עידן דויטש
u.multinet.co.il

מודלים חישוביים – פתרון תרגיל 3

.1

a. כתוב מכונת טיורינג שמחשבת את הפונקציה f(w)=101w. תן תיאור פורמאלי מלא של המכונה.

פתרון:

נגדיר מכונת טיורינג $M=\{Q,\Sigma,\Gamma,q_0,q_a,q_r,\delta\}$ כך ש:

$Q=\{q_{start}, q_{shift0}, q_{shift1}, q_{shift2}, q_{shift3}, q_{goleft}, q_{copy0}, q_{copy1}, q_{paste0}, q_{paste1}, q_{lastC0}, q_{lastC1}, q_{lastP0}, q_{lastP1}, q_{stamp1}, q_{stamp0}, q_{finish}\}$

$\Sigma=\{0,1\}$, $\Gamma=\{0,1,\bar{0},\bar{1},\neg,\bar{\neg}\}$ ($\neg$ הוא סימון לרווח), $q_0=\{q_{start}\}$, $q_a=\{q_{finish}\}$

ו- $q_r=\varnothing$. את δ נגדיר בטבלה:

$Q\setminus\Gamma$	0	1	$\neg$	$\bar{0}$	$\bar{1}$	$\bar{\neg}$
q_{start}	$q_{shift0},\bar{0},R$	$q_{shift0},\bar{1},R$	לא רלוונטי	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{shift0}	$q_{shift0},0,R$	$q_{shift0},1,R$	$q_{shift1},\neg,R$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{shift1}	$q_{shift2},\neg,R$	$q_{shift2},\neg,R$	$q_{shift2},\neg,R$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{shift2}	$q_{shift3},\neg,R$	$q_{shift3},\neg,R$	$q_{shift3},\neg,R$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{shift3}	$q_{goleft},\neg,L$	$q_{goleft},\neg,L$	$q_{goleft},\neg,L$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{goleft}	$q_{copy0},\neg,R$	$q_{copy1},\neg,R$	$q_{goleft},\neg,L$	$q_{lastC0},\mathbf{1},R$	$q_{lastC1},1,R$	לא רלוונטי
q_{copy0}	$q_{paste0},0,L$	$q_{paste0},1,L$	$q_{copy0},\neg,R$	לא רלוונטי	לא רלוונטי	$q_{goleft},0,L$
q_{copy1}	$q_{paste1},0,L$	$q_{paste1},1,L$	$q_{copy1},\neg,R$	לא רלוונטי	לא רלוונטי	$q_{goleft},1,L$
q_{paste0}	לא רלוונטי	לא רלוונטי	$q_{goleft},0,L$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{paste1}	לא רלוונטי	לא רלוונטי	$q_{goleft},1,L$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{lastC0}	$q_{lastP0},0,L$	$q_{lastP0},1,L$	$q_{lastC0},\neg,R$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{lastC1}	$q_{lastP1},0,L$	$q_{lastP1},1,L$	$q_{lastC1},\neg,R$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{lastP0}	לא רלוונטי	לא רלוונטי	$q_{stapF1},0,L$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{lastP1}	לא רלוונטי	לא רלוונטי	$q_{stapF1},1,L$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{stap1}	לא רלוונטי	לא רלוונטי	$q_{stap0},1,L$	לא רלוונטי	לא רלוונטי	לא רלוונטי
q_{stap0}	לא רלוונטי	לא רלוונטי	$q_{finish},0,L$	לא רלוונטי	לא רלוונטי	לא רלוונטי

תיאור מילולי של הטבלה:

א. נוסיף לתו הראשון קו ונרוץ לסוף המילה w (כלומר נלך ימינה עד הרווח הראשון).
ב. נלך 3 צעדים נוספים ימינה ונרשום בהם סימן רווח. במקום האחרון נסמן רווח עם קו (או פשוט סימן מינוס).
ג. נחזור חזרה שמאלה עד אשר נראה מחרוזת שאינה רווח. אם המחרוזת היא אינה עם קו, נשמור את המחרוזת בstate שלנו ונרשום במקומה רווח. נלך ימינה עד שנראה מחרוזת השונה מרווח. אם זו המחרוזת רווח עם קו, נחליף אותה ברווח, אחרת נשאיר את אותו הערך, נלך שמאלה ונחליף את הרווח במחרוזת ששמרנו בstate וחוזר חלילה.
ד. אם הגענו למחרוזת עם קו (אחרי שחזרנו שמאלה), נחליף אותו ב-1, נלך חזרה ימינה עד הרווח האחרון. במקומו, נשים את הערך ללא הקו.
ה. לאחר שמיקמנו את התו האחרון, נחליף את 2 הרווחים הנותרים ב1 ו-0, נחזור לראש הסרט ונעבור למצב סיום.

הוכחת נכונות:

<u>המכונה תעצור</u> – אורך הקלט הוא סופי, לכן הריצה ימינה לסוף המילה תסתיים (כאשר נגיע לרווח). בשלב הבא אנו מבצעים ריצות ימינה ושמאלה של לכל היותר 3 צעדים. בריצות אלו אנו למעשה מעתיקים ימינה אות אחר אות. פעולה זו תסתיים כיוון שאורך הקלט הוא סופי. בסיום העתקה ימינה, אנו מוסיפים 3 תווים בתחילת הקלט ולאחר מכן מסיימים את הריצה. כלומר, המכונה תעצור כנדרש.

<u>הערך המוחזר נכון</u> – כאמור, אנו מבצעים העתקה של אות לאחר אות 3 צעדים ימינה (תוך שמירה על הרווח שבסוף הקלט). כאשר אנו מגיעים לתו הראשון בקלט (לאחר שהסטנו את שאר התווים ימינה) אנו רושמים במקומו את האות 1 ומזיזים אותו ימינה כנדרש. לבסוף אנו רושמים ב-2 התאים שבין התו הראשון של הקלט לספרה 1 שרשמנו בראש הסרט את התוים 1 ו-0 וזזים שמאלה. כך בסיום ריצת המכונה, הראש נמצא על ראש הסרט, כל האותיות הוסטו 3 תאים ימינה, ובראש הסרט כתבנו 101 כנדרש.

b. כתוב מכונת מונים אשר מחשבת את הפונקציה $f(n)=\left\lfloor \frac{n}{2} \right\rfloor$. מותר להשתמש בכמה משתנים שרוצים. הצג תיאור מלא של המכונה.

פתרון:

נגדיר מכונת מונים עם 2 מונים. במונה הראשון נקבל את הקלט ובמונה השני נתחיל עם מונה ריק.

נגדיר את האלגוריתם של המכונה (לצורכי נוחות הקריאה כתבתי את הפקודות בעברית. עם זאת, את כולן ניתן לתרגם ישירות לאחת מהפקודות המורשות במכונת מונים):

שלב א:

i. נבדוק האם המונה הימיני שווה ל-0. אם כן, נעבור לשלב ב. אחרת נוריד 1 מהמונה הימיני.
ii. נבדוק שוב האם המונה הימיני שווה ל-0. אם כן, נעבור לשלב ב. אחרת נוריד 1 מהמונה הימיני ונוסיף 1 למונה השמאלי. לאחר עדכון המונים נחזור לשלב א.

שלב ב: נבדוק האם המונה השמאלי שווה ל-0. אם כן, נעבור לשלב ג. אחרת נוריד 1 מהמונה השמאלי, נוסיף אחד למונה הימיני ונחזור לשלב ב'.

שלב ג: נעצור. התוצאה במונה הימיני.

הוכחת נכונות:

<u>המכונה תעצור</u> – מספר החרוזים במונה הימיני (שאנו מקבלים כקלט) הוא סופי. לכן לאחר מספר מספיק של חיסורים, המונה בטוח יגיע ל-0 ושלב א יסתיים. כיוון שמספר הריצות של שלב א סופי (כיוון שהסתיים) גם מספר החרוזים שנוספו למונה השמאלי הוא סופי. לכן, לאחר מספר חיסורים מספיק, גם מונה זה יתאפס ושלב ב' יסתיים גם הוא. לבסוף נגיע לשלב ג' ונעצור.

<u>הערך המוחזר נכון</u> – בשלב א' על כל חיסור של 2 חרוזים מהמונה הימיני, אנו מוסיפים חרוז למונה השמאלי. אם הקלט הוא מספר זוגי, מספר החרוזים במונה השמאלי יהיה בדיוק מחצית מהקלט. אם הקלט הוא אי זוגי, אזי בלופ האחרון של שלב א, לא נבצע הוספה של אחד למונה השמאלי (בגלל הבדיקה ב-ii). לכן, מספר החרוזים במונה השמאלי יהיה ערך תחתון של חלוקת הקלט ב-2. בשלב ב' אנו מעבירים את החרוזים מהמונה השמאלי לימיני וסה"כ קיבלנו במונה הימיני ערך תחתון של חלוקה ב-2 של הקלט כנדרש.

.2

a. כתוב TM שמכריע את השפה $L_2 = \{l^k \mid k = 3^n, n \in N\}$ ($\Sigma = \{1\}$). אין צורך בתיאור פורמאלי, תיאור במילים יספיק.

פתרון:

נגדיר בשפת הסרט גם את התו $\cancel{1}$ ואת התו 1*.

א. נתחיל בתחילת הסרט ובמצב התחלה. אם התו הראשון הוא רווח, נחזיר T, אחרת נשנה את התו הראשון ל1*, נלך ימינה ונעבור למצב 1.

ב. במצב 1, כל עוד אנו נתקלים ב$\cancel{1}$, נלך ימינה ונשאר באותו מצב. אם נתקלנו ב-1, נהפוך אותו ל-$\cancel{1}$, נלך ימינה ונעבור למצב 2. אם נתקלנו ברווח נחזיר F.

ג. במצב 2, כל עוד אנו נתקלים ב$\cancel{1}$, נלך ימינה ונשאר באותו מצב. אם נתקלנו ב-1, נשאיר אותו (כלומר נרשום שוב 1), נלך ימינה ונעבור למצב 0. אם נתקלנו ברווח נחזיר F.

ד. במצב 0, כל עוד אנו נתקלים ב$\cancel{1}$, נלך ימינה ונשאר באותו מצב. אם נתקלנו ב-1, נהפוך אותו ל-$\cancel{1}$, נלך ימינה ונעבור למצב 1. אם נתקלנו ברווח, נפנה שמאלה ונעבור למצב left.

ה. במצב left נרוץ שמאלה עד שנגיע ל1*. נבדוק האם נשאר רק 1 יחיד בסרט שלא מחקנו (ע"י ריצה ימינה ושימוש במצב כאינדיקטור). אם כן, נחזיר T. אחרת, נרוץ חזרה ל-1* ונעבור למצב 0.

הוכחת נכונות:
<u>המכונה תעצור</u> – אורך הקלט הוא סופי. כל עוד קיימות אחדות בסרט, נבצע מחיקה באמצעות מצבים 1 ו-2. כיוון שאנו סורקים את הסרט בשיטתיות משמאל לימין, אם קיימות אחדות בוודאי שנתקל בהן בעת הסריקה ימינה. כמו כן, כאשר אנו מגיעים לסוף הקלט (מגיעים לרווח), אם צריך לבצע סריקה ימינה שוב, אנו מוודאים תחילה שאכן קיים על הסרט יותר מהופעה אחת של הספרה 1. אנו נמשיך במחיקות עד אשר יישארו פחות מ-3 אחדות על הסרט. במקרה ומספר האחדות אינו מתחלק ב-3 גם בסריקה האחרונה, נעצר באחד מהמצבים 1-2 ונחזיר F. אחרת, בעת סריקת הבדיקה (שלב ה) נתקל רק ב-1 יחיד ונחזיר T. נשים לב ששלב ה' מבטיח שלא נכנס ללולאה אינסופית כאשר לא נותרו יותר אחדות על הסרט.

<u>הערך המוחזר נכון</u> – הפעולות המבוצעות ע"י המכונה הן למעשה חלוקה ב-3 ובדיקת השארית. במידה והתנאי מתקיים, בכל פעם שנחלק ב-3 השארית תהיה 0 עד אשר נגיע לשלישיה האחרונה ובה נמחק 2 אחדות (ולכן נשאר עם הופעה יחידה של 1 על הסרט). במקרה זה המכונה תחזיר T כנדרש. אם בשלב מסוים השארית שונה מ-0, המכונה תהיה בשלב 1 או 2 ולכן כאשר תגיע לרווח תחזיר F כנדרש.

b. כתוב מכונת מונים המחשבת את הפונקציה $f(n)=n^2$. אין צורך בתיאור פורמאלי, תיאור במילים יספיק.

פתרון:

א. נגדיר מכונת מונים עם 4 חוטים: קלט, ספירה, זמני1 וזמני2. בשלב ההתחלה חוט הקלט יאתחל ל-n ושאר החוטים ל-0 (מצבי התחלה).

ב. תחילה, כל עוד חוט הקלט אינו 0, נוריד 1 מחוט הקלט ונוסיף 1 לחוט הספירה ולמונה הזמני1. כאשר חוט הקלט יגיע ל-0 נעבור לשלב הבא.

ג. בשלב זה כל עוד המונה הזמני1 אינו 0, נוריד 1 מהמונה הזמני1 ונוסיף 1 לחוט הקלט ולמונה הזמני2. כאשר המונה הזמני1 יגיע ל-0, נוריד 1 מחוט הספירה. אם חוט הספירה הגיע ל-0, נסיים את התוכנית. אחרת נעבור לשלב הבא.

ד. בשלב זה כל עוד המונה הזמני2 אינו 0, נוריד 1 מהמונה הזמני2 ונוסיף 1 לחוט הקלט ולמונה הזמני1. כאשר המונה הזמני2 יגיע ל-0, נוריד 1 מחוט הספירה. אם חוט הספירה הגיע ל-0, נסיים את התוכנית. אחרת נחזור אל סעיף ג'.

הוכחת נכונות:
<u>המכונה תעצור</u> – בשלב ההתחלתי, הקלט במכונה הוא סופי ולכן לאחר מספר מספק של החסרות, שלב זה יסתיים ונעבור לסעיף ג. בסעיף זה, מספר החרוזים במונה הוא סופי כיוון שהוא שיקוף של הקלט. לכן גם שלב זה יסתיים. בצורה דומה גם שלב ד' יסתיים. מספר החרוזים בחוט הספירה הוא גם מספר סופי (כשיקוף של הקלט) ולכן לאחר מספר ריצות של שלב ג' ושלב ד', מונה הספירה יגיע ל-0 והתוכנית כולה תסתיים.

<u>הערך המוחזר נכון</u> – בשלבים ג' וד' אנו למעשה מוסיפים למונה הקלט מספר חרוזים הזהה למספר החרוזים בקלט המקורי. אנו מריצים את שלבים ג' וד' סה"כ כגודל הקלט המקורי. כלומר, מוסיפים n פעמים לחוט הקלט n חרוזים. סה"כ נקבל בחוט הקלט n^2 חרוזים כנדרש.

.3

a. האם $L(TM') \subset RE$, $L(TM') \supset RE$ או שL(TM')=RE?

פתרון: נראה כי $L(TM') \subset RE$.

i. **טענה:** $L(TM') \subseteq RE$

הוכחה:
נראה כי כל מכונת 'TM (נסמן ב-'M) ניתן "לתרגם" למכונת TM (נסמן ב-M). מבחינת מספר המצבים וכמות סוגי התווים שניתן לרשום על הסרט, מכונות מסוג 'TM מוגבלות יותר ממכונות TM רגילות (כיוון שגודל קבוצות אלו אינו רק סופי אלא גם גודל הקבוצה מוגבל ל-2008). לכן, בעת הגדרת המכונה החדשה M, "נעתיק" את קבוצות אלו מהמכונה המקורית. בנוסף, נוכל גם להעתיק את כל מצבי ההתחלה והסיום.

נשים לב שבמכונות TM רגילות, ניתן בכל פעם לנוע צעד אחד ימינה או שמאלה על הסרט, כאשר במכונות 'TM ניתן לנוע עד 2008 צעדים בבת אחת. ניתן להתגבר על הבדל זה, ע"י בדיקת כל הפקודות במכונה המקורית 'M. פקודה בה מתבקשת המכונה לנוע k>1 צעדים ימינה או שמאלה, נתרגם ל-k פקודות אשר מלבד הפקודה המקורית יבצעו הזזה של הראש ימינה או שמאלה k צעדים (ע"י מעבר ממצב למצב). בפעולת תרגום הזו נוסיף מספר סופי של מצבים ופקודות ולכן הגבלות מכונת TM ישמרו.

כלומר, אנו יכולים לתרגם כל מכונה בTM' למכונה בTM. היות ו-L(TM)=RE נקבל כי $L(TM') \subseteq RE$.

ii. **טענה:** קיימת בעיה $P \notin R$ כך שלא ניתן להכריע אותה במכונת 'TM.

הוכחה:
נסתכל על הבעיה המקבלת מחרוזת x ומכריעה האם יש במחרוזת יותר מ-2100 תווים שונים. נשים לב תחילה שזו בעיה כריעה, שכן אורך כל קלט הוא סופי ולכן נוכל לעבור על הקלט ובאמצעות מספיק מצבים (הרבה מצבים אך בכל מקרה מספר סופי של מצבים) במכונת TM רגילה, לספור את כמות התווים השונים. במידה ויש מספיק תווים להחזיר T אחרת להחזיר F. לעומת זאת, במכונת 'TM לא נוכל להכריע את הבעיה, כיוון שמספר סוגי התווים שמותר להם להופיע על הקלט חייב להיות קטן ממש מ-2007. לכן לא נוכל אפילו להעביר קלט שעבורו צריך להחזיר T למכונה ולכן המכונה לא תוכל להכריע את השפה.

הראנו כי $L(TM') \subseteq RE$ אך גם כי $L(TM') \neq R$ ובפרט $L(TM') \neq RE$ ולכן $L(TM') \subset RE$.

b. האם $RE \subset L(TM'')$ או ש?L(TM")=RE

פתרון: נראה כי L(TM")=RE.

i. **טענה:** $L(TM'') \subseteq RE$

הוכחה:
נראה כי כל מכונת "TM (נסמן ב-"M) ניתן "לתרגם" למכונת TM (נסמן ב-M). כמו בסעיף א' "נעתיק" למכונה החדשה M את כל הקבוצות מ-'M (מצבים, תווים מורשים בקלט, תווים מורשים על הסרט וכו').

נשים לב שבמכונות TM רגילות, ניתן בכל פעם לנוע צעד אחד ימינה או שמאלה על הסרט, כאשר במכונות 'TM ניתן לנוע כמות בלתי מוגבלת של צעדים בבת אחת. ניתן להתגבר על הבדל זה, ע"י בדיקת כל הפקודות במכונה המקורית 'M. לאחר סריקה ראשונית, נוכל לבדוק מה מספר הצעדים המקסימאלי שהמכונה מבצעת בבת אחת. כל פקודה בה מתבקשת המכונה לנוע k>1 צעדים ימינה או שמאלה, נתרגם ל-k פקודות אשר מלבד הפקודה המקורית יבצעו הזזה של הראש ימינה או שמאלה k צעדים (ע"י מעבר ממצב למצב). בפעולת תרגום הזו נוסיף מספר סופי של מצבים ופקודות (כי מספר הפקודות ב"TM הוא סופי גם כן) ולכן הגבלות מכונת TM ישמרו.

כלומר, אנו יכולים לתרגם כל מכונה ב"TM למכונה בTM. היות ו-L(TM)=RE נקבל כי $L(TM'') \subseteq RE$.

ii. **טענה:** $L(TM'') \supseteq RE$

הוכחה:
מכונות מסוג TM הן למעשה מקרה פרטי של מכונות מסוג "TM. זאת, כיוון שניתן להסתכל על מכונות TM כמכונות "TM עם ההגבלה של צעד אחד בכל פעולה. לכן, ניתן לתרגם כל מכונה בTM למכונה ב"TM (פשוט ע"י ייבוא הגדרות המכונה המקורית למכונת "TM מבלי לעשות כל שינוי). לכן, היות ו-L(TM)=RE נקבל כי $L(TM'') \supseteq RE$.

הראנו כי $L(TM'') \subseteq RE$ וגם כי $L(TM'') \supseteq RE$ ולכן $L(TM'') = RE$.

4. לשפות הבאות קבע האם הן שייכות ל-R, ‏RE/R, ‏co-RE/R או אף אחת מהמחלקות. הוכח נכונות.

a. **קלט:** מכונת טיורינג M ומספר טבעי k.
שאלה: האם קיים קלט עבורו M עוצרת בפחות מ-k צעדים.

פתרון: שפה זו היא שפה כריעה **(שייכת ל-R)**

הוכחה:
נוכל להכריע את השפה ע"י כתיבת פרדיקט אשר עובר על כל הקלטים האפשריים עבור M ובודק עבור כל אחד, האם לאחר k צעדים הוא עוצר אחרי k צעדים. אם כן, מחזירים T, אחרת ממשיכים לקלט הבא. מספר הקלטים הוא סופי (הוכחת נכונות בהמשך), לכן, נוכל לבדוק את כולם. אם אף אחד מהקלטים אינו עוצר לאחר k צעדים נחזיר F.
הוכחת נכונות:
עצירה: במידה ו-M עוצרת באופן טבעי בפחות מ-k צעדים עבור x כלשהו, הרי כאשר נבדוק את ה-x הנ"ל, התוכנית בוודאות תעצור. עבור כל קלט שנבדוק, אם הוא ירוץ מעל ל-k צעדים, נפסיק את פעולתו באופן לא טבעי. נשים לב שמספר הקלטים הוא סופי, היות וכיוון שאנו בודקים האם M עוצרת תוך k צעדים, מכונת הטיורינג יכולה לכל היותר להגיע לתא ה-k בסרט. לכן, הערכים החל מהתא ה-k+1 אינם רלוונטים. היות ומספר האותיות על הסרט הוא סופי, ואורך הסרט הרלוונטי סופי, הרי שגם מספר הקלטים הרלוונטים ל-M הוא סופי. מכאן, שבמקרה ולא קיים x שעבורו M עוצרת ב-k צעדים, נוכל לעבור על כל הקלטים המייצגים הרלוונטים ואם את כולם נעצור באופן לא טבעי נוכל להחזיר F ובוודאות לעצור.

הערך המוחזר נכון: אם M עוצרת באופן טבעי בפחות מ-k צעדים, אנו מחזירים T ואכן קיים קלט שעבורו M מקיימת את הדרישה. אם לא קיים x כזה, הרי שעבור כל x שנבדוק נצטרך לעצור את M באופן לא טבעי ובסוף בדיקת כל הקלטים (יש כאמור מספר סופי) נחזיר F כנדרש.

b. **קלט:** מכונת טיורינג M ומחרוזת x.
שאלה: האם M כותב '1' על הסרט בזמן הריצה על x.

פתרון: שפה זו היא שפה הכריעה למחצה **(שייכת ל-RE/R)**.

i. **טענה:** השפה כריעה למחצה (שייכת ל-RE).

הוכחה:
נראה אלגוריתם המכריע את השפה למחצה. נריץ את M על x, אך בין צעד לצעד במכונת המונים, נבדוק האם בצעד האחרון נרשם על הסרט '1'. אם כן, נפסיק את ריצת M ונחזיר T. אחרת, נמשיך לצעד הבא. אם בשלב מסוים M נעצרה באופן טבעי נחזיר F.

הוכחת נכונות:
עצירה (במקרה ו-T): אם M רושמת בזמן הריצה על x את הספרה '1' על הסרט, הרי שלאחר הכתיבה על הסרט (ולפני הצעד הבא) נראה שאכן נרשם '1' ולכן נעצור את ריצת M ונחזיר T. כלומר, במקרה ואכן נרשם '1' על הסרט התוכנית תעצור.

הערך המוחזר נכון: כאמור, במידה ונרשם '1' על הסרט, התוכנית עוצרת ומחזירה T כנדרש. אם לעומת זאת M אינה רושמת '1' על הסרט, הרי שבמידה והיא עוצרת באופן טבעי (ובבדיקה האחרונה לא נרשם '1') אנו מחזירים F. במידה ו-M אינה עוצרת כלל גם הפרדיקט יתבדר כנדרש.

ii. **טענה:** השפה אינה כריעה (אינה שייכת ל-R).

הוכחה:
נניח בשלילה שקיים פרדיקט p המכריע את השפה ונראה רדוקציה מבעית העצירה.

עבור כל מכונת טיורינג N ומחרוזת x אשר עבורם נרצה לבדוק האם N עוצרת על הקלט x, נכתוב (באמצעות פונקציית מיפוי) מכונת טיורינג חדשה 'N. נעשה זאת ע"י העתקה של כל הגדרות המכונה מ-N אל 'N תוך הפעלת השינויים הבאים:

א. אם התו '1' קיים בין התווים החוקים האפשריים על הסרט (כקלט או לא) נגדיר תו חדש '*1' (או כל סימן אחר שאינו בקבוצות אלו) ונחליף את כל ההופעות של '1' בסימן החדש.

ב. את כל מצבי הסיום (המקבלים והלא מקבלים) נוציא מקבוצות מצבי הסיום אך בכולם נעדכן מעבר נוסף אל שלבי סיום (מקבלים או לא) חדשים אשר ידפיסו לפני סיום הריצה את הספרה '1'. נדאג שספרה זו תוגדר בתווים שיכולים להופיע על הסרט.

עתה, נעביר את 'N אל p ונעביר את התשובה שלו כתשובה לבעיית העצירה.

הוכחת נכונות:
עצירה: מספר המצבים ומספר התווים (הא"ב של המכונה) הוא מספרים סופיים. לכן, מספר המעברים האפשרי גם סופי ומכאן שפונקצית המיפוי (שרק עוברת על קבוצות סופיות ומבצעת שינויים בזמן סופי) היא פונקציה חשיבה שתעצור. אנו מעבירים את המכונה החדשה אל p. כיוון שאנו מניחים שp מכריע את השפה גם הוא בטוח יעצור.

הערך המוחזר נכון: אם המכונה N עוצרת, הרי שבשלב שהוספנו יודפס בוודאות על הסרט הספרה '1' ומכאן ש-p יחזיר בוודאות את הערך T. לעומת זאת, אם N אינה עוצרת, 'N המופעלת על x לא תדפיס אף פעם התו '1' על הסרט (כי לאחר השינויים שביצענו ב-'N, המצב היחיד בו מודפס '1' על הסרט הוא בשלב החדש שהוספנו לאחר שהחישוב של N על x מסתיים. כיוון שN אינה עוצרת, לא נגיע אף פעם לשלב זה ולכן אף פעם לא יודפס '1'). מכאן, היות ו-p חשיבה, כאשר נעביר את 'N ל-p יוחזר בוודאות F כנדרש.

מסקנה:
למעשה, הראנו רדוקציית מיפוי לבעיית העצירה. ראינו שבעיית העצירה אינה בעיה כריעה בתוכניות במודל חישובי כדוגמת baby. כיוון שכח החישוב של מודל זה, זהה לכח החישוב של מכונת טיורינג, בעיית העצירה אינה כריעה גם במודל טיורינג. כלומר קיבלנו סתירה ולכן השפה אינה כריעה.

הראנו כי השפה היא שפה הכריעה למחצה אך אינה שפה כריעה. כלומר, שפה זו שייכת ל-RE/R.

5. Accept = {<p,x>| p(x)=T} היא ב-RE/R. הוכח ש- $L \le_m Accept \Leftrightarrow L \in RE$.

הוכחה:

$\Leftarrow$ קיימת רדוקצית מיפוי $L \le_m Accept$. כלומר, קיימת פונקציה g(x) כך שאם נעביר את הערך המוחזר ממנה אל Paccept יוחזר T אמ"מ x שייך לשפה L. לכן, נוכל להגדיר את הפרדיקט המכריע את L להיות $p_L(x)$:=Paccept(g(x)). כלומר, הפרדיקט המכריע את השפה הוא רדוקציית המיפוי עצמה. כיוון ש-Accept שייכת ל-RE/R, הפרדיקט שהגדרנו מכריע ממש למחצה את L, כלומר L "לפחות" ניתן להכרעה למחצה ולכן שייך ל-RE.

הוכחת נכונות:

<u>עצירה</u> (כאשר הערך הוא T): $Paccept$ היא ב-RE/R, לכן תעצור כאשר הפרדיקט שמועבר אליה מקבל את x. הפונקציה g שמגדירה את רדוקצית המיפוי, מבטיחה שכאשר x שייך ל-L, הערך שיועבר ל- $Paccept$ יחזיר T ובפרט יעצור. לכן, עבור x השייך ל-L, הפרדיקט שהגדרנו בוודאות עוצר כנדרש משפה אשר נמצאת בRE.
<u>הערך המוחזר נכון</u>: אם x שייך ל-L אזי רדוקציית המיפוי מבטיחה שהערך שיוחזר מ- $Paccept$ יהיה T ולכן גם הפרדיקט שלנו יקבל T. אם x אינו שייך ל-L, רדוקציית המיפוי מבטיחה שהערך המוחזר מ- $Paccept$ לא יהיה T (יהיה F או יתבדר), לכן הערך שמוחזר ע"י הפרדיקט שהגדרנו, גם לא יהיה T כנדרש.

הראנו שקיים פרדיקט המכריע למחצה את L. כיוון שהפרדיקט הזה מוגדר ע"י $Paccept$ והיות וAccept שייכת ל-RE/R, ניתן גם את L באמצעות פרדיקט זה להכריע למחצה. כלומר, L שייכת ל-RE כנדרש.

$\Rightarrow$ $L \in RE$ לכן יש פרדיקט p_L המכריעה למחצה את השפה. נראה רדוקציית מיפוי: $L \le_m Accept$.

$$L(x) := Paccept(g(x)) \quad where \quad g(x) = return \quad < p_L, x >$$

הוכחת נכונות:

<u>עצירה</u> (כאשר הערך הוא T): $Paccept$ היא ב-RE/R, לכן תעצור כאשר הפרדיקט שמועבר אליה מקבל את x. ל $Paccept$ מועבר מ-g הזוג הסדור המכיל את הפרדיקט p_L המכריע למחצה את L ואת x. במקרה ש-x שייך ל-L אזי $p_L(x)$=T בוודאות. כלומר, p_L מקבל את x. לכן, Accept תעצור בוודאות.
<u>הערך המוחזר נכון</u>: אם x שייך ל-L אזי $p_L(x)$=T. ל $Paccept$ מועבר מ-g הזוג הסדור המכיל את הפרדיקט p_L המכריע למחצה את L ואת x. לכן, במקרה זה כיוון ש-p_L מקבלת את x, $Paccept$ יחזיר T. אם x אינו שייך ל-L אז $p_L(x)$=F או ש$p_L(x)$ מתבדרת. בכל מקרה, p_L אינה מקבלת את x ולכן $Paccept$ יחזיר F או יתבדר כנדרש.

6.

a. המחלקה RE סגורה על איחוד ועל חיתוך.

פתרון: זו טענה נכונה.

הוכחת החיתוך:

יהי $L_1, L_2 \in RE$ נראה כי $L_1 \cap L_2 \in RE$. כלומר עלינו להראות שקיים פרדיקט, כך ש- $x \in L_1 \cap L_2$ אמ"מ P(x)=T (ואחרת מוחזר F או ש-P מתבדר).
היות ו$L_1, L_2 \in RE$, קיימים פרדיקטים p1 ו-p2 המכריעים למחצה את השפות.
נגדיר את p המכריע למחצה את החיתוך להיות: $p(x) := p1(x) \wedge p2(x)$.

הוכחת נכונות:
אם $x \in L_1 \cap L_2$ אזי p1(x)=T וגם p2(x)=T. כלומר, שני הפרדיקטים עוצרים בוודאות ומחזירים T. לכן $p(x) := p1(x) \wedge p2(x) = T \wedge T = T$. כלומר הפרדיקט שהגדרנו <u>יעצור ויחזיר T כנדרש</u>.

אם $x \notin L_1 \cap L_2$ אזי x אינו שייך לפחות לאחת מהשפות. לכן, p1(x)=F (או יתבדר) או p2(x)=F (או יתבדר). כלומר הפרדיקט שהגדרנו <u>יעצור ויחזיר F או יתבדר כנדרש</u>.

הוכחת האיחוד:
יהי $L_1, L_2 \in RE$ נראה כי $L_1 \cup L_2 \in RE$. כלומר עלינו להראות שקיים פרדיקט, כך ש- $x \in L_1 \cup L_2$ אמ"מ P(x)=T (ואחרת מוחזר F או ש-P מתבדר).
היות ו $L_1, L_2 \in RE$, קיימים פרדיקטים p1 ו-p2 המכריעים למחצה את השפות.
נגדיר את p המכריע למחצה את איחוד: נריץ סטפר i צעדים לחלופין על p1 ו-p2 עם הערך x. כל עוד אף אחד מהם לא עצר באופן טבעי והחזיר T. אם אחד מהם יעצור באופן טבעי ויחזיר T, נחזיר T, אחרת נמשיך להגדיל את i ולרוץ עם סטפר על p1 ו-p2 לחלופין.

הוכחת נכונות:
אם $x \in L_1 \cup L_2$ אזי p1(x)=T או p2(x)=T. כלומר, לפחות אד מהפרדיקטים עוצר בוודאות ומחזיר T. לכן בשלב כלשהו של ריצת p, הסטפר ייעצר באופן טבעי על אחד מהפרדיקטים ויוחזר T ולכן גם p יחזיר T. כלומר הפרדיקט שהגדרנו <u>יעצור ויחזיר T כנדרש</u>.

אם $x \notin L_1 \cup L_2$ אזי x אינו שייך לשתי השפות. לכן, p1(x)=F (או יתבדר) או p2(x)=F (או יתבדר). במקרה כזה, או ששני הפרדיקטים ייעצרו באופן טבעי ויחזיר F או שלפחות אחד מהפרדיקטים יתבדר (ואם השני יעצור אז הוא יחזיר F). בכל מקרה, הפרדיקט שהגדרנו <u>יחזיר F או יתבדר כנדרש</u>.

b. המחלקה co-RE סגורה על איחוד ועל חיתוך.

פתרון: זו טענה נכונה.

הוכחה:
יהי $L_1, L_2 \in co-RE$ נראה כי $L_1 \cap L_2 \in co-RE$ ו- $L_1 \cup L_2 \in co-RE$.
כיוון ש- $L_1, L_2 \in co-RE$, מתקיים כי $\overline{L_1}, \overline{L_2} \in RE$. לכן, לפי מה שהוכחנו בסעיף a מתקיים $\overline{L_1} \cap \overline{L_2} \in RE$ וכן $\overline{L_1} \cup \overline{L_2} \in RE$.

מכאן, $\overline{\overline{L_1} \cap \overline{L_2}} \in co-RE$ וגם $\overline{\overline{L_1} \cup \overline{L_2}} \in co-RE$. ולכן, ע"פ חוקי תורת הקבוצות $L_1 \cup L_2 \in co-RE$ וגם $L_1 \cap L_2 \in co-RE$ כנדרש.

c. אם A לא כריעה וגם קיימת רדוקציית מיפוי מ-A למשלים של A אז $A \notin RE$ וגם $A \notin co-RE$.

פתרון: זו טענה נכונה.

הוכחה:
נניח בשלילה שהביטוי אינו נכון, כלומר $A \in RE$ או $A \in co-RE$.

אם $A \in RE$ אזי $A \in RE/R$ (כי A אינה כריעה) ומכאן ש-$\overline{A} \in co-RE/R$.
כמו כן, $A \notin co-RE/R$. כיוון שקיימת רדוקציית מיפוי מA לA משלים, נסיק כי גם $\overline{A} \notin co-RE/R$ אך זו סתירה.

אם $A \in co-RE$ אזי $A \in co-RE/R$ (כי A אינה כריעה) ומכאן ש-$\overline{A} \in RE/R$. כמו כן, $A \notin RE/R$. כיוון שקיימת רדוקציית מיפוי מA לA משלים, נסיק כי גם $\overline{A} \notin RE/R$ אך זו סתירה.

הנחנו בשלילה וקיבלנו סתירה ולכן הטענה המקורית נכונה.

7. יהי $L_1, L_2 \subseteq \{0,1\}^*$ שפות ב-RE כך ש-$L_1 \cup L_2 = \{0,1\}^*$ וגם $L_1 \cap L_2 \neq \emptyset$. הראה כי קיימת רדוקצית מיפוי $L_1 \leq_m L_1 \cap L_2$.

פתרון:
נראה פונקציה f הממפה מ-L_1 ל-$L_1 \cap L_2$. L_1, L_2 הן שפות הכריעות למחצה ולכן קיימים פרדיקטים P_{L1}, P_{L2} המכריעים את השפות למחצה. נמצא תחילה איבר מהחיתוך $L_1 \cap L_2$. נגדיר את הפרדיקט $P'(x,n) = S(P_{L1}, x, n) \wedge S(P_{L2}, x, n)$ (כלומר עבור כל x נרוץ n צעדים עם כל פרדיקט ואם שניהם עצרו באופן טבעי והחזירו T נחזיר גם T אחרת נחזיר F). נעבור בשיטת השבלול על כל האינדקסים והקלטים האפשריים. כיוון שקבוצת החיתוך אינה ריקה, בשלב כלשהו של הריצה, נגיע לאינדקס גדול מספיק ואיבר שנמצא בחיתוך ולכן שני הסטפרים יעצרו באופן טבעי והתוצאה תהיה T (כיוון ש-X שייך ל-2 הקבוצות ולכן שייך גם לחיתוך). כלומר, הריצה בוודאות <u>תעצור ותחזיר את הערך הנכון</u>. נסמן את האיבר שמצאנו כ-y.

נגדיר את הפונקציה f(x):
תחילה נבדוק האם x שייך לשפה $\{0,1\}^*$. זו בעיה כריעה. אם x אינו שייך לשפה זו, נחזיר F. אחרת נפעל כך: נגדיר משתנה אינדקס i שיאתחל ל-1. נרוץ בכל שלב i צעדים על כל אחד מהפרדיקטים עד שאחד משהם ייעצר באופן טבעי ויחזיר תשובה כלשהי. כאשר אחד מהפרדיקטים עצר, נפעל על פי המקרים הבאים:

1. אם $P_{L2}(x) = T$ (כלומר x שייך לקבוצה L_2) או $P_{L1}(x) = F$ (כלומר x אינו שייך לקבוצה L_1) – נחזיר את x.
2. אם $P_{L2}(x) = F$ (כלומר x אינו בקבוצה L_2 בוודאות) או $P_{L1}(x) = T$ (כלומר x בוודאות בקבוצה L_1) – נחזיר את y.

הוכחת נכונות:
<u>הפונקציה חשיבה ותעצור</u> – בדיקת שייכות לשפה $\{0,1\}^*$ היא בעיה כריעה ולכן תעצור. כיוון ש-$L_1 \cup L_2 = \{0,1\}^*$ x חייב להיות בוודאות באחת מהקבוצות לפחות. הקבוצות כריעות למחצה ולכן אם x שייך לקבוצה כלשהי, הפרדיקט שלה יהיה חייב לעצור ולהחזיר T. כלומר, לא ייתכן מצב ששני הפרדיקטים יתבדרו ולפחות אחד מהם יהיה חייב לעצור ולהחזיר T. ברגע שהוחזרה תוצאה מאחד הפרדיקטים (וזה כאמור בהכרח יקרה), f מחזירה את המחרוזת x או את המחרוזת y. לכן, f בהכרח חשיבה ותעצור כנדרש.

<u>הערך המוחזר נכון</u> – נסתכל על ארבעת המצבים האפשריים:

1. $P_{L2}(x) = T$ - במקרה זה x שייך ל-L_2. הפונקציה מחזירה את x ולכן, אם x שייך גם ל-L_1, הוא יהיה שייך גם לחיתוך והפרדיקט של החיתוך יחזיר T כנדרש. אם x אינו ב L_1 הוא אינו שייך לחיתוך ולכן הפרדיקט של החיתוך יחזיר F או יתבדר כנדרש.

2. $P_{L1}(x)=F$ - במקרה זה x אינו שייך ל-L_1. הפונקציה מחזירה את x אך כיוון שהוא אינו שייך לחיתוך בוודאות, הפרדיקט של החיתוך יחזיר F או יתבדר כנדרש.

3. $P_{L2}(x)=F$ - במקרה זה x אינו שייך ל-L_2 ולכן שייך בוודאות ל-L_1. הפונקציה מחזירה את y שע"פ הגדרתו שייך לחיתוך. לכן פרדיקט החיתוך יחזיר T כנדרש.

4. $P_{L1}(x)=T$ - במקרה זה x שייך בוודאות ל-L_1. הפונקציה מחזירה את y שע"פ הגדרתו שייך לחיתוך. לכן פרדיקט החיתוך יחזיר T כנדרש.

הראנו פונקציה f הממפה קלטים מהקבוצה L_1 אל קבוצת החיתוך. הוכחנו כי זו פונקציה חשיבה הממפה כראוי את האיברים. כלומר, אכן קיימת רדוקצית מיפוי בין השפות.