תרגיל בית 5
מגיש: אור דינר
ת.ז: 207035809

שאלה 1

$$L = \{< M, A >: there\ exists\ x \in L(A)\ s.t\ x \notin L(M)\}$$

נוכיח שהשפה L אינה ניתנת לקבלה באמצעות רדוקציה מהשפה $NOT - ACCEPT$.

נבנה מ״ט R שבהינתן $< M, w >$ קידוד של מ״ט מ-$NOT - ACCEPT$ – רצה, עוצרת, ופולטת קידוד $< M', A' >$ כך שיתקיים:

$$< M, w > \in NOT - ACCEPT \ \Leftrightarrow < M', A' > \in L$$

ומתקיים: $\boldsymbol{R(< M, w >) =< M', A' >}$

נגדיר:

1. M' מריצה את M על w ופועלת כמוה.
2. A' א״מ שמקבל מילים מאורך זוגי.

נוכיח שהרדוקציה נכונה:

- אם $< M, w > \in NOT - ACCEPT$, אז M לא מקבלת את w $\Leftarrow$ M' מריצה את $M(w)$ ודוחה תמיד או נכנסת ללולאה אינסופית. לכן לא תקבל אף מילה אז השפה שלה היא $L(M) = \emptyset$. A' מקבלת מילים מאורך זוגי אז יש מילים שמתקבלות על ידי A' ולא על ידי המ״ט M', לכן מתקיים: $R(x) =< \boldsymbol{M', A'} > \in L$.
- אם $< M, w > \notin NOT - ACCEPT$, אז M מקבלת את w $\Leftarrow$ M' מקבלת את כל המילים תמיד אז מתקיים $L(M') = \Sigma^*$. לכן לא קיימת אף מילה שהא״מ A' מקבל והמ״ט M' לא תקבל, אז לא מתקיים תנאי השפה ולכן: $R(x) =< M', A' > \notin L$.

שאלה 2

<u>טענה</u>: עבור L כריעה שפת הרישות שלה לא בבהכרח כריעה.

<u>הוכחה</u>: נסתכל בשפה

$$N - HALT = \{< M, w, n >: \boldsymbol{M}\ stops\ after\ \boldsymbol{n}\ steps\ for\ input\ \boldsymbol{w}\}$$

נניח שהשפה $L = prefix(N - HALT)$ כריעה ונראה רדוקציה מ- $HALT$ אליה, זה יראה לנו שגם $HALT$ כריעה אבל אנחנו יודעים שהיא לא.

נבנה מ״ט R שבהינתן $< M, w >$ קידוד של מ״ט מ-$HALT$ – רצה, עוצרת, ופולטת **רישא** של קידוד $< M', w', n >$ כך שיתקיים:

$$< M, w > \in HALT \Leftrightarrow \ some\_prefix(< M', A', n >) \in L$$

ומתקיים: $\boldsymbol{R(< M, w >)} = some\_prefix(< \boldsymbol{M', A', n} >)$

נגדיר שעבור $< M, w >$ R מחזירה את $< M, w >$**.**

נוכיח שהרדוקציה נכונה:

- אם $< M, w > \in HALT$, אז יתקיים ש-$< M, w >$ היא רישא של קידוד כלשהוא $N - HALT$ מכיוון שאם $< M, w > \in HALT$ אז בהכרח קיים n כלשהוא כך ש- $< M, w, n > \in N - HALT$.
לכן מתקיים ש- $< M, w > \in L$.
- אם $< M, w > \notin HALT$, אז לא ייתכן ש-$< M, w >$ תהיה רישא בקידוד כלשהוא ששייך ל-$N - HALT$ מכיוון שהמ״ט M לא תעצור על w עבור אף n ולכן לא תהיה שייכת לשפת הרישות של $N - HALT$. כלומר $< M, w > \notin L$.

* לא ייתכן מצב בו $< M, w >$ אינה עוצרת אבל עבור איזו מילה כלשהי $u \in \Sigma^*$ $< M, wu >$ כן עוצרת (לכאורה זה מצב בו יש רישא שתתקבל כי יש מילה אחרת שמתחילה ב-w ועוצרת). עבור המילה w המ״ט M תכינס ללולאה אינסופית ובגלל זה תיכנס ללולאה אינסופית עבור כל שתשורשר למילה w.

לפי תכונות רדוקציה אם L הייתה כריעה אז $HALT$ הייתה כריעה, אבל אנחנו יודעים שהיא לא, ולכן $N - HALT$ לא בהכרח כריעה.

שאלה 3

עבור מילה w נסמן את האות ה-i ב-w_i, עבור כל שפה L נגדיר :

$$EVEN(L) = \{w_2, w_4, \dots w_{|w|} : w \in L, |w| mod\ 2 = 0\}$$

<u>טענה</u> : אם L ב-P אז $EVEN(L)$ ב-NP.

<u>הוכחה</u> : נניח שהשפה L ב-P, כלומר קיימת לה מ״ט דטרמיניסטית שבהינתן מילה היא תדע לסווג אם היא ב-L או לא.

נראה שקיים מוודא פולינומי לשפה $EVEN(L)$ וזה יראה לנו שהיא ב-NP.

<u>רעיון</u> : ננסה לשחזר את המילה שממנה הקלט שלנו הגיע על ידי כך שננסה להכניס את כל הקומבינציות האפשריות של אותיות בין האותיות של x.

V(x,y): x= word, y = ?

>> loop for every possible combination of $\frac{|w|}{2}$ letters

>> create a new w s.t between every (and before the first one (w_1)) letter we insert a new letter from $\sum^*$ (Try to find the word x was created from)

>> Look for w in L (L is in P so it takes polynomial time)

>> if found – **return 1**

>> **return 0**

<u>נכונות</u> :

- $x \in EVEN(L)$, נרוץ על כל האופציות להכנסת אותיות בין כל 2 אותיות במילה x ובסוף נגיע למילה שתהיה שייכת ל-L ונחזיר 1.
- $x \notin EVEN(L)$, נרוץ על כל האופציות להכנסת אותיות בין כל 2 אותיות במילה x אבל לא נגיע למילה כזו (אחרת x הייתה ב-$EVEN(L)$) ונחזיר 0.

לכן השפה $EVEN(L)$ עבור L ב-P תהא גם היא ב-P ובפרט ב-NP.

שאלה 4

עלינו להוכיח שהשפות הבאות NP שלמות :

א. $4 - COLOR = \{< G >: G\ is\ 4\ colorable\}$

עלינו להראות שהשפה ב-NP בעזרת מוודא פולינומי :

V(x,y): x= graph <G>, y = Possible coloring

>> check for input validity

>> go over all edges of colored G and look for an edge with the same color in both ends

>> if found – the coloring is not legal – **return 0**

>>**return 1**

<u>נכונות</u> : האלגוריתם מקבל גרף וצביעה אפשרית עבור הגרף, כדי לבדוק האם הצביעה חוקית אנחנו עוברים על כל הקשתות בגרף ובודקים האם יש קשת שבה שני הצמתים באותו צבע, אם כן אז הצביעה לא חוקית, אחרת המשך. כשיסתיים האלגוריתם אפשר להחזיר 1 כי לא מצאנו בעיה בצביעה ולכן היא חוקית.

כעת נראה שהשפה היא NP שלמה באמצעות רדוקציה פולינומית מ-$3 - COLOR$.

נבנה מ״ט R שבהינתן $< G >$ קידוד גרף מ-$3 - COLOR$ – רצה, עוצרת, ופולטת קידוד של גרף $< G' >$ כך שיתקיים :

$$< G > \in 3 - COLOR \Leftrightarrow < G' > \in 4 - COLOR$$

ומתקיים : $R(< G >) =< G' >$

נגדיר את G' להיות G בתוספת צומת אחת שמחוברת לכל הצמתים בקשתות, הצומת החדשה תוכל להיצבע בצבע חדש ולא לפגוע בחוקיות הצביעה, וגם לאפשר 4-צביעה לגרף החדש.

נוכיח את נכונות הבנייה.

- אם $< G > \in 3 - COLOR$, קיימת ל-G צביעה תקינה ב-3 צבעים, על פי הבנייה של G' נוסיף צומת אחת מחוברת לכל הצמתים הקיימות בקשת, מכיוון שקיימת ל-G צביעה חוקית עם 3 צבעים, אפשר לצבוע את הצומת החדשה בצבע הרביעי והצביעה תהיה צביעה 4 חוקית כיוון שבכל קשת שני הצמתים יהיו בעלי צבע שונה. לכן $< G' > \in 4 - COLOR$.
- אם $< G > \notin 3 - COLOR$, לא קיימת ל-G צביעה תקינה ב-3 צבעים, על פי הבנייה של G' נוסיף צומת אחת מחוברת לכל הצמתים הקיימות בקשת, לא תיתכן 4 צביעה חוקית, במידה והייתה צביעה 4 חוקית, הצומת

החדשה בהכרח לא הייתה יכולה להיות בצבע הרביעי (אחרת היינו מוחקים אותה ומקבלים שהגרף המקורי 3 צביע), ובגלל שאותה הצומת מחוברת לכל הצמתים לא ייתכן שלא תהיה לה קשת ובה שני צמתים מאותו הצבע. לכן מתקיים $< G' > \notin 4-COLOR$.

הראינו ש-$4-COLOR$ NP שלמה על ידי רדוקציה פולינומית מבעיה NP שלמה.

ב. $HALF-CLIQUE = \{< G >: G\ has\ a\ clique\ of\ size\ \frac{|V|}{2}\}$

נראה שהשפה ב-NP על ידי יצירת מוודא פולינומי.

V(x,y): x= graph <G >, y = Clique

>> check if number of edges is $\frac{|V|}{2} \times \frac{|V|}{2}$

>> check for input validity by checking existence of edges like in y

>> if all edges appear in G, then it has a clique of size $\frac{|V|}{2}$ – **return 1**

>> otherwise **return 0**

<u>נכונות</u>: בהינתן קליקה בודקים האם הקשתות שלה נמצאות גם בגרף G והאם היא בגודל $\frac{|V|}{2}$, אם הן נמצאות אז יש ב-G קליקה בגודל $\frac{|V|}{2}$.

נראה שהשפה היא NP שלמה בעזרת רדוקציה פולינומית מ-$CLIQUE$.

נבנה מ"ט R שבהינתן $< G, k >$ קידוד גרף מ-$CLIQUE$ – רצה, עוצרת, ופולטת קידוד של גרף $< G' >$ כך שיתקיים:

$$< G, k > \in CLIQUE \Leftrightarrow < G' > \in HALF-CLIQUE$$

ומתקיים: $R(< G, k >) = < G' >$

נגדיר את G' להיות G בתוספת $\frac{|V|}{2} - k$ צמתים **(רק כאשר $\boldsymbol{k} > \frac{|V|}{2}$, אם יש קליקה שגדולה מ- $\frac{|V|}{2}$ אז יש גם קליקה בגודל $\frac{|V|}{2}$)** כך שמכל צומת חדשה נשלח קשתות לכל הצמתים ב-G, וכן לצמתים החדשים שיצרנו.

נוכיח את נכונות הבנייה.

- אם $< G, k > \in CLIQUE$, ל-G יש קליקה בגודל k, על פי ההגדרה של G', נוסיף עוד רכיב לגרף בעל $\frac{|V|}{2} - k$ צמתים כאשר מכל אחת מהצמתים נשלח קשת לכל שאר הצמתים בגרף, כולל הצמתים החדשים. משום שיש קליקה בגודל k כאשר הוספנו $\frac{|V|}{2} - k$ הגדלנו את הקליקה שגודלה k ב-$\frac{|V|}{2} - k$, שזה בעצם יוצר קליקה בגודל $\frac{|V|}{2}$, ואז בגרף G' יש קליקה מגודל חצי מכמות הצמתים. לכן $< G' > \in HALF-CLIQUE$.
- אם $< G, k > \notin CLIQUE$, אין קליקה מגודל k בגרף G ובפרט אין קליקות מגודל גדול מ-k (אחרת הייתה גם קליקה מגודל k). נניח שקיימת קליקה כלשהי בגודל $m \subseteq \mathbb{N}$ ובהכרח $m < k$. כאשר ניצור את גרף G' על ידי הוספת $\frac{|V|}{2} - k$ צמתים חדשים נקבל שיהיו בסך הכל $m + \frac{|V|}{2} - k$ קליקות, אבל משום ש-$m < k$ זה לא יהיה חצי מכמות הצמתים, ולכן $< G' > \notin HALF-CLIQUE$

בגלל שהראינו רדוקציה פולינומית מבעיה NP שלמה לבעיה NP גם $HALF-CLIQUE$ NP שלמה.

ג. $IND-SET=\{<G,l>: graph\ with\ independent\ group\ of\ at\ least\ size\ l\}$

נראה שקיים לקבוצה מוודא פולינומי על מנת להראות שהיא ב-NP.

V(x,y): x= graph <G ,l>, y = independent vertices subgraph
>> check if y has l vertices
>> for every vertex in y check if it has an edge with all other vertices of y
>> if it does, **return 0**
>> check if all edges of y are in the graph G.
>> for every vertex in y corresponding in G check if any of the other vertices of the independent subset are a neighbor, if yes then **return 0**
>> **return 1**

באלגוריתם אנחנו מקבלים תת גרף וצריך להחליט האם הוא בלתי תלוי בגודל l בגרף G, ראשית בודקים אם התת גרף בגודל l ואז בודקים שכנויות עבור כל התת גרף ב-G, אם מצאנו שני צמתים שכנים אז התת גרף לא בלתי תלוי כי יש קשת בין שני צמתים בתוך קבוצת הצמתים.

הרעיון הוא שאם בגרף יש קליקה מגודל k (קבוצת צמתים שכל זוג צמתים מחובר בקשת) להפוך את הקשתות (כך שקבוצת הצמתים תהפוך לצמתים שאין ביניהן קשת כלל).

נראה שהבעיה NP שלמה על ידי רדוקציה מ-$CLIQUE$.

נבנה מ״ט R שבהינתן $<G,k>$ קידוד גרף מ-$CLIQUE$ – רצה, עוצרת, ופולטת קידוד של גרף $<G',k'>$ כך שיתקיים:

$$<G,k>\in CLIQUE \Leftrightarrow <G',k'>\in IND-SET$$

ומתקיים: $R(<G,k>)=<G',k'>$

נגדיר את G' להיות המשלים של G, כלומר כל קשת שקיימת ב-G לא תהיה קיימת ב-G' ולהפך. $k=k'$.

נוכיח את נכונות הבנייה.

- אם $<G,k>\in CLIQUE$, ל-G יש קליקה בגודל k, על פי ההגדרה של G' כל שני זוגות צמתים בקליקה של G יהיו ללא שתות ביניהן, כלומר הצמתים של הקליקה ב-G יהוו הקבוצה הבלתי תלויה ב-G' (לכל שני צמתים בקבוצת הצמתים של הקליקה ב-G, ב-G' הן יהיו ללא קשת ביניהן, ועל פי הגדרה זו בדיוק קבוצה בלתי תלויה בגודל k). לכן $<G',k'>\in IND-SET$
- אם $<G,k>\notin CLIQUE$, אין קליקה מגודל k בגרף G לכן ב-G' לא תהא קבוצה בלתי תלויה בגודל k, מאותו היגיון כמו בכיוון הראשון . $<G',k'>\notin IND-SET$

בגלל שהראינו רדוקציה פולינומית מבעיה NP שלמה לבעיה NP גם $IND-SET$ NP שלמה.

שאלה 5

$$EMPTY_{CFG}=\{<G>: L(G)=\emptyset\}$$
$$ACCEPT_{CFG}=\{<G,w>: w\in L(G)\}$$

א. נבנה מ״ט דטרמיניסטית פולינומית שתדע לומר האם בהינתן דח״ה ומילה $<G,w>$ האם G מקבלת את w . ובכך היא תהא ב-P.

בהינתן דח״ה ומילה $<G,w>$ נבדוק עליהן האם G מקבלת את w:

1. נהפוך את הדח״ה G לא״מ M.
2. נבנה אס״ד שמקבל רק את המילה w, שייקרא D.
3. לפי סגירות של שפות ח״ה חיתוך של ח״ה ורגולרית היא ח״ה – ניצור א״מ חיתוך $M\cap D=M'$.
4. נמיר בחזרה את הא״מ M' לדח״ה G'.
5. נבדוק האם G' ריקה בעזרת הרצה שלה ב-$EMPTY_{CFG}$, אם היא ריקה (EMPTY קיבלה) – נדחה, אחרת נקבל.

<u>נכונות</u>:

- אם $<G,w>\in ACCEPT_{CFG}$, האלגוריתם ייצור את הדח״ה שמייצג את החיתוך של השפה שהיא w בלבד עם הדח״ה של G, זאת כדי לבדוק בעזרת השפה הנתונה $EMPTY_{CFG}$ האם המילה w נמצאת בשפה של הדח״ה ש-G מגדיר. כלומר כאשר החיתוך יהיה המילה w המ״ט שמכריעה את $EMPTY_{CFG}$ (בזמן פולינומי) תדחה ואז האלגוריתם יקבל את הדח״ה והמילה כי המילה היא חלק מהשפה שהדח״ה מגדיר.
- אם $<G,w>\notin ACCEPT_{CFG}$, האלגוריתם ייצור את הדח״ה שמייצג את החיתוך של השפה שהיא w בלבד עם הדח״ה של G, זאת כדי לבדוק בעזרת השפה הנתונה $EMPTY_{CFG}$ האם המילה w נמצאת בשפה של הדח״ה ש-G

מגדיר. כלומר כאשר החיתוך יהיה ריק המ״ט שמכריעה את $EMPTY_{CFG}$ (בזמן פולינומי) תקבל ואז האלגוריתם ידחה את הדח״ה והמילה כי המילה אינה חלק מהשפה שהדח״ה מגדיר.

ניתוח סיבוכיות:

1. נהפוך את הדח״ה G לא״מ M. זמן פולינומי ליצירת א״מ על ידי דח״ה
2. נבנה אס״ד שמקבל רק את המילה w, שייקרא D. זמן פולינומי
3. לפי סגירות של שפות ח״ה חיתוך של ח״ה ורגולרית היא ח״ה – ניצור א״מ חיתוך $M \cap D = M'$. זמן פולינומי
4. נמיר בחזרה את הא״מ M' לדח״ה G'. זמן פולינומי
5. נבדוק האם G' ריקה בעזרת הרצה שלה ב-$EMPTY_{CFG}$, אם היא ריקה (EMPTY קיבלה) – נדחה, אחרת נקבל. השפה $EMPTY_{CFG}$ ב-P לכן הבדיקה תיערך בזמן פולינומי גם היא

סה״כ קיבלנו שהאלגוריתם רץ בזמן פולינומי כנדרש, $ACCEPT_{CFG} \in P$.

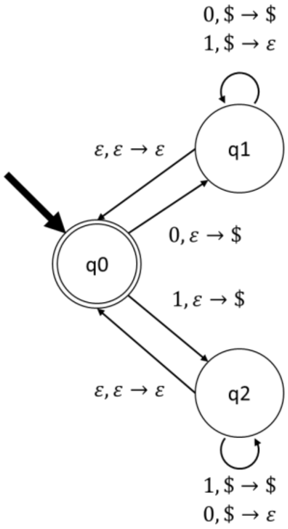
ב. על בסיס האלגוריתם מסעיף א׳ ניתן להסיק שכל שפה חסרת הקשר היא ב-P, משום שניתן יהיה להריץ את האלגוריתם על כל המילים בשפה עם הדח״ה שמגדיר אותה, ומשום שהפעולה הזו היא בזמן פולינומי, כל בדיקה האם מילה שייכת לשפה של הדח״ה היא בזמן פולינומי, כל שפה ח״ה שייכת ל-P.

שאלה 6

סווגו את השפות הבאות:

א. $L = \{w \in \{0,1\}^*: same\ amount\ of\ 0's\ and\ 1's\}$

השפה חסרת הקשר (אפשר לראות כי יש פה ״אלמנט של ספירה״ – דבר ששפה רגולרית לא יכולה לבצע), נראה זאת בעזרת בניית א״מ ושלילתה כרגולרית בעזרת למת הניפוח.



הסבר: אנחנו נרצה למדוד את ״המרחק״ בין כמות האפסים והאחדים בעזרת המחסנית, נכניס את התו ״\$״ כאשר כמות האפסים והאחדים אינה שווה.

המצב q_0 – מייצג את המצב בו כמות האחדים והאפסים שווה.

המצב q_1 – מייצג את המצב בו יש יותר אפסים מאחדים, כל פעם שרואים אפס מכניסים עוד דולר וכל פעם שרואים אחד מוציאים דולר, ברגע שהמחסנית ריקה אנחנו עוברים למצב q_0 בעזרת מעבר אפסילון.

המצב q_2 – מייצג את המצב בו יש יותר אחדים מאפסים, כל פעם שרואים אחד מכניסים עוד דולר וכל פעם שרואים אפס מוציאים דולר, ברגע שהמחסנית ריקה אנחנו עוברים למצב q_0 בעזרת מעבר אפסילון.

נפריך שהשפה רגולרית בעזרת למת הניפוח, נניח ש-L רגולרית אז היא צריכה לקיים את למת הניפוח, עבור n_0 המקיים את למת הניפוח נבחר את $w = 1^{n_0}0^{n_0}$, המילה נמצאת בשפה, ועבור $w = xyz$ מתקיים ש-$|xy| \le n_0$. כאשר ננפח את המילה ננפח רק את החלק שיש בו רק אחדים, וכשנוסיף עוד אחדים האיזון יופר וכמות האפסים לא תהיה שווה לכמות האחדים. אז השפה לא יכולה להיות רגולרית כי אינה מקיימת את למת הניפוח.

ב. $L = \{< G >: G \; does \; not \; contain \; a \; bigraph \; K_{3,3}\}$

השפה ב-P, נראה מ״ט דטרמיניסטית שבהינתן גרף תדע לבדוק האם יש בה גרף דו צדדי $K_{3,3}$ בזמן פולינומי.

1. נעבור על כל השלשות האפשריות של צמתים כך שהן אינן שכנות.
   1.1. נעבור על כל השלשות האפשריות (ללא 3 הצמתים שבחרנו באיטרציה הנוכחית) ונבדוק האם בין כל זוג צמתים מ-2 קבוצות הצמתים שבחרנו קיימת קשת.
   1.2. אם מצאנו 3 צמתים כך שהן יוצרות גרף דו צדדי 3,3 עם הצמתים שבחרנו באיטרציה הנוכחית – **נחזיר 0.**
2. **נחזיר 1** כאשר עברנו על כל האפשרויות ולא החזרנו 0.

<u>הסבר</u> : ננסה למצוא גרף דו צדדי מצורה $K_{3,3}$ בעזרת מעבר על כל זוג שלשות אפשריות של צמתים ובדיקה האם הן מהוות גרף דו צדדי מהצורה $K_{3,3}$ על ידי בדיקת קיום כל הקשתות הדרושות.

<u>סיבוכיות</u> : צעד 1. ירוץ ככמות השלשות בגרף – זמן פולינומי, צעד 1.1. ירוץ גם הוא ככמות השלשות בגרף. צעד 1.2 ירוץ בזמן קבוע (בדיקת קיום 9 הקשתות הדרושות ליצירת גרף דו צדדי $K_{3,3}$). לכן סה״כ האלגוריתם ירוץ בזמן פולינומי.

ג. $L = \{< M >: M \; accepts \; at \; most \; 1000 \; words \; within \; 1000 \; steps\}$

השפה ב-P, נראה הרצה מבוקרת של מ״ט שתדע להחליט אם מ״ט עונה על דרישות השפה L.

1. $i \rightarrow 0$
2. עבור על כל המילים עד אורך 1000 והרץ אותן למשך 1000 צעדים.
   2.1. בכל איטרציה אם המילה הנוכחית מתקבלת בפחות מ-1000 צעדים : $i \rightarrow i + 1$
3. אם $i > 1000$ דחה
4. אחרת קבל

<u>הסבר</u> : האלגוריתם רץ על כל המילים עד אורך 1000 (זמן קבוע כי יש כמות קבועה של מילים כאלו), בוכל איטרציה בודק האם המילה מתקבלת בתוך 1000 צעדים, אם כן מגדיל את i. בסוף כל הלולאה עברנו על כל המילים עד אורך 1000 וספרנו כמה מהן מתקבלות בתוך 1000 צעדים. אם הכמות גדולה מ-1000 דוחים, ואם פחות מ-1000 מקבלים.