פתרונות נבחרים לחלק שני

פרק 12

תרגיל 12.1: לשם הפשטות נבנה תחילה מכונה המחזירה כפלט את המספר $t_M(x)$ ביצוג אונרי. תהי M מכונת טיורינג. נבנה מכונת טיורינג, M', בעלת שני סרטים, אשר מחשבת את סיבוכיות הזמן של המכונה M. בהנתן קלט x, המכונה M' מריצה את המכונה M על הקלט x ובמקביל סופרת על הסרט השני (סרט הפלט) את מספר צעדי החישוב שהמכונה M עושה על הקלט x (בכל צעד M' תרשום 1 על סרט השני ותתקדם ימינה). סיבוכיות הזמן של M' שווה בדיוק לסיבוכיות הזמן של M. על מנת להחזיר את $t_M(x)$ ביצוג בינארי, יש להפעיל מכונה להעברה מיצוג אונרי ליצוג בינארי. ניתן לבנות מכונה כזו, אשר על קלט 1^n עוצרת עם פלט n (ביצוג בינארי!) בזמן $O(n^2)$.

תרגיל 12.2: המכונה M' המחשבת את T_M, מריצה את M על כל הקלטים מאורך n, סופרת את מספר הצעדים ש-M עושה בכל ריצה כזו ומחזירה כפלט את זמן הריצה המקסימלי. מתקיים כי M' עוצרת על קלט n רק אם M עוצרת על כל קלט מאורך n. חסם מלעיל על הסיבוכיות של המכונה M', מורכב מ:

א. $2^n \cdot T(n)$ - עבור ריצת M על 2^n הקלטים מאורך n, למשך $T(n)$ צעדים לכל היותר.

ב. תקורה (overhead) הנדרשת לייצור 2^n הקלטים על סרט עזר, להעתקתם אל הסרט בו מורצת M, ולמחיקת סרט זה לקראת העתקת הקלט הבא (ודא כי תקורה זו ניתן לחסום ע"י $2^n(2 \cdot n + 2 \cdot T(n))$).

חישוב יעיל יותר יכול להיות כאשר ידוע על המכונה M פרטים נוספים. למשל עבור מכונה M אשר רצה 100 צעדים על כל קלט (ללא קשר לארכו) ניתן לחשב חסם סיבוכיות ע"י מכונה המדפיסה על כל קלט את המספר 100.

פרק 13

תרגיל 13.1: יהי R יחס ב-*PolyFind*. מהגדרה 13.3 נובע כי בעית החיפוש של R ניתנת לפתרון בזמן פולינומיאלי, כלומר קיימת מכונת הטיורינג הפולינומיאלית, M, המובטחת מהגדרה 13.2. מהנתון בתרגיל (שלכל x מתאים y אחד לכל היותר המקיים $(x,y) \in R$) נובע כי המכונה M, עבור כל זוג $(x,y) \in R$, מקיימת $y = M(x)$. יהי Q, הפולינום החוסם את זמן ריצתה של המכונה M. לכל $(x,y) \in R$ מתקיים: $y = M(x) \le Q(|x|)$, ולכן לפי הגדרה 13.1 היחס R הינו חסום פולינומיאלית.

תרגיל 13.2:

(1) ביוון I: אם $f \in \mathbf{POLY}$ אז קיימת מכונת טיורינג, M_f המחשבת את f בסיבוכיות זמן פולינומיאלית. המכונה M_f עונה על דרישת הגדרה 13.2 ולכן בעית החיפוש של R_f ניתנת לפתרון בזמן פולינומיאלי, ולכן

(לפי הגדרה 13.3) $R_f \in PolyFind$.

כיוון II: אם $R_f \in PolyFind$ אז לפי הגדרה 13.3 בעית החיפוש שלו ניתנת לפתרון בזמן פולינומיאלי. המכונה המובטחת מהגדרה 13.2 מחשבת למעשה את הפונקציה f בזמן פולינומיאלי, ולכן $f \in \mathbf{POLY}$.

(2) יהי $R_f \in PolyFind$. נשתמש במכונה M המובטחת מהגדרה 13.2 לבנית המכונה הנדרשת בהגדרה 13.4 (שנסמנה M'). בהנתן קלט $x\#y$, המכונה M' תריץ את M על x, ותקבל את הקלט אם $y = M(x)$. מכיוון שלכל x מתאים בדיוק y אחד המקיים $(x,y) \in R_f$ (y זה הינו $f(x)$) נקבל כי $M(x) = f(x)$, ולכן M' מקבלת את הקלט $x\#y$ אם $y = f(x)$ (כלומר, אם $(x,y) \in R_f$).

תרגיל 13.3: נשתמש ברמז ונגדיר את היחס $R = R_1 \bigcup R_2$, כאשר $(x,y) \in R_1$ אם $x = y$, ו- $(x,y) \in R_2$ אם מתקיים $x \in L$ וגם $y = 1$, כאשר L הינה שפה שאינה ניתנת להכרעה. בעית החיפוש של R ניתנת לפתרון בזמן פולינומיאלי ע"י העתקת הזהות ולכן $R \in PolyFind$. בעית הזיהוי של R אינה ניתנת להכרעה. לו בעיה זו היתה ניתנת להכרעה, אז יכולנו להכריע את השפה L ע"י הפעלת המכונה המכריעה בעית זיהוי זו על הקלט $x\#1$ והכרעה בהתאם. לכן בפרט בעית הזיהוי של R אינה ניתנת לפתרון בזמן פולינומיאלי, ולכן $R \notin PolyVerify$.

תרגיל 13.4: המכונה M תשתמש במכונה (שנסמנה M') המובטחת לפי הגדרה 13.2 מהיות $R \in PolyFind$, ובמכונה (שנסמנה M'') המובטחת לפי הגדרה 13.4 מהיות $R \in PolyVerify$. יהי P הפולינום החוסם את זמן ריצתה של המכונה M'. בהנתן קלט x, המכונה M תנסה להריץ את M' על x לכל היותר $P(|x|)$ צעדים, אם M' לא עצרה על x במצב מקבל אז M דוחה את x. אחרת, M מריצה את M'' על הזוג $(x, M'(x))$ ומחזירה את $M(x)$ אם M'' מקבלת (אחרת, דוחה כמובן). מכיוון ש-M' ו-M'' הן מכונות פולינומיאליות, אז גם M פולינומיאלית. שים לב לחשיבות הרמז לאור ההערה למתקדמים שלאחר הגדרה 13.2.

תרגיל 13.5: ההוכחה זהה להוכחת תרגיל 13.5, רק שכאן המכונה, M, (המובטחת בתרגיל 13.5 מהיות $R \in PolyFind$) נתונה במפורש.

תרגיל 13.6: בהנתן מכונה פולינומיאלית, M, המקבלת שפה L, נבנה מכונה M' המכריעה את L. המכונה M', תשתמש בפולינום, P, החוסם את זמן הריצה של M. בהנתן קלט x, המכונה M' תנסה להריץ את M על x לכל היותר $P(|x|)$ צעדים. אם M קיבלה את x אז גם M' תקבל אותו, אחרת M' תדחה את x.

פרק 14

תרגיל 14.1: מהיות $L \in \mathbf{NP}$ נובע (מהגדרה 13.7) כי $R_L \in PolyVerify$. המכונה M המובטחת בתרגיל מקיימת

את תנאי הגדרה 13.2, ולכן $R_L \in PolyFind$. כעת אנו עומדים בתנאי תרגיל 13.4, המכונה הפולינומיאלית המובטחת מתרגיל זה מכריעה את השפה L (ודא) ולכן $L \in \mathbf{P}$.

<u>תרגיל 14.3:</u> תהי $L \in \mathbf{NP}$, אז קיים יחס דו-מקומי, R_L, המאפיין את L, וקיימת מכונת טיורינג פולינומיאלית, M_L אשר מזהה את R_L. יהי Q_{M_L} הפולינום ה"חוסם" את זמן ריצתה של M_L, ויהי Q_L הפולינום החוסם את האורך של y המקיים $(x,y) \in R_L$). בהנתן קלט x, המכונה M תריץ את M_L על (x,y), בכל הרצה $Q_{M_L}(|x|)$ צעדים, עבור כל המחרוזות y המקיימות $|y| \le Q_L(|x|)$. המכונה M תקבל את הקלט x אם באחת ההרצות (עבור אחד ה-yים) המכונה M קיבלה את הזוג (x,y). ברור כי אין טעם לבדוק את ריצתה של M_L עבור "עדים" ארוכים יותר, כי עדים כאלו אינם קיימים (מכיוון ש-R_L הינו יחס חסום פולינומיאלית). זמן הריצה של M חסום ע"י מספר המחרוזות שאינן ארוכות מ-$Q_L(|x|)$ (שמספרן $2^{Q_L(|x|)+1}$) כפול $Q_{M_L}(|x|)$ צעדי ריצה עבור כל מחרוזת כזו, בתוספת לזמן תקורה (overhead). זמן התקורה הוא הזמן הנדרש לייצר את המחרוזות הנ"ל (לפי הסדר הלקסיקוגרפי) על סרט עזר, ולהעתקתו לסרט בו מורצת M_L. נסמן ב-$T(n)$ חסם על זמן זה (ודא כי זמן זה הינו לינארי ב-$\max[Q_L(n),Q_{M_L}(n)]$ ולכן הינו פולינומיאלי ב-$n=|x|$). קיבלנו כי זמן הריצה של M על קלט באורך n חסום ע"י $\left(Q_{M_L}(n)+T(n)\right)\cdot 2^{Q_L(n)+1}$. יהי c הקבוע הדרוש לאי-השוויון $2^{n+c} \ge Q_{M_L}(n)+T(n)$ להתקיים עבור כל n (ודא כי קיים c כזה). כעת נקבע את P_L להיות: $P_L(n) = Q_L(n)+1+n+c$, ונקבל:

$$2^{P_L(|x|)} = 2^{Q_L(n)+1+n+c} = 2^{n+c}\cdot 2^{Q_L(n)+1} \ge \left(Q_{M_L}(n)+T(n)\right)\cdot 2^{Q_L(n)+1}$$

קיבלנו כי P_L עונה על דרישת התרגיל.

<u>תרגיל 14.5:</u> מכיוון ש-L_1 ניתנת לרדוקצית קוק ל-L_2, נובע כי קיימת מכונה, M_1, בעלת גישה לאוב לשפה L_2, הרצה בזמן פולינומיאלי ומכריעה את L_1. נחליף במכונה זו את הקריאות לאוב בהפעלת המכונה הפולינומיאלית, M_2, המכריעה את L_2 (שקיומה מובטח מהיות $L_2 \in \mathbf{P}$), על סרט האוב. מכיוון שמכונת האוב, M_1, הינה פולינומיאלית (באורך הקלט שלה) אורך ה"שאלות" שהיא שואלת את M_2 גם הוא פולינומיאלי באורך הקלט של M_1. מכאן נובע כי ריצת M_2 על שאלות אלו (שהיא פולינומיאלית בארכן) הינה פולינומיאלית באורך הקלט של M_1 (הרכבת שני פולינומים הינה גם-כן פולינום). כמו כן, M_2 מופעלת מספר פולינומיאלי (באורך הקלט של M_1) של פעמים ע"י M_1 ולכן זמן הריצה הנוסף שהיא מוסיפה על זמן ריצת M_1 הינו פולינומיאלי באורך הקלט של M_1 (מכפלה של פולינומים הינה פולינום). כלומר, קיבלנו כי המכונה המורכבת משילוב M_1 ו-M_2 הינה פולינומיאלית ומכריעה את L_1, ולכן גם $L_1 \in \mathbf{P}$.

<u>תרגיל 14.6:</u> תהי f הפונקציה המובטחת מהגדרה 14.4 המקיימת, $x \in L_1$ אם $f(x) \in L_2$, ותהי M_f המכונה המחשבת את f בזמן פולינומיאלי. המכונה הפולינומיאלית המכריעה את L_1 עם גישה לאוב לשפה L_2 פועלת כדלקמן: בהנתן קלט x, תפעיל את M_f על x, ותחזיר את תשובת האוב על השאלה $M_f(x)$.

<u>תרגיל 14.7:</u> המכונה הפולינומיאלית המכריעה את L_1 תפעל באופן דומה למכונה מתרגיל 14.6, רק שבמקום קריאה לאוב היא תפעיל את המכונה הפולינומיאלית המכריעה את L_2 על הקלט $M_f(x)$ ותחזיר את תשובתה. ודא (באופן דומה לתרגיל 14.3) כי המכונה המורכבת הינה פולינומיאלית.

<u>תרגיל 14.8:</u> תהיינה $L_1,L_2 \in \mathbf{P}$ ותהי M המכונה המכריעה את L_1 בזמן פולינומיאלי. המכונה המכריעה את L_1 עם גישה לאוב לשפה L_2 בזמן פולינומיאלי פועלת כדלקמן: בהנתן קלט x, המכונה תריץ את M על x ותחזיר את תשובתה (תוך התעלמות מהאוב). מכונת האוב הינה בדיוק המכונה M, ולכן הינה פולינומיאלית.

<u>תרגיל 14.9:</u> הרדוקציה תפעל לפי הרמז. יש להדגיש כי חשיבות התנאי שהשפות אינן טריביאליות, היא בקיום w_Y ו-w_N כנדרש. כמו בתרגיל 14.8, הרדוקציה תפעיל את M (המכריעה את L_1 בזמן פולינומיאלי) על הקלט הנתון, ותחזיר את w_Y או w_N אם M קיבלה את הקלט או לא קיבלה אותו, בהתאמה.

<u>פרק 15</u>

<u>תרגיל 15.3:</u> קל לראות כי $gSAT \in \mathbf{NP}$, כי (כרגיל) ה"עד" לשייכות קלט לשפה $gSAT$ יהיה הצבת האמת המספקת את הפסוק, וניתן בזמן פולינומיאלי לזהות האם ההצבה הינה אכן הצבה מספקת. כדי להוכיח **NP**-קושי נבנה רדוקציה מבעית SAT לבעיה $gSAT$ המבצעת את העתקת הזהות (למתקדמים: שים לב כי $gSAT$ הינה הרחבת הקלטים האפשריים ל-SAT, ולכן על פי הנאמר בפרק 17, אינה יכולה להיות "קלה" מ-SAT).

<u>תרגיל 15.4:</u> גם כאן קל לראות כי $3SAT \in \mathbf{NP}$. כרגיל, ה"עד" לשייכות קלט לשפה $3SAT$ יהיה הצבת האמת המספקת את הפסוק, וניתן בזמן פולינומיאלי לזהות האם ההצבה הינה אכן הצבה מספקת. כדי להוכיח **NP**-קושי נבנה רדוקציה מבעית $3SAT$ המקורית לבעיה זו (כלומר, $3SAT$ המתוקנת) המוסיפה לכל פסוקית המכילה פחות משלושה ליטראלים, ליטראלים שכבר קיימים בה.

<u>הערה:</u> בתרגיל 15.4 לא היתה דרישה שלא יופיעו שני ליטראלים זהים באותה פסוקית, ולכן הפתרון שלעי"ל הינו מספק. אם אכן נדרש כי לא יופיעו שני ליטראלים זהים באותה פסוקית, נוסיף לכל פסוקית המכילה פחות משלושה ליטראלים, ליטראל או שני ליטראלים (לפי הצורך) שערכם ע"י כל הצבה מספקת הוא F. על מנת להבטיח שערך הליטראל x_1 (שאינו מופיע בפסוק המקורי) הינו F בכל הצבה מספקת, נוסיף לפסוק את הפסוקיות הבאות: $(x_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge \overline{x}_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge \overline{x}_3) \vee (x_1 \wedge \overline{x}_2 \wedge \overline{x}_3)$ כאשר גם המשתנים x_2 ו-x_3 אינם מופיעים בפסוק המקורי.

<u>פרק 16</u>

<u>תרגיל 16.2:</u> ראשית יש לודא כי $\text{Clique} \in \mathbf{NP}$. בהנתן קלט, גרף G ושלם K, העדות לכך כי הגרף G מכיל קליק בגודל K תהיה הקליק עצמו. יש לעבור על כל $\binom{K}{2}$ הקשתות בקליק ולודא שהן אכן נמצאות בגרף. כדי לבדוק האם קשת נמצאת בגרף מספיק מעבר יחיד על הקלט, לכן הבדיקה יכולה להתבצע בזמן פולינומיאלי (ליתר דיוק $O(n^2)$, כאשר n הוא אורך הקלט).

נראה כי הבעיה Clique הינה **NP**-שלמה, ע"י רדוקציה אליה מהבעיה IS (שהיא כידוע **NP**-שלמה). בהנתן גרף $G=(V,E)$ ושלם K כקלט לבעיה IS, מכונת הרדוקציה תייצר את הגרף $G'=(V,E')$ המשלים ל-G (כלומר, לכל שני צמתים $u,v \in V$ מתקיים: $(u,v) \in E'$ אסם $(u,v) \notin E$) ואת השלם $K'=K$. ברור כי ביצוע הרדוקציה דורש זמן $O(n^2)$ ולכן הרדוקציה הינה פולינומיאלית. בהוכחת תקפות הרדוקציה, שים לב כי קבוצת צמתים מהווה קליק בגרף אסם היא קבוצה בלתי תלויה בגרף המשלים.

<u>תרגיל 16.3:</u>

א. נבצע את הרדוקציה $VC \propto_p VC/2$. בהנתן קלט G,K לבעית VC נפריד לשלושה מקרים:

I $K = \left\lfloor \frac{|V|}{2} \right\rfloor$ - נחזיר את G כפלט של פונקצית הרדוקציה.

II $K > \left\lfloor \frac{|V|}{2} \right\rfloor$ - הפלט של פונקצית הרדוקציה יהיה גרף המכיל בנוסף ל-G גם כוכב בגודל S (שים לב כי כל הקשתות בכוכב ניתנות לכיסוי ע"י הוספת מרכזו לכיסוי). את S נקבע כך ש-$S + |V| = 2 \cdot (K+1)$.

III $K < \left\lfloor \frac{|V|}{2} \right\rfloor$ - הפלט של פונקצית הרדוקציה יהיה גרף המכיל בנוסף ל-G גם קליק בגודל S (שים לב כי לשם כיסוי כל הקשתות בקליק בגודל S, יש להוסיף לכיסוי $S-1$ מצמתיו. את S נקבע כך ש-$S + |V| = 2 \cdot (K+S-1)$.

ודא כי הרדוקציה פולינומיאלית ותקפה.

ב. הרדוקציה פועלת בדומה לרדוקציה מסעיף א'. הערה למתקדמים: אותה רדוקציה תתאים עבור כל פונקציה, f, שהיא מונוטונית לא יורדת, המוגדרת מעל השלמים, הניתנת לחישוב, אשר לכל n מקיימת: $n^{\varepsilon} \le f(n) \le n - n^{\varepsilon}$ וכן $f(n+1) = f(n)$ או $f(n+1) = f(n) + 1$.

<u>תרגיל 16.4:</u> ההוכחה זהה להוכחת משפט 16.3, הן בהוכחה שהשפה $SC_d \in \mathbf{NP}$ והן ברדוקציה $VC_d \propto_p SC_d$. יש לשים לב כי כאשר נפעיל את פונקצית הרדוקציה שתוארה בהוכחת משפט 16.3, על גרף שדרגתו חסומה ע"י d, נקבל כי באוסף הקבוצות המתקבל גודל כל קבוצה חסום ע"י d (כי כל קבוצה היא קבוצת הקשתות הפוגעות

בצומת מסוים).

פרק 18

<u>תרגיל 18.1:</u> ההסתברות שהמכונה עוצרת לאחר t צעדים בדיוק, היא ההסתברות שהמכונה תבחר $t-1$ פעמים (באופן רצוף) בחוק השמאלי ובפעם ה-t-ית את החוק הימני. מאורע זה קורה בהסתברות $\frac{1}{2^t}$, ולכן ההסתברות שהמכונה תעצור היא: $\sum_{t=1}^{\infty}\frac{1}{2^t}=1$.

<u>תרגיל 18.2:</u> במקרה בו $|x|$ הינו אי-זוגי התשובה היא 0, מכיוון ש-$\frac{x}{2}$ אינו מספר שלם. במקרה בו $|x|$ הינו זוגי התשובה היא $\frac{1}{2^{|x|}}\cdot\binom{|x|}{|x|/2}$, מכיוון שזהו מספר האפשרויות לבחור את $\frac{|x|}{2}$ המקומות בהם $f_2(x)$ שונה מ-x (המתאים באופן חח"ע לריצות היוצרות פלט זה) מחולק במספר הריצות האפשריות של המכונה. כדי לקרב ביטוי זה נשתמש בנוסחת הקירוב של סטרלינג: $n!\approx\sqrt{2\pi n}\cdot\left(\frac{n}{e}\right)^n$.

בעזרת הקירוב נקבל:

$$\frac{1}{2^{|x|}}\cdot\binom{|x|}{|x|/2}=\frac{1}{2^{|x|}}\cdot\frac{|x|!}{((|x|/2)!)^2}\approx\frac{1}{2^{|x|}}\cdot\frac{\sqrt{2\pi|x|}\cdot\left(\frac{|x|}{e}\right)^{|x|}}{\pi|x|\left(\frac{|x|}{2e}\right)^{|x|}}=\sqrt{\frac{2}{\pi|x|}}$$

<u>תרגיל 18.3:</u>

1. בהנתן מכונה אקראית פולינומיאלית, M, המקבלת את השפה L, נבנה מכונה אקראית פולינומיאלית, $\overline{M}$, המקבלת את השפה המשלימה $\overline{L}$. בהנתן קלט x, המכונה $\overline{M}$ מפעילה את המכונה M על הקלט x. אם M עוצרת ומקבלת $\overline{M}$ עוצרת ודוחה, ואם M עוצרת ודוחה אז $\overline{M}$ עוצרת ומקבלת. כעת לכל $x\in L$, ההסתברות שהמכונה $\overline{M}$ דוחה את x הינה $\frac{2}{3}$ לכל הפחות, ולכל $x\notin L$, ההסתברות שהמכונה $\overline{M}$ מקבלת את x הינה $\frac{2}{3}$ לכל הפחות. קיבלנו כי $\overline{M}$ מקיימת את הגדרה 18.2 לגבי $\overline{L}$ ולכן $\overline{L}\in\mathbf{BPP}$.
2. בנית $\overline{M}$ נעשית בדיוק באותו אופן כמו בסעיף (1). שים לב כי על פי דיון מקביל לזה שנעשה בסעיף (1) (לגבי הגדרה 18.2) לגבי הגדרה 18.1, נקבל כי $\overline{L}\in\mathbf{ZPP}$ (שים לב כי תנאי (3) מהגדרה 18.1 מתקיים גם עבור $\overline{M}$).

<u>תרגיל 18.4:</u> רעיון ההוכחה הוא, בהנתן שפה $L \in \mathbf{ZPP}$ והמכונה M (המובטחת מהגדרה 18.1) יש לעצור את M בכל מקרה לאחר $Q(n)$ צעדים (n הוא אורך הקלט) ולהריצה פעם נוספת (שוב לא יותר מ-$Q(n)$ צעדים) במקרה בו M לא עצרה. ההסתברות ש-M לא תעצור בהרצה בודדת הינה $\frac{1}{2}$ לכל היותר, לכן ההסתברות ש-M לא תעצור בשתי הרצות רצופות הינה $\frac{1}{4}$ לכל היותר.במקרה ש-M לא תעצור בשתי הרצות רצופות, הקלט ידחה באופן שרירותי (ניתן באופן שרירותי להחליט גם אחרת). במקרה הגרוע ביותר, תטעה המכונה בכל ההרצות שהופסקו באופן שרירותי, אך בכל מקרה המכונה תפעל (תדחה או תקבל) באופן נכון בהסתברות $\frac{3}{4}$ על כל קלט ולכן תעמוד בדרישות הגדרה 18.2 ויתקיים $L \in \mathbf{BPP}$. שים לב כי הפולינום $Q_2(n)=2\cdot Q(n)$ חוסם את זמן ריצתה של המכונה שבנינו.

<u>תרגיל 18.5:</u> אם נרשה שהסתברות העצירה תהיה 0, אז כל שפה תהיה ב-**ZPP** והמכונה המתאימה לה בהגדרה 18.1 תהיה מכונה שאינה עוצרת כלל. אם נדרוש שהסתברות העצירה בזמן פולינומיאלי תהיה 1, אז המחלקה **ZPP** תהיה בדיוק המחלקה **P** כי נוכל לקחת לכל שפה במחלקה **ZPP** את המכונה המובטחת מהגדרה 18.1 ולבחור באופן דטרמיניסטי באיזה חוק להשתמש (למשל תמיד בחוק הראשון) במקרה של פיצול לא-דטרמיניסטי. נקבל מכונה פולינומיאלית המקבלת את השפה הנ"ל ולכן היא במחלקה **P**.

<u>תרגיל 18.6:</u> בהנתן מכונה אשר בהסתברות $\frac{2}{3}$ עוצרת בזמן פולינומיאלי (החסום ע"י הפולינום Q) עם פלט נכון, נוכל לעצור אותה בכל מקרה לאחר שהיא רצה $Q(|x|)$ צעדים (על הקלט x) ולענות תשובה שרירותית (למשל לקבל תמיד). עצירה זו לא תפגע בעובדה כי הפלט יהיה נכון על לפחות $\frac{2}{3}$ מהקלטים (שהריצה עליהם לא השתנתה).

<u>תרגיל 18.7:</u> נסתכל על אוסף הריצות האפשריות של המכונה מהגדרת **BPP** על קלט x, כעל מרחב הסתברותי (בו סדרות הטלות המטבע מתפלגות באופן אחיד) אשר מתוכו מספר ההרצות המשיבות נכונה על הקלט (כלומר, דוחות או מקבלות בהתאם) גדול ממש מחצי המרחב. נקבל כי ההסתברות לריצה נכונה על הקלט גדולה מ-$\frac{1}{2}$. נקרא להפרש בין הסתברות זו לבין $\frac{1}{2}$ בשם **היתרון** של אלגוריתם המכונה על הקלט x. ניתוח הסתברותי פשוט מראה כי מכונה אשר מריצה את המכונה מהגדרת **BPP** הרבה פעמים על הקלט, x, ומכריעה לפי רוב, משפרת יתרון זה. ניתן לשפר יתרון זה ככל שנרצה (עד ל-$\frac{1}{2}$ כמובן) ע"י הרצת המכונה מספר גדול יותר של פעמים. כדי לשמור על היעילות (כלומר, ריצה בזמן פולינומיאלי) בהגדרת **BPP** אנו מוגבלים להרצת המכונה מספר פולינומיאלי של פעמים. לא נביא כאן את הניתוח ההסתברותי המדויק כדי להראות כי החלפת $\frac{2}{3}$ בהגדרת **BPP** ב-$\frac{1}{P(|x|)}$ או ב-$\frac{1}{2}+1-2^{-|x|}$, אינו משנה את הגדרת **BPP**, אך נציין כי ניתן לעבור מיתרון אחד לשני ע"י הרצת המכונה מספר

פולינומיאלי של פעמים. למתעניינים נציין כי הניתוח ההסתברותי המדויק לגבי החלפת $\frac{2}{3}$ בהגדרת **BPP** ב-$\frac{1}{2}+\frac{1}{P(|x|)}$ יכול להעשות בעזרת אי-שיוויון Chebyshev, והניתוח לגבי החלפה ב-$1-2^{-|x|}$ נעשה בעזרת החסם (החזק יותר) של Chernoff.

<u>תרגיל 18.8:</u> במקרה של שינוי כזה כל שפה היתה במחלקה **BPP**, והמכונה המתאימה לה בהגדרה 18.2 היתה מכונה אשר על כל קלט היא מטילה מטבע, ולפי תשובתה מחליטה אם לקבל או לדחות (למשל הטלת 1 תוביל לקבלה והטלת 0 תוביל לדחיה). שים לב כי מכונה זו צודקת על כל קלט בהסתברות $\frac{1}{2}$ בדיוק.

<u>תרגיל 18.10:</u> תהא L שפה במחלקה **RP** ותהי M המכונה המובטחת מהגדרה 18.3. אם נסתכל על המכונה M כעל מכונה לא-דטרמיניסטית (תוך כדי התעלמות מההסתברות לבחירת צעדי החישוב הלא-דטרמיניסטים) נקבל כי M דוחה כל קלט $x\notin L$ וקיים לה מסלול חישוב מקבל עבור כל קלט $x\in L$ ולכן קיבלנו (על פי הגדרה 14.2) כי $L\in\mathbf{NP}$.

<u>תרגיל 18.12:</u> נחליף בהגדרת **RP** את $\frac{1}{2}$ ב-$\frac{1}{P(|x|)}$. תהי L שפה במחלקה **RP** לפי הגדרתה החדשה, ותהי M המכונה המתאימה לה לפי הגדרה זו. שים לב כי ההסתברות ש-M דוחה $x\in L$ הינה לכל היותר $1-\frac{1}{P(|x|)}$.
נבנה מכונה, M', שתקיים עבור L את ההגדרה המקורית. בהנתן קלט x, המכונה M' תריץ את M (על הקלט x) $P(|x|)$ פעמים (כאשר בכל פעם תטיל M מטבעות מחדש) ותקבל את x אם המכונה M קיבלה אותו באחת ההרצות. עבור $x\in L$, ההסתברות ש-M' תדחה את x הינה לכל היותר: $\left(1-\frac{1}{P(|x|)}\right)^{P(|x|)}$ (ודא כי אם $P(|x|)\geq 2$ אז הסתברות זו קטנה מ-$\frac{1}{2}$). עבור $x\notin L$, המכונה M תדחה את כל ההרצות (בהסתברות 1) ולכן M' תדחה תמיד את x. המכונה M' הינה פולינומיאלית (כי היא מריצה את M מספר פולינומיאלי של פעמים) ולכן $L\in\mathbf{RP}$ גם לפי ההגדרה המקורית.

כעת נחליף בהגדרת **RP** את $\frac{1}{2}$ ב-$1-2^{-|x|}$. כעת עבור שפה $L\in\mathbf{RP}$ לפי ההגדרה המקורית, נריץ את המכונה המובטחת מההגדרה $|x|$ פעמים, ונקבל מכונה המתאימה להגדרה החדשה (ודא זאת!).

עבור המקרה של **ZPP**, בהנתן מכונה M המקיימת את הגדרת **ZPP** עבור הסתברות p לעצירה לאחר $Q(n)$ צעדים (על קלט באורך n) נבנה מכונה M' אשר מריצה את M על הקלט הנתון מספר פעמים (עם חסם זמן $Q(n)$ עד אשר באחת ההרצות M עוצרת תוך $Q(n)$ צעדים. המכונה M' תענה את אותה התשובה שענתה M בריצה שבה היא עצרה. שים לב כי ככל שנריץ את M יותר פעמים ע"י M', המכונה M תקיים את הגדרת **ZPP** עבור הסתברות עצירה גדולה יותר. ערוך ניתוח הסתברותי (מקביל לזה שנערך עבור הגדרת **RP**) עבור הסתברויות העצירה שבתרגיל.

תרגיל 18.13: אם $\mathbf{NP} \neq co\text{–}\mathbf{NP}$, אז קיבלנו כי המחלקה **NP** אינה שווה למחלקה המשלימה לה ($co\text{–}\mathbf{NP}$) בעוד שהמחלקה **P** שווה למחלקה המשלימה לה (כלומר, $L \in \mathbf{P}$ אמ"מ $\overline{L} \in \mathbf{P}$) ולכן המחלקות **P** ו-**NP** אינן שוות.

תרגיל 18.14: תהי L שפה **NP**-שלמה השייכת למחלקה $co\text{–}\mathbf{NP}$. תהי $\overline{L'}$ שפה כלשהי במחלקה $co\text{–}\mathbf{NP}$. השפה L' שייכת למחלקה **NP** ולכן קיימת פונקציה, f (פונקצית הרדוקציה מ-L' ל-L) הניתנת לחישוב בזמן פולינומיאלי ומקיימת: $x \in L'$ אמ"מ $f(x) \in L$. כמו כן מתקיים (התנאי השקול לוגית) $x \notin L'$ אמ"מ $f(x) \notin L$, כלומר, $x \in \overline{L'}$ אמ"מ $f(x) \in \overline{L}$, ולכן f הינה רדוקציה מהשפה $\overline{L'}$ אל השפה $\overline{L}$. השפה $\overline{L}$ הינה במחלקה **NP** (כי $L \in co\text{–}\mathbf{NP}$) ולכן קיימת פונקציה g הניתנת לחישוב בזמן פולינומיאלית ומהווה רדוקציה מהשפה $\overline{L}$ אל השפה L. מטרנזיטיביות הרדוקציה הפולינומיאלית נקבל כי קיימת רדוקציה פולינומיאלית מהשפה $\overline{L'}$ אל השפה L. כעת נוכל לבנות מכונה לא-דטרמיניסטית פולינומיאלית המקבלת את $\overline{L'}$. המכונה תחשב את פונקצית הרדוקציה (מ-$\overline{L'}$ אל L) ותפעיל את המכונה הלא-דטרמיניסטית המקבלת את L (ולפיכך $\overline{L'} \in \mathbf{NP}$). קיבלנו כי כל שפה במחלקה $co\text{–}\mathbf{NP}$ נמצאת במחלקה **NP**, ומכיוון שאלו מחלקות משלימות ($L \in \mathbf{NP}$ גורר $\overline{L} \in co\text{–}\mathbf{NP}$ גורר $\overline{L} \in \mathbf{NP}$ גורר $L \in co\text{–}\mathbf{NP}$) נקבל כי $\mathbf{NP} = co\text{–}\mathbf{NP}$.

תרגיל 18.15: תהי L שפה במחלקה **BPP**, ותהי M המכונה האקראית הפולינומיאלית, המובטחת מהגדרה 18.2. נראה כעת הוכחה אינטראקטיבית לשפה L. המכונה V תהיה בדיוק המכונה M, והמכונה P תהיה מכונה כלשהי. האינטראקציה בין שתי המכונות תהיה ריקה ומהגדרה 18.2 ינבע כי לכל $x \in L$ המכונה V תקבל בהסתברות $\frac{2}{3}$ לכל הפחות, ולכל $x \notin L$ המכונה V תדחה בהסתברות $\frac{2}{3}$ לכל הפחות. הוכחה אינטראקטיבית זו עונה על תנאי הגדרה 18.6, ולכן $L \in \mathbf{IP}$

תהי L שפה במחלקה **NP**, ויהי R_L הפרדיקט המתאים לה. בהנתן קלט $x \in L$, המוכיח, P, יחפש את ה"עד", y, המקיים $(x,y) \in R_L$ וישלח אותו למוודא, V. המוודא יקבל את x, אמ"מ $(x,y) \in R_L$. נקבל כי לכל $x \in L$ המכונה V תקבל בהסתברות 1, וכן לכל $x \notin L$ המכונה V תדחה בהסתברות 1. קיבלנו הוכחה אינטראקטיבית ל-L, ולכן $L \in \mathbf{IP}$.

תרגיל 18.17: תהי L שפה אשר קיימת לה הוכחה אינטראקטיבית בה אין המוודא מטיל מטבעות. היחס, R_L, יוגדר: $(x,y) \in R_L$ אמ"מ y הינו קידוד של שיחה אפשרית על הקלט x בה המוודא מקבל את הקלט x. ברור כי לכל $x \in L$ קיים y כזה, כי קיימת שיחה בה הוא מתקבל. ל-$x \notin L$ לא קיימת שיחה מקבלת. לו היתה כזו, היה קיים מוכיח P^* אשר היה מנהל את השיחה על הקלט x על פי שיחה מקבלת ספציפית זו, וגורם למוודא (הפועל באופן דטרמיניסטי, ולכן תמיד ינהל את חלקו בדו-שיח באותו אופן) לקבל בכל שיחה את הקלט x בסתירה לתנאי הנאותות מהגדרה 18.6. פעולת P^* כזה לא תצלח במקרה בו המוודא מטיל מטבעות, מכיוון שהמוודא אינו דטרמיניסטי (אלא חלקו בדו-שיח יכול להיות תלוי בהטלות מטבע אקראיות) ולכן לא בהכרח יבצע את חלקו בדו-שיח כמו בשיחה הספציפית המקבלת. יש לשים לב לכך כי אורך כל שיחה, y, הוא פולינומיאלי באורך הקלט x, וכן כי ניתן לוודא

בזמן פולינומיאלי כי מחרוזת y מהווה קידוד של שיחה. לכן נקבל כי $R_L \in PolyVerify$, וכן $L = \{x : \exists y\ (x,y) \in R_L\}$, ולכן $L \in \mathbf{NP}$ (על פי הגדרה 13.7).

<u>תרגיל 18.18:</u> תהי $L \in \mathbf{PSPACE}$. קיימת מכונה M המקבלת את L העובדת בשטח פולינומיאלי באורך הקלט (החסום ע"י פולינום P). מכיוון ש-M (בהנתן קלט x) מוגבלת לעבוד רק ב-$P(|x|)$ התאים הראשונים של הסרט, מספר הקונפיגורציות האפשריות שהיא יכולה להגיע אליהן הינו סופי. מספר זה (שנסמנו ב-K) חסום ע"י המכפלה של מספר האפשרויות לתוכן הסרט ($|\Sigma|^{P(|x|)}$) מספר האפשרויות למיקום הראש ($P(|x|)$) ומספר מצבי המכונה האפשריים ($|Q|$). המכונה $\overline{M}$ המקבלת את $\overline{L}$ עובדת כדלקמן: בהנתן קלט x, המכונה $\overline{M}$ מריצה את M על x למשך K צעדים לכל היותר, ומקבלת אותו אם המכונה M לא קיבלה אותו. שים לב כי אם המכונה M רצה על x יותר מ-K צעדים (ולא חרגה מ-$P(|x|)$ התאים הראשונים) קיימת קונפיגורציה בה המכונה היתה יותר מפעם אחת, ולכן המכונה בלולאה אינסופית. אם נתיר בהגדרת **PSPACE** לחרוג מ-$P(|x|)$ צעדים על קלט x שאינו בשפה, המכונה $\overline{M}$ המפעילה את M תוכל לגלות "חריגה" כזו ולקבל את הקלט. ודא כי $\overline{M}$ רצה בשטח פולינומיאלי (כלומר, ודא כי תחזוק המונה הסופר את מספר צעדי ריצת M דורש שטח פולינומיאלי).

<u>תרגיל 18.19:</u> תהי L שפה במחלקה **BPP**, תהי M המכונה האקראית הפולינומיאלית, המובטחת מהגדרה 18.2, ויהי Q הפולינום החוסם את זמן ריצתה. בהנתן קלט x, המכונה הדטרמיניסטית, M', המכריעה את L בשטח פולינומיאלי, מריצה את M על x על כל סדרות הטלות המטבע שארכן $Q(|x|)$ ומכריעה על פי רוב (כלומר, אם M קיבלה את x ברוב הריצות אז M' מקבלת את x, אחרת M' דוחה את x). כל הרצה של M על x דורשת שטח פולינומיאלי, ומכיוון שההרצות הן סדרתיות ניתן לבצע את כולן על אותו מקום בסרט, ולכן מספיק ל-M' שטח פולינומיאלי (על אף שמספר ההרצות הינו אקספוננציאלי). שים לב כי מכיוון ש-M עונה תשובה נכונה לפחות על $\frac{2}{3}$ מהטלות המטבע בחירת הרוב הינה מוצדקת.

<u>תרגיל 18.22:</u> תהי $\#_R$ פונקציה במחלקה **#P**, יהי R היחס הדו-מקומי המתאים לה, תהי M המכונה המכריעה על קלט (x,y) האם $(x,y) \in R$ בזמן פולינומיאלי ב-$|x|$, ויהי P הפולינום אשר לכל $(x,y) \in R$ מתקיים $|y| \le P(|x|)$. המכונה M' אשר תחשב את הפונקציה $\#_R$ תפעל כדלקמן: בהנתן קלט x, המכונה M' תייצר את כל המחרוזות, y, המקיימות $|y| \le P(|x|)$, על כל אחת מהן תריץ את M על (x,y), תספור את מספר ההרצות המקבלות, ותחזיר מספר זה כפלט. גם כאן ההרצות סדרתיות ויכולות להשתמש באותו סרט עבודה, ולכן מהיות M פולינומיאלית נובע כי מספיק שטח פולינומיאלי על מנת לבצע את החישוב (ודא כי המניה דורשת שטח פולינומיאלי).

<u>תרגיל 18.23:</u> תהי L שפה במחלקה **NP**, ויהי R_L הפרדיקט המתאים לה. מהנחת התרגיל נובע כי ניתן לחשב את $\#_R(x)$ בזמן פולינומיאלי. בהנתן קלט x, המכונה הפולינומיאלית הדטרמיניסטית, M, המכריעה את L, תחשב את

$\#_R(x)$ ואם התקבלה תוצאה גדולה מ-0 (כלומר, קיימת מחרוזת y, המקיימת $(x,y)\in R$) המכונה M תקבל את x. אחרת, M תדחה אותו.

<u>תרגיל 18.25:</u>

א. שתי הכלות אילו הינן טריביאליות כי מכונה דטרמיניסטית היא בפרט מכונה לא-דטרמיניסטית בעלת דרגת פיצול 1.

ב. גם שתי הכלות אילו הינן פשוטות, מכיוון שכל מכונה העובדת בזמן $t(n)$ אינה יכולה להשתמש בשטח גדול מ-$t(n)$ כי רק על מנת לעבור על כל השטח (הגדול מ-$t(n)$) היא זקוקה ליותר מ-$t(n)$ צעדי חישוב.