ממן 13 - אורי זית

1. א' -  *, משווים את האות הראשונה לאחרונה - n צעדים*

*את הלפני אחרונה לשניה n-1 .. סה"כ*  .

ב' - , מעתיקים את המילה לסרט השני (n) ומחזירים את אחד הראשים לתחילת הסרט (n)

*ואז משווים את הראשונה לאחרונה(1) את הלפני אחרונה לשניה (1) .. n פעמים סה"כ 3n = .*

*ג'* - , *מעבירים ראש אחד לסוף הקלט (n) ואז משווים ראשונה לאחרונה וכו' (n) סה"כ3n = .*

*2. א' - השפה המתקבלת ע"י B תהיה אינסופית אם"ם יש מסלול מהמצב ההתחלתי S1 אל מצב מקבל ובתוך המסלול יש מעגל, נוכל לבדוק האם יש מסלול כזה כך :*

*נסמן: n = מספר המצבים ב-B*

*1- נמצא בעזרת bfs את כל המצבים שניתן להגיע אליהם מ- S1*

*2- עבור n מצבים נבדוק שוב בעזרת bfs - 1.אם הוא ישיג מעצמו 2. אם יש מסלול ממנו אל מצב מקבל . סה"כ .*

*אם מצאנו מצב שמקיים את התנאים ב-2 אז יש מסלול עם מעגל מ-S1 אל מצב מקבל ולכן <B> בשפה אחרת <B>לא בשפה - הראנו אלגוריתם שמכריע בזמן פולינומיאלי ולכן השפה שייכת ל-P.*

*ב' - נסמן: n = מספר הצמתים ב-G*

*לכל זוג צמתים סדור (x,y) (יש n2 כאלה) נבדוק בעזרת bfs אם קיים מסלול מ-x ל-y אם קיים מסלול כזה לכל זוג אז <G> בשפה אחרת <G> לא בשפה.*

*הראנו אלגוריתם שמכריע בזמן פולינומיאלי ולכן השפה שייכת ל-P.*

*3. כן, הוכחה - עבור כל מילה w נוכל להכריע אם היא ב-B כך:*

*נריץ את המאמת על w עם כל מחרוזת שאורכה אינו גדול מההגבלה על זמן הריצה שלו(המאמת מעצם הגדרתו לא יכול לקרוא מילים שאורכן יותר מכך) ע"פ הגדרה 7.18 - קיימת מחרוזת שבעזרתה המאמת מכריע אם w בשפה , לכן אם המילה התקבלה באחת הריצות נקבל אחרת נדחה.*

4. נבנה מאמת V ש*יקבל כקלט כאשר n,m המספרים המוגדרים בהגדרת השפה, וc היא מחרוזת האימות – פירוק לגורמים ראשוניים, כך:*

= על קלט כאשר מספרים ו הוא פירוק לגורמים:

1. נוודא שכל הגורמים בפירוק הם אכן ראשוניים. אחרת, *נדחה*.
2. נוודא שמכפלת כל הגורמים בפירוק היא n. אחרת, *נדחה*.
3. נוודא שמספר הגורמים השונים בפירוק הוא איננו גדול מ-m. אחרת, *נדחה*.
4. אם הגענו לשלב זה, *נקבל*.

*נבחין שn וm שניהם מיוצגים ע"י מחרוזת בגודל ו- בהתאם.*

*(\*)מספר הגורמים הראשוניים של n הוא אינו גדול מ*, ולכן אורך הייצוג שלהם במחרוזת הוא כמובן לא גדול מ. לפיכך, גודל מחרוזת האישור היא פולינומיאלית בגודל הקלט.

מכיוון שקיים אלגוריתם שרץ בזמן פולינומיאלי  [מבחן AKS לראשוניות](https://he.wikipedia.org/wiki/%D7%9E%D7%91%D7%97%D7%9F_AKS_%D7%9C%D7%A8%D7%90%D7%A9%D7%95%D7%A0%D7%99%D7%95%D7%AA) להכרעת השאלה האם מספר הוא ראשוני אז יחד עם *(\*) נקבל ש*זמן הריצה של שלב 1 כולו *פולינומיאלי*.

*ברור ש-2 ו-3 גם הם* פולינומיאלים לכן V כולו רץ בזמן *פולינומיאלי ולכן מ7.19*

*השפה שייכת ל-.*

*5. הרדוקציה ב7.37 נעשית ממכונה לא דטרמניסטית כאשר אנו בונים לכל מחרוזת קלט נוסחה המחקה את כל תרחישי הפעולה האפשריים של המכונה על המילה, כלומר אם קיים תרחיש פעולה שהמכונה מקבלת את המילה אז קיימת הצבה המקבילה לתרחיש זה כך שהנוסחה ספיקה, מכיוון שהמכונה לא דטרמניסטית* ***ייתכנו כמה תרחישים אפשריים עבור כל מילה כך שהמילה תתקבל, ומספר ההשמות לנוסחה יהיה בהתאם לכך****.*

*6.נפעל בהתאם להדרכה בשאלה:*

*א' - נראה שהבעיה שייכת ל-NP:*

*נציג מאמת V שרץ בזמן פולינומיאלי: מחרוזת אימות - c - קידוד של k קבוצות מתוך הקלט , ברור שהיא* פולינומיאלית בגודל הקלט *כי c קטנה מגודל הקלט.*

*נסמן - m = מספר האיברים בכל הקבוצות שמוקדדות ב-c.*

*פעולת המאמת -*

*- נבדוק שיש k קבוצות בc ושכל קבוצה בc נמצאת בקבוצות המקודדות בקלט.(m2)*

*- עבור כל קבוצה x שהיא אחת מ-k הקבוצות ב-c נעבור על כל האיברים בה ונבדוק בכל שאר הקבוצות(בבדיקה על כל האיברים בה) אם איבר זה מופיע בהן, אם לא מצאנו ב-x איבר כזה שמופיע בכל הקבוצות אז נדחה, אם בכל הקבוצות קיים איבר כזה - נקבל.*

*המאמת רץ בזמן פולינומיאלי כי אנחנו רצים במקרה הגרוע עבור כל איבר בקידוד c על m-1 האיברים האחרים כלומר סה"כ m2*

*לכן מ7.19 השפה שייכת ל-.*

*ב' - נראה רדוקציה CLIQUE*

*קלט - <G,k>כאשר G=(V,E) , נסמן : כמות הצמתים=n*

*לכל צומת נייצר קבוצה si שאיבריה יהיו כל הקשתות מהצורה (vr,vi) (vi,vr) כאשר vr∈V*

*כלומר הפלט שנקבל אחרי הפעלת הפונקציה על <G,k> יהיה <S,k> כאשר*

*הפונקציה רצה בזמן לינארי של אורך הקלט .*

*בנוסף מתקיים :*

*קיימים k צמתים שבין כל 2 מהם נסמנם vi,vr יש קשת (vi,vr)*

*קיימות k קבוצות ב-S שלכל 2 מהם יש איבר משותף (vi,vr)*

*מסקנה XS היא NP-COMLETE*

*7.*

*א' -*

*נראה מאמת V :*

*קלט: <G,k>כאשר G=(V,E)*

*נסמן : כמות הצמתים=n , כמות הקשתות=m אז אורך הקלט הוא m\*n*

*כאשר E' קבוצת קשתות.*

*פעולה: 1. נבדוק ש|E'|=k וגם (m2+m)*

*2. נבנה G'=(V,E-E')( m\*n)*

*3. נבדוק בעזרת bfs לכל צומת ב G' אם הוא ישיג מעצמו ((m\*n)\*n)*

*אם 1. התקיים וב3. לכל הצמתים ראינו שאין מעגל לעצמם נקבל, אחרת נדחה.*

*קלט האימות c חלקי לקלט ובנוסף זמן הריצה של כל חלק במאמת הוא פולינומאלי*

*ולכן גם המאמת כולו - לכן מ7.19 השפה שייכת ל-.*

*ב' - הנקודה השגויה היא המעגלים שנוצרים בגרף* ***בעקבות משתנים משותפים*** *בין הפסוקיות השונות, כי אמנם לכל פסוקית יש 4 מעגלים שמייצגים אותה בגרף הרדוקציה(3: אחד לכל משתנה +1 :המעגל שקושר בין המשתנים ומתבטל בנתינת ערך אמת לאחד המשתנים) כך שיש הקבלה בין נתינת ערך אמת למשתנה מהפסוקית ולקטיעת המעגל, אך מלבד המעגלים "המייצגים" יש מעגלים נוספים שנוצרים, לדוגמה - עבור הפסוקית שבדוגמה ואותה פסוקית רק עם שלילה על כל משתנה (x,¬y,z)(¬x,y,¬z) יש השמות רבות שמספקות את 2 ההשמות(למשל x=1,y=1 ) אך בגרף המתקבל ע"י הרדוקציה יהיו 6 מעגלים בגודל 2 (שמורכבים מ-2 קשתות) ולכן אין 3 קשתות שאפשר להוריד כדי שלא יהיו מעגלים בגרף - למרות ש-n=3 .*

*ג' - הרדוקציה המוצעת :*

*- לכל משתנה x נגדיר צמתים x1,x2 בנוסף יהיו הצמתים x1-out ו- x2-out*

*- יהיה ביניהם המעגל x1->x1-out->x2->x2-out->x1,*

*ואז עבור כל פסוקית עבור משתנה x שמופיע בפסוקית תהיה קשת נכנסת ל-x1 וקשת יוצאת מx1-out- ועבור שלילתו אותו דבר רק עם x2*

*ונוסיף קשתות שיוצרות מעגל בין משתני הפסוקית מצומת out אל צומת שמייצג את המשתנה או שלילתו בפסוקית, אם בבניית הגרף במעבר על פסוקית מסוימת אין 2 משתנים שהופיעו בפסוקית אחרת ניתן להחליט על סדר כלשהו של הקשתות בין משתני הפסוקית , אחרת אם היו משתנים x,y שכבר הופיעו הקשתות בין המעגלים שמייצגים אותם בגרף נניח בסדר x1-out->y1*

*אז נשתמש בקשת הזאת ולא נוסיף קשת y1-out->x1 .*

