ אז M בריצתה על ε מקבלת, ולכן M' בריצתה על ε מבצעת שלושה צעדים רצופים ימינה, ואפילו אינסוף (שימו לב ש־M' **אינה עוצרת** ואין בעיה עם זה).

אם $\langle M\rangle\notin L_{\varepsilon}$ אז M' אינה מבצעת שלושה צעדים רצופים ימינה, שכן במהלך ריצת M היא ודאי שאינה עושה זאת, ואחרי ש־M עוצרת כך גם M' (כי q_{rej} נותר מצב סופי).