**ממן 15 – מבוא לתורת החישוביות והסיבוכיות – ברנדס איתי**

**שאלה 1:**

סעיף א':

כאשר נריץ את המכונה (הוכחת משפט 9.3, ע"מ 366 בספר הלימוד) על הקלט :

1. בשלב 1, יוגדר n בתור אורך .
2. בשלב 2 מוגדר חסם על כמות הזיכרון שהשלבים הבאים רשאים להשתמש בה –.
3. בשלב 3 נבדק האם הוא מהצורה כאשר היא מכונת טיורינג. התשובה היא שכן. לכן, המכונה ממשיכה לשלב הבא.
4. בשלב 4 מבצעים סימולציה של המכונה על עצמה. כאן למעשה מתחילה לולאה אינסופית, שכן במהלך הסימולציה נגיע לשלב 4, וגם בסימולציה שלה נגיע לשלב 4, וכך הלאה.
5. נבחין שבשלב 4 קיים לנו חסם על זמן הריצה (ע"י הגבלת מספר הצעדים), וגם חסם על המקום.

ולכן, משום שהגענו לולאה אינסופי, ומשום שמספר הצעדים אינסופי, ובכל פעולת סימולציה של המכונה עלינו להקצות כמות זיכרון נוספת, בשלב מסוים בהכרח נגיע לאחד החסמים (חסם הפעולות או חסם הזיכרון), והמכונה תדחה את הקלט.

סעיף ב':

כאשר נריץ את המכונה (הוכחת משפט 9.10, ע"מ 369 בספר הלימוד) על הקלט :

1. בשלב 1, יוגדר n בתור אורך .
2. בשלב 2 מוגדר חסם על כמות הצעדים שהשלבים הבאים רשאים להשתמש בה –.
3. בשלב 3 נבדק האם הוא מהצורה כאשר היא מכונת טיורינג. התשובה היא שכן. לכן, המכונה ממשיכה לשלב הבא.
4. בשלב 4 מבצעים סימולציה של המכונה על עצמה. כאן למעשה מתחילה לולאה אינסופית, שכן במהלך הסימולציה נגיע לשלב 4, וגם בסימולציה שלה נגיע לשלב 4, וכך הלאה.
5. בסופו של דבר, בשלב 4 קיים לנו חסם על זמן הריצה (ע"י הגבלת מספר הצעדים), ומשום שהגענו ללולאה אינסופית, בוודאי נגיע לחסם זה, והמכונה תדחה את הקלט.

**שאלה 2:**

לא ניתן להסיק טענה זו באמצעות מה שנלמד בסעיף 9.1.

בסעיף 9.1 משפט 9.6 (ע"מ 367 בספר הלימוד) למדנו ש (מוכל ממש). בפרט, אך .

אך עובדה זו לא סותרת קיום שפה המקיימת המקיימת וגם .

לכאורה, אם קיימת שפה כזאת, אז אפשר לעשות רדוקציה מכל שפה לשפת , שהיא ב, ולכן כל השפות ב הן גם ב, בסתירה להכלה ממש ש שהוכחה בעזרת משפט היררכיית המקום, ולכך ש אך , ולכן לא קיימת שפה כזאת.

אך זוהי לא באמת סתירה. נראה מדוע:

רדוקציות בין שפות היא בסיבוכיות **זמן** **פולינומיאלית**, ואילו רדוקציות בין שפות הן בעזרת סיבוכיות **מקום** **לוגריתמית**.

ז"א, אם קיימת שפה המקיימת המקיימת וגם , אז קיימת רדוקציה , אך זה לא סותר את , שכן לא בהכרח קיימת רדוקציה .

לפיכך, יתכן שקיימת שפה כזאת, ויתכן שלא. לא ניתן להסיק את התשובה מהחומר הנלמד בסעיף 9.1. יתרה מכך, ניתן להקביל את שאלת קיום שפה כזאת לשאלה , שהיא שאלה פתוחה במדעי המחשב.

**שאלה 3:**

נראה ש סגורה לרדוקציית מקום לוגריתמית, אך איננה סגורה, ולכן לא יתכן

ש.

*נבחין שמחלקת סיבוכיות היא סגורה לרדוקציית מקום לוגריתמית אם לכל רדוקציית מקום לוגריתמית המקיימת , אם אז גם .*

*נראה ש סגורה לרדוקציית מקום לוגריתמית:*

*יהי רדוקציית מקום לוגריתמית המקיימת , ו.*

*נבחין שהרדוקציה היא רדוקציית מקום פולינומיאלית, ולכן מההגדרה נמצאת ב ולפי היררכיית מחלקות הסיבוכיות הידועה לנו מתקיים , ולכן .*

*בנוסף לכך, נתון ש.*

*על קלט , נוכל להכריע את ע"י שימוש ברדוקציה על הקלט בעזרת מכונת טיורינג אי-דטרמיניסטית, ואז בהפעלת מכונת טיורינג אי-דטרמיניסטית המכריעה את על הקלט המתקבל מהרדוקציה.*

*לפיכך, להכרעת נשתמש ברדוקציה, בעלות פולינומיאלית לגודל הקלט, ואז נפעיל את מכונת הטיורינג האי-דטרמיניסטית המכריעה את על פלט הרדוקציה. כמות הריצה הכוללת היא כמות הריצה של הרכבת פולינום זמן הריצה של הרדוקציה על זמן הריצה של המכונה המכריעה את . אך פולינום הרכבת פולינום הוא פולינום, ולכן נקבל שניתן להכריע את בזמן ריצה פולינומיאלי על מכונת טיורינג אי-דטרמיניסטית, ולכן .*

***לפיכך, סגורה לרדוקציית מקום לוגריתמית.***

*נראה ש* ***לא*** *סגורה לרדוקציית מקום לוגריתמית:*

*לכל שפה נגדיר:*

*(השפה PAD והפונקציה מוגדרות בשאלה 9.21 בספר הלימוד, בעמוד 390. נניח שהתו # אינו מופיע בשפה ובמכונות המכריעות אותה. אם כן, נבחר בתו פנוי אחר כלשהו.)*

*נבחין שהפונקציה היא למעשה רדוקציית מקום לוגריתמית: בהינתן קלט באורך n עליה להוסיף סימני # בסוף הקלט. על מנת לעשות זאת, ניתן לשמור את כמונה, להדפיס תו #, ולהוריד את המונה ב-1, עד שנגיע ל-0. את מונה זה ניתן לשמור בזיכרון במקום של*

במילים אחרות, במקום לוגריתמי. לפיכך, היא אכן רדוקציה במקום לוגריתמי.

יהי מכונת טיורינג המכריעה את , ו מכונת טיורינג המכריעה את .

המכונה תבדוק על קלט :

1. שה*קלט הוא מהצורה כאשר s לא מכיל תווי #.*
2. *שתווי s נמצאים כולם באלפבית של .*
3. *ש*
4. *ולבסוף, ש, ע"י סימולציה של על הקלט s.*

יהי .

*נבחין ששלושת השלבים הראשונים, המכילים ספירות ובדיקות, רצים בסיבוכיות מקום של .*

*השלב הרביעי, שלב הסימולציה, רץ בסיבוכיות מקום של (נבחין שn מכיל את המחרוזת המרופדת, ולא את המחרוזת המקורית, שהיא באורך ).*

*לפיכך, מכונת הטיורינג רצה בסיבוכיות מקום לינארית, ולכן, .*

*לסיכום, לכל שפה ניתן להשתמש ברדוקציית המקום הלוגריתמית כדי לעבור לשפה .*

*נשתמש כעת במשפט היררכיית המקום (משפט 9.4 בעמוד 367 בספר הלימוד). לפי משפט זה, קיימת שפה המקיימת אך .*

*הרדוקציה pad מקיימת . אך , לפי משפט ההיררכיה. ולכן, אינה סגורה לרדוקציות מקום לוגריתמיות.*

נניח בשלילה ש. אך הראשונה סגורה לרדוקציות מקום לוגריתמיות, והשנייה איננה. סתירה. לכן .

**שאלה 4:**

נראה שיחס הקירוב 2 הוא הדוק ביחס לאלגוריתם המוצג בעמוד 394 בספר הלימוד.

לכל טבעי גדול מ-0, נגדיר את הגרף הדו-צדדי הבא:

ניצור  *צמתים: . כמו כן, לכל ניצור קשת . כמובן שקיימות קשתות.*

*בכך נקבל למעשה גרף דו-צדדי לא-מכוון בעל הצדדים ו .*

*אם נבחר לדוגמא את הקבוצה , נגלה שהיא כיסוי קודקודים בגודל , שכן כל קשתות הגרף מגיעות אל איברים ב. בנוסף, כיסוי קודקודים זה הוא מינימלי, כי הגרף הוא גרף דו-צדדי, ולכן כל קשת בו מגיעה מ ל והפוך, ולא קיים צומת ב שאיננה מחוברת לקשת (לפי צורת הבניה). לכן היא כיסוי קודקודים מינימלי בגודל .*

*עם זאת, נראה שהאלגוריתם ימצא כיסוי שגודלו :*

*בכל שלב באלגוריתם, נבחר קשת שהיא וצמתיה לא נבחרו לפני כן. משום שהגרף דו-צדדי, אחד מהצמתים נמצא ב והשני ב. נניח בלי הגבלת הכלליות ש ו (אחרת נחליף בין סימוני הצמתים). בכל שלב, שומרים את u ואת v אל תוך קבוצת כיסוי הקודקודים המיועדת, ומסמנים את הקשת . או, אפשר להגיד, מחסירים את הצמתים u,v ואת הקשת מקבוצת הצמתים והקשתות שנרוץ עליהם באיטרציות הבאות.*

*בכל איטרציה נחסיר שני צמתים וקשת אחת. פעולה זו תמשיך פעמים עד ש הקשתות יעלמו. ב איטרציות נבחרו צמתים לכיסוי הקודקודים. לפיכך, בסיום הריצה, האלגוריתם יחזיר כיסוי קודקודים בגודל , ז"א, בגודל הכפול לכיסוי הקודקודים המינימלי של הגרף.*

*לפיכך, לכל n טבעי גדול מ-0 קיים גרף עליו האלגוריתם מוצא כיסוי קודקודים שהוא גדול פי 2 בדיוק מכיסוי הקודקודים המינימלי.* מכאן נסיק שלא קיים כך שקירוב האלגוריתם הוא , שכן לכל מתקיים , ולכן, לא ניתן לטעון שקירוב האלגוריתם הוא , שכן קיים קירוב גדול ממנו.

*לפיכך, האלגוריתם הוא בעל יחס קירוב . בספר הלימוד הוכח שיחס הקירוב הוא . לכן יחס הקירוב הוא 2 (הדוק).*

**שאלה 5:**

סעיף א':

יהי גרף לא-מכוון עם מחירים חיוביים על הקשתות, המקיימות את אי-שיוויון המשולש.

יהי עץ פורש מינימלי של , ו מעגל המילטוני בגרף , ו המעגל ההימלטוני האופטימלי. כמו כן, נסמן:

* – סכום מחירי הקשתות ב-.
* – סכום מחירי הקשתות ב-.
* – סכום מחירי הקשתות ב-.

נבחין ש הוא עץ פורש מינימלי של , ולכן הוא משתמש במשקלי הקשתות הזולים **ביותר** על מנת לפרוש את צמתי הגרף (שכן הוא מינימלי). המעגל ההימלטוני האופטימלי גם הוא משתמש במשקלי הקשתות הזולים **ביותר** (שכן הוא אופטימלי), ולכן צורת הבניה של ו דומה, והקשתות הנבחרות בשתיהן זהות, מלבד כך שבמעגל ההימלטוני קיימת גם קשת מהסוף לההתחלה ע"מ לסגור מעגל.

מתקיים, אם כן:

לפי תנאי השאלה, קשת היא בעלת משקל **חיובי** כלשהו, ולכן .

בספר הלימוד הוכח ש, ולכן , ומכן ש, כנדרש.

סעיף ב':

*יהי גרף מלא ולא מכוון בעל מספר צמתים אי-זוגי גדול מ-5. נגדיר . נשתמש באותם סימונים מסעיף א'.*

*לכל מתקיים:*

נראה שבגרף זה מתקיים אי-שיוויון המשולש:

לכל שונים מתקיים .

לכל שונים מתקיים .

לפיכך, לכל שונים מתקיים .

נראה שהקירוב שהאלגוריתם משיג על גרף הוא :

נתון ש אי-זוגי וגדול מ-5. נפעיל את האלגוריתם על הגרף ונקבל בחזרה את המסלול ההמילטוני .

במצב הגרוע, הגרף הפורש שנבחר בתחילה עושה את כל הבחירות ה"לא נכונות":

נתחיל מאיזושהי צומת . נבחר לה בן אחד (שמאלי או ימני, לא משנה), שהוא צומת בעל זוגיות שונה ל, שאיננו צמוד ל. ולצומת נבחר בן נוסף, לפי אותו כלל בדיוק, עד שנגמרים הצמתים. נבחין שבדרך בניה זו, תמיד נוכל להוסיף צומת חדש כבן, עד שכמובן נוסיף את כל הצמתים לעץ (אפשר להוכיח זאת באינדוקציה על מספר הצמתים כאשר ואי זוגי). בכך העץ יהפוך לעץ פורש של . כמו כן, מובטח שצמתים צמודים לא יהיו בעלי קשת בעץ .

לאחר מכן, האלגוריתם ייצור ממנו מסלול הימלטוני כך שלכל צומת, 2 הצמתים הצמודים לו לא יהיו בעלי מספרים סידוריים עוקבים. במעגל קשתות, כאשר כל קשת מחברת בין 2 צמתים שמספרם הסידורי אינם צמוד. לכן, עלותם 2, חוץ מ2 הקשתות שנמצאות מסביב צומת מספר 1 – אשר עלותן היא 1 בדיוק.

לכן מתקיים:

כעת, קל לראות ש, שכן מעגל הימלטוני אופטימלי, ולכן הוא יכיל את הקשתות בלבד שעלותן 1, ויש כאלה, ואז קיימת עוד קשת אחת לסגירת המעגל שגם עלותה 1, ולכן - .

*בעיה זו היא בעיית מינימיזציה, ולכן לפי הדיון במדריך הלמידה (ע"מ 125) מתקיים:*

*כנדרש.*

*נסיק שהאלגוריתם הוא בעל יחס קירוב 2 הדוק:*

הראנו שלכל (אי-זוגי הגדול מ-5), קירוב האלגוריתם הוא . מכאן נסיק שלא קיים כך שקירוב האלגוריתם הוא , שכן לכל נוכל למצוא עבורו יתקיים , שכן שואף ל-2. ולכן, לא ניתן לטעון שקירוב האלגוריתם הוא , שכן קיים קירוב גדול ממנו.

לפיכך, בנוסף לכך שהוכח בספר שיחס הקירוב הוא קטן או שווה ל-2, נקבל שיחס הקירוב הוא בדיוק 2, והיחס הוא הדוק.

**שאלה 6:**

סעיף א':

יהי p מספר ראשוני. נראה בעזרת אינדוקציה שלכל a טבעי או 0, .

הנחת האינדוקציה: נניח שעבור טבעי כלשהו מתקיים *ונראה שמתקיים*

*.*

בסיס האינדוקציה: עבור זה טריוויאלי כי ו- .

*צעד האינדוקציה עבור : מתקיים*

*נבחין ש, ומשום שp מספר ראשוני, לא קיימים מספרים המחלקים אותו. לכן, אם נחלק את הביטוי בp, הp החבוי ב במונה יצטמטם, והשארית תהיה 0. לפיכך:*

עבור או מתקיים . ולכן:

*(\*) לפי הנחת האינדוקציה.*

*ובכך, צעד האינדוקציה הושלם, והטענה נכונה* לכל a טבעי או 0.

סעיף ב':

*נוכיח את המשפט הקטן של פרמה (משפט 10.6):*

יהי מספר ראשוני ו. נבחין שלפי חוקי מודולו מתקיים

ולכן:

*(\*) לפי הטענה שהוכחה באינדוקציה בסעיף א'.*

**שאלה 7:**

סעיף א':

*בספר הלימוד הוכח ש היא בעיית . אם , אז קיימת מכונה דטרמיניסטית המכריעה את בזמן פולינומיאלי.*

*להלן אלגוריתם למציאת הצבה מספקת לנוסחה בוליאנית :*

*A = " על קלט כאשר נוסחה בוליאנית:*

*יהי מספר הליטרלים בנוסחה.*

1. *נריץ את מכונת הטיורינג עם הקלט . אם המכונה תדחה,* נדחה*.*
2. *לכל ליטרל מ1 ועד n, נבצע:*
   1. *נציב .*
   2. *נריץ את מכונת הטיורינג עם הקלט .*
   3. *אם המכונה קיבלה, נשמור את ההצבה בצד.*
   4. *אחרת, נציב , נשמור את ההצבה בצד.*
3. *נחזיר את הצבות הליטרלים ששמרנו בצד, ו*נקבל.

"

***נכונות:***

*ראשית, בשלב 1 נבדוק האם הנוסחה ספיקה או לא. אם היא לא ספיקה, הרי שלא קיימת השמה מתאימה, ואנו דוחים.*

*בשלב 2 אנו כבר יודעים שהנוסחה ספיקה, ז"א, קיימת השמה כלשהי של ליטרלים שמספקת את הנוסחה. לכל ליטרל , אנו מנסים להציב 0, ולראות האם כעת הנוסחה עדיין ספיקה. אם כן, הרי שהצבת 0 בליטרל תוביל לרצף הצבות מתאים, ולכן נשמור את ההצבה בצד, ונמשיך בלולאה למציאת ערכי שאר הליטרלים. אם הצבת 0 מובילה לנוסחה שאיננה ספיקה, הרי שהצבת 1 תוביל בהכרח לנוסחה ספיקה, ואז נציב ונמשיך בלולאה למציאת ערכי שאר הליטרלים.*

*נסביר מדוע אם ההצבה נכשלת, אז ההצבה בהכרח מתקיימת:*

*נניח בשלילה ש2 ההצבות נכשלות, ושום הצבה של ליטרל לא מספקת את הנוסחה.*

*אם , אז מכונת הטיורינג בשלב 1 החזירה תשובה חיובית לגבי ספיקות , ולכן בהכרח קיימת הצבה עבור שמספקת את הנוסחה, בסתירה לאי-קיום הצבה מספקת ל.*

*אם , אז אנו כבר יודעים שהאיטרציות הקודמות בלולאה הציבו ערכים מספקים עבור , ולכן בהכרח קיים ערך כלשהו של שחייב לספק את הנוסחה, בסתירה לאי-קיום הצבה מספקת ל.*

*בסוף הריצה האלגוריתם מחזיר את רשימת ההצבות שנשמרו.*

***סיבוכיות זמן ריצה:***

*שלב 1 רץ בזמן ריצה פולינומיאלי לגודל הקלט, שכן זהו זמן הריצה של המכונה .*

*שלב 2.2 רץ בזמן פולינומיאלי (זמן ריצת המכונה ), ושאר תת השלבים בשלב 2 רצים בזמן קבוע.*

*סה"כ הלולאה בשלב 2 רצה במקרה הגרוע פעמים, ובכל פעם רצה בזמן פולינומיאלי. לכן סיבוכיות זמן הריצה היא פולינומיאלית כפול לינארית, שהיא כמובן פולינומיאלית (כפל פולינומים הוא פולינום).*

*לכן, האלגוריתם רץ בזמן פולינומיאלי לגודל הקלט, כנדרש.*

סעיף ב':

עלינו להראות שאם , אז .

לפי תרגיל 7.7 במדריך הלמידה (ע"מ 141), מתקיים . לכן נותר להראות ש.

נניח ש .

*בספר הלימוד הוכח ש היא בעיית . אם , אז, לפי ההגדרה, קיימת מכונת טיורינג הסתברותית המכריעה את בזמן פולינומיאלי עם שגיאה של לכל היותר. לפי למה 10.5 (ע"מ 397 בספר הלימוד) קיימת מכונת טיורינג הסתברותית מקבילה אשר מכריעה את בזמן פולינומיאלי עם שגיאה של לכל היותר, כאשר .*

*נגדיר אלגוריתם שיפעל בדיוק כמו אלגוריתם מסעיף א', אך עם השינויים הבאים:*

* *במקום מכונת הטיורינג המוגדרת בשלבים 1 ו2.2, נשתמש במכונה שהוגדרה כאן.*
* *בשלב 3, במקום להחזיר את כל ההשמות, נוודא שההשמות אכן מספקות את הנוסחה בצורה דטרמיניסטית, ע"י בדיקה האם . אם כן, המכונה* תקבל*. אחרת, היא* תדחה*.*

*אם איננה ספיקה, אז אנחנו תמיד דוחים, כי בשלב האחרון תמיד ישנה בדיקה דטרמיניסטית (בניגוד לבדיקה אי-דטרמיניסטית הסתברותית) לגבי האם ההשמה שנמצאה ספיקה. (יתכן שגם לפני שלב זה המכונה תדחה, אך השלב האחרון מבטיח שהאלגוריתם ידחה תמיד אפילו אם אחת ההרצות של טעתה).*

*אם ספיקה, נראה ש מקבלת בהסתברות של לפחות .*

*במהלך ריצת אנו קוראים למכונה בדיוק פעמים. ההסתברות שהמכונה טועה היא לכל היותר , כפי שכבר הראנו. אז הסבירות ש תדחה למרות ש ספיקה היא לכל היותר .*

*נבחין ש (שכן אורך המחרוזת גדול לפחות ב-1 ממספר הליטרלים בו, שכן יש בו סימני סוגריים וכמתים), ולכן:*

*(\*) עבור ים גדולים מספיק.*

*ולכן ההסתברות ש תטעה לגבי נוסחה ספיקה היא קטנה מ. במילים אחרות, ההסתברות ש תקבל מילים בשפה גדולה מ.*

סיבוכיות זמן ריצה:

האלגוריתם דומה לאלגוריתם מסעיף א'; החלפנו את מכונת הטיורינג שם באחת אחרת, אך שתיהן רצות בזמן פולינומיאלי. בנוסף, הוספנו בשלב 3 את הבדיקה  *, אך היא לוקחת זמן ריצה לינארי בגודל הקלט, ולכן איננה משנה את זמן הריצה הפולינומיאלי של האלגוריתם.*

*לפיכך, זמן הריצה של הוא פולינומיאלי בגודל הקלט.*

*ולכן, מכריעה את שפת בעזרת מכונת טיורינג הסתברותית, כך שעל מילים בשפה היא עושה זאת בהסתברות של , ועל מילים שלא בשפה היא מכריעה בהסתברות 1 (ודאי). ולכן, .*

*משום ש היא שפה -שלמה, קיימת רדוקציה פולינומיאלית מכל שפת אליה. הראנו שמתקיים , ולכן למעשה כל שפת ניתנת לרדוקציה פולינומיאלית לשפת , ולכן , כנדרש.*