1. בנימין, גד ודן סטודנטים בקורס חישוביות, והחליטו לחקור את השפה
$$L = \{< G >|G \text{ is undirected graph with } |V(G)| - 10 - size\ clique\}$$

בנימין אמר: "השפה ב $RE \setminus R$. אוכיח על ידי לכסון".
גד אמר: "השפה ב P. אבנה אלגוריתם דטרמיניסטי פולינומי".
דן אמר: "השפה ב NP. אבנה אלגוריתם אי דטרמיניסטי פולינומי שזמנו עד ריבועי באורך הקלט".

לכל טענה קבעו האם היא נכונה או שגויה. הוכיחו את תשובתכם (נניח שקבעתם כי דן צודק, אזי יש לבנות אלגוריתם אי דטרמיניסטי פולינומי). (24 נקודות)

פתרון:
בנימין טועה. ניתן להכריע את השפה, כפי שנראה עוד מעט.

<u>גד צודק</u>, השפה אכן ב-P. ההוכחה תתבסס על השיוויון הבא שהוכח בקורס "מבנים דיסקרטים":
$$\binom{n}{k} = \binom{n}{n-k}$$

לכן, בתרגיל שלנו מתקיים
$$\binom{|V(G)|}{|V(G)| - 10} = \binom{|V(G)|}{10} = \theta(|V(G)|^{10})$$

להלן אלגוריתם דטרמיניסטי לשפה:
- עבור על כל קבוצות הקודקודים בגודל $|V(G)| - 10$.
- בכל קבוצה כזו, בדוק האם היא מהווה קליקה. אם כן, קבל.
- דחה.

האלגוריתם פולינומי: יש $\theta(|V(G)|^{10})$ קבוצות קודקודים, ובכל אחת עוברים על $|V(G)|^2$ זוגות קודקודים כדי לבדוק קיום צלעות. סה"כ פולינומי.
אם $< G >\in L$ אזי קיימת בגרף קליקה שמקיפה את כל הצלעות פרט ל-10 מתוכן. לכן, כשנגיע לקבוצה הנכונה, האלגוריתם יקבל.
אם $< G >\notin L$ אזי לא קיימת בגרף קליקה מספיק גדולה, ולכן האלגוריתם יעבור על כל הקבוצות וידחה.

<u>גם דן צודק</u>, כפי שלמדנו, $P \subseteq NP$. לפי הגדרת השאלה, יש גם לבנות אלגוריתם אי-דטרמיניסטי פולינומי.
המכונה N על הקלט $< G >$:
- תנחש קבוצת קודקודים בגודל $|V(G)| - 10$.
- תוודא כי בין כל שני קודקודים קיימת צלע. אם אחת הצלעות חסרה – דחה.
- קבל.

האלגוריתם ריבועי באורך הקלט (ניחוש ליניארי, ועוד זמן ריבועי לבדיקת צלעות).
אם $< G >\in L$ אזי קיימת בגרף קליקה מספיק גדולה, ולכן N תוכל לנחש את הקבוצה הזו ולענות "כן".
אם $< G >\notin L$ אזי לא קיימת בגרף קליקה מספיק גדולה, ולכן לכל ניחוש N תענה "לא".

2. בשאלה זו, נניח שקיים אלגוריתם דטרמיניסטי פולינומי (הממומש במכונת טיורינג M_4) המכריע את השפה
$$4 - COL = \{< G >|G \text{ is undirected graph that is } 4 - colorable\}$$
(שפת כל הגרפים שניתן לצבוע בארבעה צבעים).
א. מה משמעות ההנחה לגבי שאלת $P = NP$? (3 נקודות)
ב. היעזרו באלגוריתם כדי למצוא צביעה חוקית בארבעה צבעים. אם אין כזו, האלגוריתם יחזיר "אין צביעה חוקית". הסבירו את פעילות האלגוריתם במילים (אין צורך להוכיח נכונות)

אך ספקו ניתוח סיבוכיות זמן ריצה מדויק ככל יכולתכם.
הנחיה: כווצו זוג קודקודים לקודקוד בודד. (17 נקודות)

פתרון:

א. היות והוכח במהלך הקורס כי $3-COL \in NPC$ וכי $3-COL \leq_P 4-COL$ (וקל להראות כי $4-COL \in NP$) אזי משמעות ההנחה היא כי $P=NP$ (נמצאה שפה NPC שגם ב-P, אז מכל שפה אחרת ב NP יש רדוקציה אליה, ולכן לכולן יש פתרון פולינומי דטרמיניסטי).

ב. יהי M_4 האלגוריתם (מ"ט) המכריע את הבעיה $4-COL$ בזמן פולינומי $t(\cdot)$. נבנה בעזרתו אלגוריתם המוצא פתרון לבעיה בזמן פולינומי.

הרץ את $M_4(<G>)$. אם הוחזר "אין צביעה חוקית", החזר "אין צביעה חוקית".
- אם $|V(G)| \leq 4$ החזר "הצביעה היא $1,2,\dots,|V(G)|$".
- יהי $G_j \coloneqq G$.
- כל עוד $|V(G_j)| > 4$:
  1. בחר קודקוד, ובחר קודקוד שאינו מחובר אליו[1], וכווץ את שניהם לקודקוד יחיד. (כל הצלעות שלהם עדיין קיימות)
  2. הגדר את הגרף החדש בתור G_j.
  3. אם $M_4(<G_j>) = 0$: חזור לגרף הקודם, ובחר זוג קודקודים אחרים (שאינם שכנים).
  4. המשך כך עד שתמצא G_j עבורו $M_4(<G_j>) = 1$. (כלומר, עד שיהיה קודקוד שניתן לכווץ). שמור תיעוד של האינדקסים המקוריים שכווצו בזוג זה.
- כל הקודקודים שכווצו יחד מהווים מחלקת צבע. היעזר בתיעוד הכיווצים כדי לשחזר את הצביעה- כל מי שכווץ לקודקוד v_i מקבל צבע i (יש רק ארבעה קודקודים בסוף הריצה). זהו הפלט (צביעה לכל הקודקודים, לפי התיעוד של הכיווצים)

נכונות: (לא נדרש, אלא רק הסבר)
אם $<G> \in 4-COL$ אזי קיימת צביעה בארבעה צבעים. האלגוריתם בונה את הקבוצות הבלתי תלויות בתור קודקודים המייצגים קבוצה, עד שהוא יגיע לבדיוק ארבע קבוצות בלתי תלויות. אלו הן מחלקות הצבע. בכל שלב האלגוריתם בודק האם הוספת קודקוד ספציפי לקבוצה שומרת על הצביעה החוקית או הורסת אותה, ולפי זה מחליט האם לצרף או לא.
אם $<G> \notin 4-COL$ אזי האלגוריתם יחזיר "אין צביעה חוקית" כבר בהתחלת הריצה.

זמן ריצה:
במקרה הגרוע, בכל פעם האלגוריתם יעבור על $O(\binom{|V(G)|}{2}) = |V(G)|^2$ זוגות, ובכל אחד מהם יריץ $t(\cdot)$ פעולות מהמכונה M_4. הלולאה מתבצעת $|V(G)|-5$ איטרציות, ולכן נקבל זמן ריצה $O\left(|V(G)|^3 \cdot t(\cdot)\right)$.

3. סווגו את השפות הבאות למחלקות $(R, RE \setminus R, coRE \setminus R, \overline{RE \cup coRE})$. הוכיחו את תשובתכם באופן מלא (רדוקציות, רייס וכדומה). (10 נקודות לכל סעיף)

$L_1 = \{<M>|L(M) = 17\}$
$L_2 = \{<M_1>,<M_2>|\exists n > 0 \text{ such that } |L(M_1) \cap L(M_2)| \geq n\}$
$L_3 = \{<M_1>,<M_2>,k|\exists x \in \{0,1\}^* \text{ of even length such that } M_1$
$\text{accepts } x \text{ within } k \text{ steps, and } M_2 \text{ rejects } x \text{ within } k \text{ steps}\}$

[1] אם אין זוג קודקודים שאינם שכנים, ויש יותר מארבעה קודקודים, הגרף אינו 4-צביע.

פתרון:

$L_1 \in \overline{RE \cup coRE}$. נוכיח בעזרת משפט רייס כי $L_1 \notin coRE$, ובעזרת רדוקציה כי $L_1 \notin RE$.

נגדיר את מחלקת השפות $S = \{L \in RE || L| = 17\}$. לכך מתאימה השפה
$L_1 = L_S = \{< M > | L(M) = 17\}$.

התכונה אינה טריוויאלית ($S \neq \phi,\ S \neq \Sigma^*$) ולכן $L_S \notin R$. (לא חייבים)
נשים לב כי $\phi \notin S$ ולכן $L_S \notin coRE$ לפי משפט רייס המורחב.
נותר רק לבצע רדוקציה מ $\overline{HP}$ כדי להראות כי $L_S \notin RE$ ולקבל את הטענה מראש התרגיל.

$$f(< M >, < x >) \rightarrow < M_x >$$

המכונה M על הקלט y:
מריצה את M על x למשך $|y|$ צעדים.
אם M לא עצרה וגם $y \in \{0, 0^2, \dots, 0^{17}\}$ קבל. אחרת (כן עצרה, או y שונה) דחה.

הרדוקציה מלאה (מוגדרת לכל קלט) וניתנת לחישוב (בפרט, אין בה הרצת מכונה שעלולה לא לעצור). נוכיח כי היא תקפה:

אם $< M >, < x > \in \overline{HP}$ אזי M לא עוצרת על x. לכן, היא לא תעצור עליו תוך $|y|$ צעדים, ואז $L(M_x) = \{0, 0^2, \dots, 0^{17}\}$ כלומר $|L(M_x)| = 17$ ולכן $< M_x > \in L_1$.
אם $< M >, < x > \notin \overline{HP}$ אזי M עוצרת על x. בפרט, היא עוצרת עליו תוך $|y|$ צעדים. לכן המכונה דוחה כל y כך ש $L(M_x) = \phi$ ולכן $< M_x > \notin L_1$.

ולכן הוכחנו כי $L_1 \in \overline{RE \cup coRE}$.

השפה $L_2 \in RE \setminus R$. נבנה מכונה מקבלת, עם הרצה מבוקרת:

עבור $i = 0, 1, \dots$:

- עבור $j = 0, \dots, i$
  - הרץ את המילה ה-j למשך i צעדים על M_1 ועל M_2. אם שתיהן קיבלו – קבל.

נכונות:
$< M_1 >, < M_2 > \in L_2$ אזי קיימת לפחות מילה אחת שמתקבלת בשתי השפות. לכן, ההרצה המבוקרת תמצא מילה אחת כזו ותקבל. ($n = 1 > 0$)

$< M_1 >, < M_2 > \notin L_2$ אזי אין אפילו מילה משותפת אחת, וההרצה המבוקרת תרוץ לנצח.

כעת נראה כי $L_2 \notin R$ ברדוקציה מ HP:
$$f(< M >, < x >) \rightarrow < M_x, M_x >$$

המכונה M_x על קלט y:
- מריצה את M על x.
- מקבלת את y.

הרדוקציה מלאה (מוגדרת לכל קלט) וניתנת לחישוב (בפרט, אין בה הרצה של מכונה שעלולה לא לעצור). נוכיח כי היא תקפה:

אם $< M >, < x > \in HP$ אזי M עוצרת על x. לכן $L(M_x) = \Sigma^*$. ומכאן $L(M_x) \cap L(M_x) = \Sigma^*$ ובפרט לכל $n > 0$ מתקיים $|L(M_x) \cap L(M_x)| \geq n$. לכן $< M_x >, < M_x > \in L_2$.

אם $< M >,< x >\notin HP$ אזי M לא עוצרת על x, ומכאן $L(M_x) = \phi$. בפרט, לכל $n > 0$ מתקיים $|L(M_x) \cap L(M_x)| = 0 < n$ ולכן $< M_x >,< M_x >\notin L_2$.

$L_3 \in R$! נבנה מכונה מכריעה:
אבחנה: אם המכונה רצה k צעדים, היא לא קוראת מילה שאורכה גדול ממש מ-k.
עבור על כל המילים x שאורכן זוגי עד אורך k, בסדר לקסיקוגרפי.
- הרץ את M_1 על x למשך k צעדים. אם M_1 לא עצרה, או דחתה, עבור למילה הבאה.
- אחרת, הרץ את M_2 על x למשך k צעדים. אם דחתה – קבל. אחרת, עבור למילה הבאה.
- דחה.

המכונה עוברת על כמות סופית של מילים, ובכל אחת היא מקדישה כמות סופית של צעדים ולכן היא מכונה מכריעה לשפה!
אם $< M_1, M_2, k >\in L_3$ אזי קיימת מילה באורך זוגי הדורשת עד k צעדים בשתי המכונות, ולכן האלגוריתם יזהה שהיא מתקבלת ב M_1 ונדחת ב M_2 ויקבל.
אם $< M_1, M_2, k >\notin L_3$ אזי לא קיימת מילה באורך זוגי כנדרש, והאלגוריתם יעבור על כל המילים וידחה.

4. תהי

$$most - CNF = \{\phi | \phi \text{ is a Boolean formula such that more than } \frac{15}{16} \text{ of the assignments to it are satisfying}\}$$

א. הוכיחו כי $most - CNF \in NPh$ (12 נקודות)
ב. האם $most - CNF \in NP$? הסבירו את תשובתכם, אך אין צורך להוכיח. (4 נקודות)

פתרון:

א. נראה רדוקציה $CNF - SAT \leq_P most - CNF$ ומכאן נקבל $most - CNF \in NPh$.

$$f: \phi - \phi'$$

__לכל__ פסוקית $C_i \in \phi$ נוסיף עוד 4 משתנים __חדשים__ $a \vee b \vee c \vee d$.
הרדוקציה פולינומית: לכל פסוקית הוספנו עוד 7 תווים. $O(m)$ (כמות הפסוקיות). נוכיח שהיא תקפה:
$\phi \in CNF - SAT$ אזי קיימת השמה מספקת. השמה זו עדיין מספקת את ϕ'. נוספו אליה כל ההשמות בהן $a = T$ or $b = T$ or $c = T$ or $d = T$, כלומר, עוד בדיוק $\frac{15}{16}$ מההשמות.
לכן, ב ϕ' יש יותר מ $\frac{15}{16}$ השמות מספקות.
$\phi \notin CNF - SAT$ אזי אין השמה מספקת. לכן, ב ϕ' יש בדיוק $\frac{15}{16}$ השמות מספקות, ולא יותר מכך, שכן אין דרך לספק את הפסוק כאשר $a = b = c = d = F$.

ב. כדי להעיד על כך שיותר מ $\frac{15}{16}$ ההשמות מספקות, צריך לנחש את כל ההשמות האלו (או, לקבל עד מספיק ארוך). אבל יש n משתנים, ולכן קיימות 2^n השמות. מכאן, כתיבת $\frac{15}{16}$ מההשמות מהווה $\frac{15}{16} \cdot 2^n$! זה אקספוננציאלי בקלט. (גם כתיבת ההשמות הלא מספקות, "רק" עד $\frac{1}{16}$ זה עדיין אקספוננציאלי). לכן, לא נוכל בוודאות לקבוע שייכות ל-NP או ל-$coNP$.
יתכן וכן, אך הדרכים הרגילות אינן עובדות במקרה זה.

5. תהי

$$L_{Ham3} = \{< G > | G \text{ is undirected graph that has no more than } 3 \text{ different Hamiltonian cycles}\}$$

הוכיחו/הפריכו: $L \in coNP$ (10 נקודות)

פתרון:

נבנה לשפה $\overline{L_{Ham}}$ מ"ט א"ד פולינומית, מה שיוכיח שהמשלימה נמצאת ב-NP. משום כך, לפי הגדרת $coNP$ נקבל שהשפה המקורית במחלקה $coNP$.

פרמוטציה על קודקודי הגרף תראה באופן הבא: $(v_1, v_2, \dots, v_n)$, בהנחה שיש n קודקודים בגרף.

<u>$M(<G>)$:</u>

1. נחש 4 פרמוטציות **שונות** של קודקודי הגרף. ($v_1, v_2, v_3 \equiv v_3, v_2, v_1$)
2. עבור כל פרמוטציה בדוק:
   - 2.1. האם לכל $i \in [n-1]$ מתקיים $(v_i, v_{i+1}) \in E(G)$. אם לא, דחה.
   - 2.2. בדוק האם מתקיים $(v_n, v_1) \in E(G)$. אם לא, דחה.
3. קבל.

<u>זמן ריצה:</u> ניחוש 4 פרמוטציות מצריך $O(4n) = O(n)$ פעולות.
וידוא קיום $4n$ צלעות מצריך זמן לינארי גם כן- סה"כ האלגוריתם לינארי ולכן גם פולינומי.

<u>נכונות:</u>

אם $< G > \in \overline{L_{HAM3}}$ אז יש ב-G לפחות 4 מעגלי המילטון. אם כך, באחד המסלולים ינוחשו ארבעת הפרמוטציות הנכונות- הן 'יעברו' את הבדיקה שהן אכן מהוות מעגלי המילטון, המכונה תקבל אותן. יהיה לפחות מסלול אחד מקבל בעץ החישוב של M על $< G >$ ולכן $< G > \in L(M)$.

אם $< G > \notin \overline{L_{HAM3}}$ אז אין ב-G יותר מ-3 מעגלי המילטון. אם כך, בכל מסלול חישוב של M ינוחשו ארבע פרמוטציות כשלפחות אחת מהן לא מהווה מעגל המילטוני- בבדיקה האם פרמוטציה זו מהווה מעגל המילטוני תחסר לנו צלע ולכן נדחה. הדבר קורה עבור כל 4 פרמוטציות ולכן כל מסלול חישוב ייגמר במצב דוחה. סה"כ כל המסלולים דוחים בעץ החישוב של M על $< G >$ ולכן $< G > \notin L(M)$.