קווים לפתרון כמה שאלות בממ"ן 14 סמסטר 2013ב

שאלה 1

הרעיון: עוברים על כל תת-קבוצה $\{y_1, ..., y_m\}$ של $S = \{x_1, ..., x_k\}$,
מחשבים את הסכום $y_1 + \cdots + y_m$, ומשווים את הסכום הזה ל-t.
אם הסכום שווה ל-t, מקבלים. אם לכל תת-קבוצה, הסכום הזה שונה מ-t, דוחים.

כדי לעבור על כל התת-קבוצות של S, נשים לב שלכל תת-קבוצה כזו מתאימה מילה באורך k של 0-ים ו-1-ים. במילה הזו יש 1 במקום ה-i אם x_i שייך לתת-קבוצה, ויש 0 במקום ה-i אם x_i לא שייך לתת-קבוצה.
כדי לעבור על כל התת-קבוצות, עוברים בסדר לקסיקוגרפי על כל המילים באורך k מעל $\{0, 1\}$.

האלגוריתם: רושמים תחילה k אפסים. (זה מייצג את הקבוצה הריקה).
עוברים על S, ובוחרים מתוך S את האיברים שמתאימים ל-1-ים במילה באורך k מעל $\{0, 1\}$.
מחשבים את הסכום של התת-קבוצה שהתקבלה. אם הסכום שווה ל-t, מקבלים.
אם הסכום שונה מ-t, עוברים לתת-קבוצה הבאה: מוסיפים 1 (בבינארי) למילה באורך k, ובודקים את הסכום של התת-קבוצה החדשה.
ממשיכים כך עד שנמצאת תת-קבוצה שסכום איבריה שווה ל-t, או עד שמגיעים למילה באורך k שכולה 1-ים. (מילה זו מייצגת את S). אם הסכום של כל תת-קבוצה שונה מ-t, דוחים.

חישוב המקום הדרוש: המקום הדרוש למילה באורך k הוא לכל היותר ליניארי בגודל הקלט.
כך גם המקום הדרוש לסכום של התת-קבוצה. (המקום הדרוש לסכום של m מספרים איננו גדול מן המקום הדרוש ל-m המספרים).
האלגוריתם עושה שימוש חוזר במקום. לכן המקום הדרוש לו הוא $O(n)$.

שאלה 3

כל שפה ב-PSPACE אפשר להכריע **במקום פולינומיאלי**.
נבחר נוסחה ϕ_1 ששייכת ל-SAT (למשל, $x \vee y$) ונוסחה ϕ_2 שלא שייכת ל-SAT (למשל, $x \wedge \neg x$).

תהי A שפה ב-PSPACE. נתאר רדוקציה **במקום פולינומיאלי** של A ל-SAT:
"על קלט w:

1. בדוק האם w שייכת ל-A.
2. אם כן, החזר את ϕ_1; אם לא, החזר את ϕ_2."

הכרעת השייכות ל-A (בסעיף 1) מתבצעת **במקום פולינומיאלי**. לכן הרדוקציה כולה מתבצעת **במקום פולינומיאלי**.

SAT שייכת ל-PSPACE. לכן היא תהיה PSPACE-שלמה.

שאלה 4

$ADD \in L$: נתאר מכונה שמכריעה את ADD במקום לוגריתמי :

המכונה מחזיקה בסרט העבודה מונה c שבתחילה ערכו 1.

המכונה זוכרת בעזרת מצבים האם יש או אין נשא (carry), מהי הספרה של x במקום ה-c מימין, ומהי הספרה של y במקום ה-c מימין.

בתחילה המכונה נכנסת למצב שזוכר שאין נשא (carry).

המכונה נעה בסרט הקלט עד מימין ל-x. אז המכונה נעה לספרה ה-c מימין של x. (את זה אפשר לממש בעזרת העתקת המונה c למשתנה עזר d. מחסרים 1 מ-d אחרי כל תנועה שמאלה לספרה הבאה של x. כאשר ערכו של d מתאפס, נמצאים על הספרה המתאימה של x).

המכונה זוכרת במצב את הספרה ה-c מימין של x, וכן האם יש או אין נשא. (יש ארבעה מצבים כאלה).

באופן דומה, המכונה תנוע מימין ל-y, תגיע לספרה ה-c מימין של y, ותזכור במצב את הספרה ה-c מימין של y, את הספרה ה-c מימין של x, וכן האם יש או אין נשא. (יש שמונה מצבים כאלה - יש/אין נשא, הספרה ב-x היא 0/1, הספרה ב-y היא 0/1).

לכל מצב כזה מתאימה הספרה שצריכה להיות ב-z במקום ה-c מימין.

המכונה מגיעה לספרה הזו, ובודקת האם הספרה מתאימה למה שצריך להיות אם אכן $z = x+y$.

אם לא, המכונה נכנסת למצב הדוחה. אם כן, המכונה עוברת למצב שזוכר האם יש או אין נשא, מגדילה ב-1 את c, ועוברת לבדיקת הספרה הבאה.

ממשיכים כך עד שמסיימים לעבור על z (מימין לשמאל). אם כל הספרות של z נמצאו מתאימות ל-$x+y$, והמעבר על x ועל y הסתיים, מקבלים.

הערה : אם ערכו של c גדול מן האורך של x או של y, מתייחסים לספרה במספר המתאים כאל 0.

$PAL\text{-}ADD \in L$: נתאר מכונה שמכריעה את $PAL\text{-}ADD$ במקום לוגריתמי :

תחילה המכונה תחשב את האורך m של $x+y$ בייצוג בינרי. את זה אפשר לבצע במקום לוגריתמי בדומה למכונה של ADD. (מחשבים את הספרות של הסכום עד שמסיימים לעבור על x ועל y ועד שאין נשא).

מחשבים את החלק השלם של $m/2$. (מוחקים את הספרה הימנית ביותר של m).

כעת מבצעים בלולאה מ-1 עד החלק השלם של $m/2$:

מחשבים את הספרה ה-i של $x+y$ (כמו במכונה של ADD).

מחשבים את הספרה ה- $m-i+1$ של $x+y$ (כמו במכונה של ADD).

אם הספרות שונות, דוחים.

אם כל הספרות נמצאו שוות, מקבלים.

המקום הנדרש : m, החלק השלם של $m/2$, משתנה הלולאה i והמקום הדרוש לחישוב הספרה ה-i והספרה ה- $m-i+1$ של $x+y$.

בעזרת שימוש חוזר במקום, אפשר להסתפק במקום שהוא מסדר גודל לוגריתמי של גודל הקלט.

שאלה 6

$\overline{E_{\mathrm{DFA}}}$ היא שפת התיאורים של אוטומטים סופיים דטרמיניסטיים שמקבלים **לפחות מילה אחת**.

שייכות ל-NL:

"על קלט $<M>$ כאשר M הוא אוטומט סופי דטרמיניסטי:

1. ספור את המצבים של M לתוך מונה (בייצוג בינארי).
2. שמור על סרט העבודה את מספרו של המצב הנוכחי. בתחילה זהו המצב ההתחלתי.
3. בצע עד שהמצב הנוכחי יהיה מצב מקבל באוטומט, או עד שהמונה יתאפס:
   3.1 בחר באופן לא דטרמיניסטי סמל מן האלפבית של האוטומט M.
   3.2 מצא בפונקצית המעברים של M את המעבר המתאים למצב הנוכחי ולסמל שנבחר.
   3.3 מחק את המצב הנוכחי, וכתוב במקומו את המצב שאליו עוברים.
   3.4 חסר 1 מן המונה.
4. אם המצב הנוכחי הוא מצב מקבל, קבל. אחרת, דחה."

המכונה מכריעה באופן לא דטרמיניסטי את השפה, ומשתמשת במקום לוגריתמי בסרט העבודה.

רדוקציה של *PATH*:

"על קלט $<G, s, t>$ כאשר G הוא גרף מכוון, s ו-t הם צמתים ב-G:

1. ספור את הצמתים של G לתוך מונה (בייצוג בינארי). נסמן מספר זה על-ידי n.
2. הגדר אוטומט סופי דטרמיניסטי $<M>$:
   2.1 קבוצת המצבים של האוטומט תהיה קבוצת הצמתים של G + מצב אחד נוסף q.
   2.2 המצב ההתחלתי יהיה s.
   2.3 האלפבית של האוטומט יהיה בעל n סמלים.
   2.4 המצב t יהיה המצב המקבל היחיד. כל שאר המצבים הם מצבים לא מקבלים.
   2.5 פונקצית המעברים: אם מצומת v בגרף G יוצאות k קשתות ל-k צמתים, אז מהמצב המתאים ל-v באוטומט יצאו k קשתות למצבים המתאימים. כל קשת כזו תתויג באחד מ-k הסמלים הראשונים באלפבית של האוטומט. קשת נוספת, מתויגת בשאר $n-k$ הסמלים, תצא אל המצב q. q יהיה מצב מלכודת לא מקבל.
3. החזר את האוטומט $<M>$."

נותר להוכיח שהרדוקציה מתבצעת במקום לוגריתמי, וש-$<M>$ שייכת ל- $\overline{E_{\mathrm{DFA}}}$ אם ורק אם $<G, s, t>$ שייכת ל-*PATH*.