תרגול מס' 12 – אלגוריתם אימות, מחלקת שפות NP, רדוקציות

מחלקות סיבוכיות P ו-NP

שפה $L \in P$ אם קיים **אלגוריתם המכריע** את L בזמן פולינומיאלי.
במילים אחרות, P מהווה את אוסף השפות בעלות אלגוריתם "יעיל" (פולינומיאלי) להכרעתם.

שפה $L \in NP$ אם קיים **אלגוריתם אימות** $V_L(x,y)$ ופולינום $P_L(x)$ כך ש:

1. אם $x \in L$ אזי קיים עד y כך ש- $|y| \le P_L(|x|)$ ו- $V_L(x,y) = TRUE$.
2. אם $x \notin L$ אזי לכל עד y מתקיים $V_L(x,y) = FALSE$.
3. האלגוריתם $V_L(x,y)$ הוא פולינומיאלי,
   כלומר פולינומיאלי ב-$|y|+|x|$ ולכן פולינומיאלי ב-$|x|$.

אלגוריתם אימות

אלגוריתם A הוא אלגוריתם אימות לשפה L מעל $\sum$ מקבל שני פרמטרים: קלט x ועד אימות y המקיים:

1. אם $x \in L$ אזי קיים **עד** $y \in \Sigma^*$ כך ש- $A(x,y) = TRUE$.
2. אחרת, לכל $y \in \Sigma^*$ מתקיים $A(x,y) = FALSE$.

רדוקציה פולינומיאלית

נאמר כי שפה A ניתנת לרדוקציה פולינומיאלית (בזמן) לשפה B ונסמן $A \le_P B$ אם קיימת פונקציית העתקה f המקיימת:

1. $x \in A \Leftrightarrow f(x) \in B$.
2. קיים אלגוריתם פולינומיאלי המחשב את f.

תרגיל: סגירות מחלקת השפות NP

נתונות שתי שפות $L_1, L_2 \in \mathcal{NP}$

סעיף א'

הוכיחו כי: $L_1 \cup L_2 \in \mathcal{NP}$.

סעיף ב'

הוכיחו כי: $L_1 \cap L_2 \in \mathcal{NP}$.

סעיף ג'

הוכיחו כי: $L_1^* \in \mathcal{NP}$.

סעיף א'

הוכיחו כי: $L_1 \cup L_2 \in \mathcal{NP}$.

עלינו להראות כי איחוד כל שתי שפות ב-NP הוא שפה ב-NP. כלומר, עלינו להראות כי קיימים עד ואלגוריתם אימות פולונומיאלי המקבל את העד אמ"מ העד בשפה.

יהי $x \in L_1 \cup L_2$. נניח ש-$i \in \{1,2\}$ $x \in L_i$.

$L_i \in \mathcal{NP}$ לכן קיים עד y ואלגוריתם אימות A_i עבור x בשפה L_i.

אפשרות 1: העד עבור L1∪L2 יהיה העד <y,i>.

אלגוריתם האימות יהיה החזר את הפעלת העד y על A_i.

אפשרות 2: העד עבור L1∪L2 יהיה העד <y >.

אלגוריתם האימות יהיה:

הפעל את עד y על A_1 אם מקבל – החזר בשפה

אחרת, הפעל את עד y על A_2 אם מקבל – החזר בשפה

אחרת החזר לא בשפה.

בשני המקרים, הוכחת הנכונות טרוויאלית.

העד ואלגוריתם נכונות פולונמיאליים.

סעיף ב'

הוכיחו כי: $L_1 \cap L_2 \in \mathcal{NP}$.

פתרון – בבית לבד

סעיף ג'

הוכיחו כי: $L_1^* \in \mathcal{NP}$.

עלינו להוכיח כי עבור כל שפה $L \in NP$ מתקיים $L^* \subseteq NP$. נראה אלגוריתם אימות פולינומיאלי A_{L^*} עבור L^*. הרעיון הוא שעבור כל מילה $x = x_1x_2\ldots x_k \in L^*$, כאשר $x_i \in L$ ו- $1 \le i \le k$, נבנה עד שיהיה מורכב מ- k עדים, $y_1, y_2, \ldots, y_k$ עבור המילים $x_1, x_2, \ldots, x_k \in L$. כמו כן, נצטרך רשימת אינדקסים כדי להפריד את המילה $x_1x_2\ldots x_k$ למילים ב- L. בצורה פורמאלית, העד $y(x)$ עבור $x = x_1x_2\ldots x_k \in L^*$ מורכב משתי רשימות:

1. רשימת תעודות אימות $y_1, y_2, \ldots, y_k$, כאשר y_i הוא העד של x_i, $1 \le i \le k$.
2. רשימת אינדקסים $l_0, l_1, l_2, \ldots, l_{k-1}, l_k$, כאשר $l_j = \sum_{i=1}^{j} |x_i|$, עבור $1 \le j \le k-1$ ו- $l_0 = 0$, $j_k = |x|$.

אלגוריתם האימות A_{L^*} מקבל את המילה x ועד $y(x)$. האלגוריתם תחילה מחלק את המילה למילים $x_1, x_2, \ldots, x_k$ לפי רשימת האינדקסים, ז"א $x_i = x[l_{i-1}, l_i]$ ואז מאמת את כל המילים עם העדים המתאימים $< x_i, y_i >$ בעזרת אלגוריתם האימות A של L. אם התשובה לכל האימותים היא $TRUE$, אזי האלגוריתם מחזיר $TRUE$, אחרת $FALSE$.

גודל העד $y(x)$ הוא פולינומיאלי ב- $|x|$.
נראה כי קיים פולינום $P_{L^*}(x)$ כך ש- $|y(x)| \le P_{L^*}(x)$ עבור כל $x \in L^*$. נשים לב כי,

$$|y| = \sum_{i=1}^{k} |y_i| + (k+1)\log|x| + O(k),$$

כאשר $O(k)$ זאת תוספת עבור תחזוקת הרשימות (לדוגמא, תווי הפרדה). כיוון ש- $L \in NP$, קיים פולינום מונוטוני $P(x)$ כך שלכל $<x_i, y_i>$ מתקיים $|y_i| \le P(x_i) \le P(|x|)$. כמו כן, $k \le |x|$ ו- $|x_i| \le |x|$ לכל $1 \le i \le k$. נקבל:

$$|y| = \sum_{i=1}^{k} |y_i| + (k+1)\log|x| + O(k) \le kP(|x|) + (k+1)|x| + O(|x|) \le |x|P(|x|) + |x|^2 + O(|x|)$$

.

נגדיר, $P_{L^*} = 2|x|^2 + |x|P(x)$ ונקבל עבור כל $x \in L^*$, $|y(x)| \le P_{L^*}(x)$.

נוכיח את נכונותו של A_{L^*}:
אם $x = x_1x_2\dots x_k \in L^*$ אזי העד $y(x)$ עבורו יהיה כמוגדר מקודם. כיוון שאלגוריתם האימות ל- L יחזיר $TRUE$ על כל המופעים מהצורה $<x_i, y_i>$, אזי גם A_{L^*} יחזיר $TRUE$.
אם $x \notin L^*$, נניח בשלילה שקיים עד y (מוגדר כמקודם) כך ש- $A_{L^*}(x, y) = TRUE$. אזי קיימת חלוקה $x_1x_2\dots x_k$ של המילה x, כך שאלגוריתם האימות של- L מחזיר $TRUE$ על כל זוג $<x_i, y_i>$. ז"א לכל $1 \le i \le k$ מתקיים $x_i \in L$ ולכן $x = x_1x_2\dots x_k \in L^*$. סתירה.

זמן ריצה של A_{L^*}:
הפירוק של x למילים $x_1, x_2, \dots, x_k$ לפי האינדקסים לוקח $O(|x|)$. כיוון שהאלגוריתם A רץ בזמן פולינומיאלי על כל זוג $<x_i, y_i>$ ולכן גם פולינומיאלי ב- $|x| + |y|$, אזי גם המעבר על $k \le |x|$ זוגות הוא פולינומיאלי ב- $|x| + |y|$. ולכן סך זמן הריצה הכללי הוא פולינומיאלי ב- $|x| + |y|$.

רדוקציה פולינומיאלית

נאמר כי שפה A ניתנת לרדוקציה פולינומיאלית (בזמן) לשפה B ונסמן $A \leq_P B$ אם קיימת פונקצית העתקה f המקיימת:

1. $x \in A \Leftrightarrow f(x) \in B$.
2. קיים אלגוריתם פולינומיאלי המחשב את f.

הגדרה: מעגל המילטוני בגרף G הוא מעגל העובר בכל קודקוד בדיוק פעם אחת.

נתונות שתי בעיות:

HAMCYCLE (G): בהינתן גרף מכוון G,
האם קיים מעגל המילטוני ב- G?

HAMPATH (G,s,t): בהינתן גרף מכוון G ושני קודקודים s,t,
האם קיים מסלול המילטוני ב- G שמתחיל ב- s ומסתיים ב- t?

נראה רדוקציות פולינומיות בין שתי הבעיות HAMCYCLE ו- HAMPATH.

1) $HAMPATH \leq_p HAMCYCLE$

<u>הרדוקציה</u>: יהי $<G,s,t>$ המופע של $HAMPATH$. נוסיף ל- G קודקוד נוסף x, ושתי צלעות (x,s) ו-(t,x) ונקבל גרף חדש G'. המופע לבעיית $HAMCYCLE$ יהיה $<G'>$.

<u>פולינומיאליות הרדוקציה</u>: חישוב הרדוקציה (בניית G') לוקח זמן קבוע כיוון שהוספנו רק קודקוד ושתי צלעות.

<u>הסבר נכונות</u>: צ"ל: $<G,s,t>\in L' \Leftrightarrow <G'>\in L$, כלומר G מכיל מסלול המילטוני מ-s ל-t אם"ם G' מכיל מעגל המילטון. נניח $<G,s,t>\in L'$ אזי קיים מסלול המילטוני מ- s ל- t ב- $<G,s,t>$, שיסומן $(s,v_{i1},...,v_{in},t)$. ב- $<G'>$ קיים המעגל $(s,v_{i1},...,v_{in},t,x,s)$. מכאן $<G'>\in L$. בכוון השני נניח ש $<G'>\in L$, כלומר קיים מעגל המילטוני ב- $<G'>$. כיוון של-x יש רק צלע אחת שנכנסת אליו וצלע אחת שיוצאת ממנו המעגל הוא בה"כ מהצורה $(s,v_{i1},...,v_{in},t,x,s)$. מהמעגל ב-G' ניתן להוריד את x ואת הצלעות (x,s) ו-(t,x) ולקבל מסלול $(s,v_{i1},...,v_{in},t)$ שהוא מסלול המילטוני מ- s ל- t ב- $<G,s,t>$(כל הצלעות במסלול זה קיימות ב G כיוון שהן קיימות ב G'). מכאן $<G,s,t>\in L'$.

נא **לשים לב** כי פשוט לחבר את s ו-t בצלע לא עובד כאן. נסו למצוא דוגמה לכך.

$HAMCYCLE \leq_P HAMPATH$

<u>הרדוקציה</u>: עבור המופע $<G>$ נבחר קודקוד כלשהו x ב- G. נחליף את הקודקוד לשני קודקודים x_s ו- x_t (כולל שכפול כל הצלעות היוצאות ממנו והנכנסות אליו) ונקבל גרף G'. המופע לבעיית HAMPATH תהיה $<G',x_s,x_t>$.

<u>פולינומיאליות הרדוקציה</u>: חישוב הרדוקציה לוקח זמן החסום ע"י |3|V. מ-x יוצאות לכל היותר V-1 צלעות והחלפנו כל צלע כזו בשתי צלעות חדשות. בנוסף החלפנו את x בשני קודקודים חדשים. מאחר והקלט הוא בגודל לפחות |V| החישוב ניתן לביצוע בזמן פולינומיאלי באורך הקלט.

<u>הסבר נכונות</u>: צ"ל: $<G>\in L \Leftrightarrow <G',x_s,x_t>\in L'$ כלומר G מכיל מעגל המילטון אם"ם G' מכיל מסלול המילטוני מ- x_s ל- x_t.

נניח ש $L \in\, <G>$ כלומר שקיים מעגל המילטוני ב- $<G>$ שנסמן בה"כ $(x, v_{i1}, ..., v_{in}, x)$. נוציא ממנו את x ונחבר את הקצוות ל- x_s ו- x_t. כך נקבל מסלול המילטוני מ- x_s ל- x_t ב- $<G', x_s, x_t>$ $(x_s, v_{i1}, ..., v_{in}, x_t)$. מכאן $<G', s, t> \in L'$.

בכיוון השני, נניח ש- $<G', s, t> \in L'$ כלומר קיים מסלול המילטוני מ- x_s ל- x_t ב- $<G', x_s, x_t>$ $(x_s, v_{i1}, ..., v_{in}, x_t)$. קל לראות שניתן לתרגם אותו למעגל המילטוני ב- $<G>$ $(x, v_{i1}, ..., v_{in}, x)$ ע"י החלפת x_s ו- x_t ב x. מכאן $<G> \in L$.

שאלה

HAMP (G): בהינתן גרף מכוון G האם קיים מסלול המילטוני ב- G?

א. הראו רדוקציה פולינומיאלית בן HAMP ל- HAMPATH.

ב. הראו רדוקציה פולינומיאלית בן HAMCYCLE ל- HAMPATH.

סעיף א'

עבור מופע $G=(V,E)$ של בעיית HAMP נבנה מופע $G'=((V',E'),s,t)$ של בעיית HAMPATH:

$$E' = E \cup \{(s,v): v \in V\} \cup \{(v,t): v \in V\}$$

$$V' = V \cup \{s,t\}$$

כלומר הוספנו לגרף שני קדקדים s,t, כאשר מ-s יוצאות צלעות לכל קדקדי הגרף המקורי ול- t נכנסות צלעות מכל קדקדי הגרף המקורי.

הוכחה:

נתון $(G', s, t) \in HAMPATH$ כלומר קיים ב-G' מסלול המלטוני מ-s אל t. בה"כ נסמן את המסלול $(s, v_1, v_2, ... v_k, t)$ כשכל קודקוד מופיע פעם אחת בדיוק.

כיוון שהוספנו ל-G' רק צלעות ש-s או t הוא הקצה שלהן, הצלעות (v_i, v_{i+1}) עבור $1 \le i < k$ נמצאות גם ב-G. לכן $(v_1, v_2, ... v_k)$ מסלול המלטוני ב-G.

הוכחה לכיוון השני – תחשבו לבד.

סעיף ב'

עבור מופע $G=(V,E)$ של בעיית HAMCYCLE נבנה מופע $G'=(V',E')$ של בעיית HAMP.

עבור קדקוד שרירותי v נגדיר

$$E' = \{(E \setminus E_1) \cup E_2 \cup E_3\}$$

$$V' = \{(V \setminus v) \cup \{v', v''\}$$

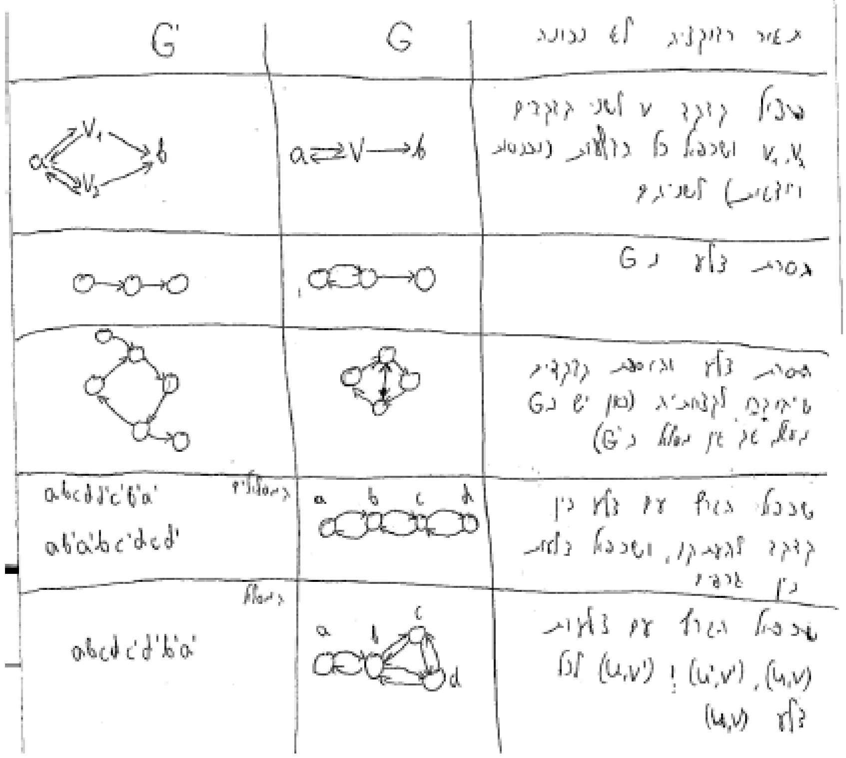
$$E_1 = \{(u,v),(v,u): u \in V\}$$

$$E_2 = \{(v',u): u \in V\}$$

$$E_3 = \{(u,v''): u \in V\}$$

כלומר החלפנו את v בשני קדקודים שלאחד הכנסנו את הצלעות שנכנסו ל-v ומהשני הוצאנו את הצלעות היוצאות ל-v.

בכל אחת מהדוגמאות הבאות ב- 'G יש מסלול המלטוני אבל ב- G אין מעגל המלטוני.



שאלה

הגדרה:

$AT\ MOST\ 3\ SAT$ =$\{\varphi$| φ היא נוסחת CNF ספיקה בה כל פסוקית מכילה <u>לכל היותר</u> שלושה ליטרלים }

הוכיחו כי $AT\ MOST\ 3\ SAT \leq_p 3\ SAT$.

<u>הרדוקציה:</u> בהינתן פסוק φ בנה פסוק φ' באופן הבאה.

1) כל פסוקית φ_{i} מהצורה $(x_n \lor x_m)$ החלף בשני פסוקיות הבאות

$(y_i \lor x_n \lor x_m) \land (\overline{y_1} \lor x_n \lor x_m)$.

2) כל פסוקית φ_{j} מהצורה (x_k) החלף ב 4 פסוקיות הבאות

$(y_{j1} \lor y_{j2} \lor x_k) \land (y_{j1} \lor \overline{y_{j2}} \lor x_k) \land (\overline{y_{j1}} \lor y_{j2} \lor x_k) \land (\overline{y_{j1}} \lor \overline{y_{j2}} \lor x_k)$.

3) פסוקיות עם שלושה משתנים השאר כמו שהן.

<u>הסבר נכונות:</u> קל לראות כי המשנים החדשים לא יכולים לספק את כל הפסוקיות החדשות שנוצרו, ולכן בעצם נצטרך לספק את הפסוקיות המקוריות.

<u>זמן חישוב הרדוקציה:</u> הרדוקציה מייצרת פסוק בעל לכל היותר פי-4 פסוקיות מהפסוק המקורי. ולכן פולינומיאלית.

משפט:
תהי L שפה כך ש- $L \in NP$. עבור שפה L', אם מתקיים $L' \leq_P L$ אזי $L' \in NP$.

<u>הוכחה</u>: עלינו בעצם להראות אלגוריתם אימות ועד.
תהי f פונקציית הרדוקציה מ- L' ל- L ויהי p הפולינום המונוטוני שחוסם את זמן ריצתו.
מפולינומיאליות הרדוקציה נקבל כי $|f(x)| \leq p(|x|)$. יהי V אלגוריתם אימות של L ויהי q הפולינום המונוטוני שחוסם את זמן ריצתו. כלומר על קלט (x, y), זמן הריצה של V הוא לכל היותר $q(x)$.
האלגוריתם הבא הינו אלגוריתם אימות עבור השפה L' בהינתן x ועד y.

- חשב את $f(x)$
- החזר את התשובה של $V(x, y)$.

אורך $f(x)$ הוא לכל היותר $p(|x|)$ וזמן הריצה של אלגוריתם האימות הוא לכל היותר $q(p(|x|))$, כלומר שניהם פולינומיאליים ב- $|x|$.
עבור $x \in L'$ מתקיים ש $f(x) \in L$ (מתכונות הרדוקציה), כמו כן מהעובדה ש $V(x, y)$ הוא אלגוריתם אימות, נקבל שקיים עד y, שעבורו $V(x, y)$ יחזיר true.
עבור $x \notin L'$ מתקיים ש $f(x) \notin L$ (מתכונות הרדוקציה), כמו כן מהעובדה ש $V(x, y)$ הוא אלגוריתם אימות, נקבל שלכל עד y, $V(x, y)$ יחזיר false.

תרגיל:

תזכורת: צביעה של גרף בלתי מכוון $G = (E,V)$ ב- k היא פונקציה $c: V \rightarrow \{0,1,\dots,k\text{-}1\}$ המקיימת $c(v) \neq c(u)$ לכל קשת $(u,v) \in E$. במילים אחרות קודקודים המחוברים על ידי קשת בגרף חייבים להיות צבועים בצבעים שונים.

סעיף א
נגדיר
4-coloring ={ G : קיימת ל- G צביעה ב- 4 צבעים }.

הוכיחו כי 4-coloring שייכת למחלקה $\mathcal{NP}$.

סעיף ב
הראו כי 4-coloring $\leq_p$ 3-coloring

פיתרון :

סעיף א'

העד- C צביעה חוקית. גודל העד פולנומי.

בהינתן 4 צביע של הגרף קל לבדוק בזמן פולינמיאלי שהיא אכן חוקית. לכל צלע (u,v) בודקים ש C(u)≠C(v) ובודקים שהצביעה היא ב-4 צבעים לכל היותר.

סעיף ב' :

הרדוקציה: בהינתן מופע של 3-coloring, הגרף $G=(V,E)$, פונקצית הרדוקציה f יוצרת גרף חדש $f(G)=G'=(V',E')$, ע"י הוספת קודקוד חדש המחובר לכל הקודקודים הישנים. באופן פורמאלי $G'=(V\cup v', E\cup E')$ כש-v הוא הקודקוד החדש ו- $E'=\{(u,v')| u\in E\}$.

זמן בנית הרדוקציה: יצירת קודקוד חדש ועוד יצירת |V| קשתות – סכ"ה פולנמאלי.

הוכחת נכונות:

כיוון ראשון: נראה כי לכל גרף G שלוש צביע הגרף 'G אחרי הבניה החדשה היא גרף ארבע צביעה. אם G שלוש צביע אזי קיימת צביעה C של G בשלוש צבעים. נצבע את כל הקודקודים V לפי הצביעה C ואת 'v בצבע הרביעי. כל $e\in E$, מחבר בין שני קודקודים בעלי צבע שונה לפי C וכל $e\in E'$ מחבר בין קודקוד בעל הצבע הרביעי לקודקוד בעל צבע מהשלושה הראשונים, לכן הצביעה היא צביעה חוקית של G ב-4 צבעים.

כיוון שני: נראה שאם הגרף 'G הנוצר על ידי הרדוקציה מגרף G הוא ארבע צביע אזי גרף G הוא 3 צביע.

תהי 'C צביעה של גרף 'G בעזרת ארבעה צבעים. נגדיר צביעה של קודקודי G ע"י $C(v)=C'(v)$, $v\in V$. ב.ה.כ. ניתן להניח $C'=(v')=3$. לכל $u\in V$ ישנה צלע (u,v') ב-'G, ולכן $C(u)\in\{0,1,2\}$.

כלומר, קבוצת הקודקודים V צבועים בשלושה צבעים. ובוודאי שלקודקודים שכנים יש צבעים שונים כי כל צלע (u,v) ב-G היא גם צלע ב-'G, ולכן $C'(v)=C(v)\neq C(u)=C'(u)$.

<u>רדוקציה נוספת לתרגול עצמי</u> :

בהינתן n קבוצות $S_1,\dots,S_n$, נאמר כי קבוצה $I\subseteq\{1,\dots,n\}$ היא אוסף של קבוצות נחתכות אם לכל $i,j\in I$ מתקיים $S_i\cap S_j\neq\varnothing$. כלומר, כל שתי קבוצות באוסף נחתכות.

נגדיר את השפה CNE

$CNE=\{S_1,\dots,S_n, k$: ב- $S_1,\dots,S_n$ קיים אוסף בגודל k של קבוצות נחתכות $\}$.

הראו כי $\text{Clique}\leq_p\text{CNE}$:

לפי סעיף קודם השפה ב-NP, נראה שהיא ב-NP-C על ידי Clique≤CNE

על קלט G,K פונקצית הרדוקציה, f, מתאימה לכל קודקוד u בגרף G=(V,E) הרדוקציה תבנה קבוצה

S_u. איברי הקבוצה S_u הן הצלעות המחוברות לקודקוד. פלט הרדוקציה הוא k ,$\{S_u\}_{u\in V}$.

זמן יצירת הרדוקציה: ליצור את הקבוצות עולה O(|V|), ליצור את איברי הקבוצה עולה O(|E|) והזמן פולינומיאלי.

הוכחת נכונות הרדוקציה:

<u>כיוון ראשון</u>: נניח בגרף G יש קליקה בגודל k, המורכבת מהקודקודים $W=\{u_1,\ldots,u_k\}$. במופע $f(G,k)$, נסתכל על אוסף הקבוצות $A=\{S_u : u\in W\}$. גודלו של A הינו k, ולכן נותר לבדוק שהחיתוך בין כל שתי קבוצות $S_u,S_v\in A$ לא ריק. לפי בנייה $u,v\in W$, כלומר $(u,v)\in E$, ולכן $(u,v)\in S_u$ וגם $(u,v)\in S_v$, ולכן $S_u\cap S_v\neq\varnothing$.

<u>כיוון שני</u>: נניח ש- $A=\{S_{u_1},S_{u_1},\ldots,S_{u_k}\}$ אוסף של k קבוצות לא זרות במופע $f(G,k)$. נראה כי קבוצת הקודקודים המתאימה $W=\{u_1,u_2,\ldots,u_k\}$ מהווה קליקה ב-G. אכן, לכל $u_i,u_j\in W$ מתקיים $S_{u_i}\cap S_{u_j}\neq\varnothing$. הקשת היחידה האפשרית בחיתוך קבוצות אלו היא הקשת (u_i,u_j), כלומר ישנה קשת בין כל זוג קודקודים מ-W, ולכן W קליקה ב-G.

הערה חשובה: בזמן חישוב הרדוקציה אנחנו בונים את הקבוצות $\{S_u\}$ מבלי לדעת אם $\{G,k\}\in$ Clique או לא. כנ"ל בזמן ניתוח זמן הריצה של הרדוקציה.

רק בהוכחת הנכונות של הרדוציה אנו מסתכלים בהוכחה על המקרה ש- $\{G,k\}\in$ Clique ולכן מסתכלים על קליקה בגודל k בגרף.

<u>רדוקציה עצמית</u>

כנאמר מקודם, ניתן לסווג בעיות כבעיות הכרעה, בעיות אופטימיזציה ובעיות מציאה. לעתים ניתן לפתור בעיית אופטימיזציה/מציאה ע"י בעיות הכרעה ולהפך.
לדוגמא, בעיית הקליקה בגרף לא מכוון G.

<u>ניסוח בעיית הכרעה</u> $CLIQUE(G,k)$: בהינתן גרף לא מכוון G ומספר טבעי k, האם קיימת קליקה בגודל k ב- G?
<u>ניסוח בעיית מציאה:</u> $FIND-CLIQUE(G,k)$: בהינתן גרף לא מכוון G, מצא קליקה בגודל k ב- G.
<u>ניסוח בעיית אופטימיזציה:</u> $MAX-CLIQUE(G)$: בהינתן גרף לא מכוון G, מצא את גודל הקליקה המקסימאלית ב- G.

ברור כי ניתן לפתור את $CLIQUE(G,k)$ ע"י $FIND-CLIQUE(G,k)$ או $MAX-CLIQUE(G)$.
נראה כיצד ניתן לפתור את $FIND-CLIQUE(G,k)$ או $MAX-CLIQUE(G)$ ע"י $CLIQUE(G,k)$.

1. בהינתן מופע של $<G=(V,E),k>$ של $FIND-CLIQUE(G,k)$:
   - אם $CLIQUE(G,k)$=FALSE, החזר "אין קליקה בגודל k".
   - $V' \leftarrow V$
   - עבור כל $v \in V'$:
     - אם $CLIQUE((V'\setminus\{v\},E),k)$=TRUE אזי:
       - $V' \leftarrow V'\setminus\{v\}$
   - החזר את V'.

<u>אבחנה 1 להוכחת נכונות</u>: עבור קודקוד $v \in V$ מתקיים $CLIQUE((V\setminus\{v\},E),k)$=TRUE אם"ם קיימת ב-G קליקה בגודל k ש- v **אינו חלק ממנה**.
<u>אבחנה 2 להוכחת נכונות</u>: עבור קודקוד $v \in V$ מתקיים $CLIQUE((V\setminus\{v\},E),k)$ =FALSE אם"ם v **חלק מכל הקליקות** בגודל k ב- G.

2. בהינתן מופע של $<G>$ של $MAX-CLIQUE(G)$ נמצא את הערך המקסימאלי של k עבורו $CLIQUE(G,k)$=TRUE.
   <u>אבחנה להוכחת נכונות</u>: אם אין קליקה בגודל k כלשהו, אזי אין קליקה בגודל l לכל $l>k$.