**ממ"ן 12**

**שאלה 1**

תיאור האלגוריתם:נתון גרף  המתקבל מהגרף G ע"י הסרת אחת הקשתות בגרף. יהי T עץ פורש מינימלי של G. נבדוק תחילה האם הקשת e שהוסרה נמצאת בעץ T. אם לא, מתקיים  מאחר וכל קשתותT מופיעות ב-G' ומשקלן יוצר עץ מינימלי שכן אחרת היתה מתקבלת סתירה למינימליות של T.

אם e מופיעה בעץ T, נסיר אותה ונקבל שני רכיבי קשירות המכילים את כל צמתי הגרף. נריץ BFS על העץ  ונסווג כל צומת לאחד מרכיבי הקשירות לו שייכת. כעת נמצא קשת e' המחברת בין שני רכיבי הקשירות ועלותה מינימלית. נחבר את רכיבי הקשירות בעזרת e' ונקבל עץ פורש מינימלי של G'.

האלגוריתם:

1 if e is not edge in T then

1.2 return T

2 else

2.1 

2.2 run BFS on T and sign node connected component

2.3 

2.4 foreach edge  in 

2.4.1 if  and then

2.4.1.2

2.4.1.1

2.5 add min\_e to T' tree

2.6 return T'

נכונות:

טענה 1: הסרת קשת e מעץ פורש מינימליT תיצור שני רכיבי קשירות שכל אחד מהם הוא עץ פורש מינימלי לקבוצת הצמתים ברכיב.

הוכחה: הסרת קשת מעץ T בהכרח תיצור שני רכיבי קשירות שכן בעץ אין מעגלים. יהי w(T) משקל העץ המינימלי, w(A) ו-w(B) משקל עצי רכיבי הקשירות לאחר הסרת קשת e. משקל העץ T מקיים: 

נניח בשלילה שמשקל עצי הקשירות אינו מינימלי כלומר קיימת קבוצת קשתות המחברת את צמתי A עבורה מתקיים: 

מכאן מתקיים:  וזאת סתירה למינימליות של T.

טענה 2: הוספת קשת בעלות מינימלית בין שני עצים מינימלים תיצור עץ פורש מינימלי של אחוד הצמתים.

הוכחה: יהי S קבוצת צמתי אחד מרכיבי הקשירות ו- V-S קבוצת צמתי רכיב הקשירות השני. תהי  הקשת בעלת העלות המינימלית שקצה אחד שלה ב-S וקצה שני ב-V-S . ע"פ משפט 4.17 e' קיימת בכל עץ פורש מינימלי ולכן הוספתה לשני עצים מינימלים תיצור עץ פורש של אחוד הצמתים.

נכונות האלגוריתם נובעת משתי הטענות שהוכחנו לפיהם רכיבי הקשירות שנוצרו מהסרת קשת e בעץ T הם עצים מינימלים והוספת קשת בעלות מינימלית ביניהם תיצור עץ הפורש את הגרף כולו. כמו כן האלגוריתם מתבסס על נכונות BFS למציאת רכיבי הקשירות בגרף.

סיבוכיות: סיבוכיות בדיקה האם קשת e נמצאת בעץT היא כמעבר על קשתות העץ כלומר O(m). סיבוכיות הסרת הקשת מהעץ היא O(1). סיבוכיות הרצת BFS למציאת רכיבי הקשירות היא O(m+n). אנו מבצעיםBFS על עץ בעל שני רכיבי קשירות, לכן מספר הקשתות ב-T בהכרח קטן מ-n ולכן סיבוכיות הרצת BFS במקרה הזה היא: O(n+n)=O(n).

לבסוף אנו עוברים על כל קשתות G' ומחפשים קשת מינימלית המחברת בין רכיבי הקשירות. סיבוכיות בדיקת רכיב הקשירות של צומת היא O(1) ומעבר על כל הקשתות הוא O(m). מספר הקשתות בגרף קשיר מקיים  ולכן סיבוכיות האלגוריתם כולו היא O(m).

**שאלה 2**

תיאור האלגוריתם: נתון גרף לא מכוון קשיר G. תהי e הקשת שמשקלה שלילי . ניצור גרף חדש G' המכיל את כל צמתי וקשתות G מלבד הקשת e. נוסיף צומת חדשה ל-G', הצומת x, ונחברה לצמתי הקשת e. משקל הקשתות המחברות את x יהיה אפס ויתר המשקלים זהים. כעת נמצא באמצעות דייקסטרה מסלול s-t קצר ביותר בגרף G'. לאחר מכן נמצא את אורך המסלול הקצר ביותר בין הצמתים שעובר דרך הקשת e. מסלול זה שווה לסכום המסלולים s- x, x-t ומשקל הקשת e. נמצא את המסלולים באמצעות דייקסטרה החל מצומת x. המסלול הקצר מבין אלו שמצאנו יהיה המסלול המבוקש.

האלגוריתם:

1 create graph G' from G

2 

3 

4 

5 

6 if  then

6.1 return r1

7 else

7.1 returen r2

נכונות:

טענה: אורך המסלול s-t הקצר ביותר בגרף G העובר דרך הקשת השלילית e שווה לסכום:  כאשר r1, r2 הם המסלולים x-t, x-s הקצרים ביותר בגרף G'.

הוכחה: יהי P המסלול s-t הקצר ביותר בגרף G העובר דרך הקשת השלילית e.



כאשר . נסמן:





המסלולים שהגדרנו מקיימים את השוויון:



כעת נראה כי מתקיים: וגם 

כאשר r1,r2 הם מסלולים x-t, x-s הקצרים ביותר בגרף G'.

ע"פ הגדרת G' צומת x מחובר לצמתי הקשת e כלומר ,  ומשקל הקשתות שווה 0.

נניח בשלילה שהטענה אינה נכונה, כלומר: או .

בה"כ נניח כי מתקיים . נפריד למקרים:

1. כל צמתי המסלול r1 מלבד הצומת x נמצאים ב-G. לכן אם נוריד את הצומת x נקבל מסלול  ב-G. משקל הקשת  אפס לכן קיבלנו מסלול ב-G בעל משקל קטן מ-P1 וזאת בסתירה למינימליות של P1.
2. משקל הצומת x המופיע במסלול r1 הוא 0 לכן אם נבחר בצמתי P1 ונוסיף את הקשת  נקבל מסלול ב-G' הקצר מ-r1 בסתירה למינימליות שלו.

קיבלנו סתירה בשני המקרים ולכן הטענה בהכרח מתקיימת.

נכונות האלגוריתם נובעת מנכונות אלגוריתם דייקסטרה המוצא את המסלול s-t הקצר ביותר בגרף G' וכן את המסלולים x-s, x-t הקצרים ביותר. המסלול הקצר ביותר בין הצמתים עשוי להתקבל במעבר בקשתe מאחר וזה יפחית ממשקל המסלול הכללי. ע"פ הטענה שהוכחנו, מסלול s-t הקצר ביותר העובר דרך קשת e שווה לסכום המסלולים x-s, x-t ומשקל הקשת e. לכן המסלול הקצר ביותר בין הצמתים הוא המסלול קצר מבין אלו שמצאנו.

סיבוכיות: סיבוכיות ביצוע אלגוריתים דייקסטרה על הגרף G' היא O(mlgn).

יתר הפעולות של מציאת המסלולים בעץ וחישוב משקלם מתבצעות בסיבוכיות קטנה יותר ולכן סיבוכיות האלגוריתם כולו היא O(mlgn).

**שאלה 3**

תיאור האלגוריתם: נתון גרף לא מכוון . תחילה נהפוך את הגרף G לגרף מכוון  ע"י החלפת כל קשת לא מכוונות  בזוג הקשתות המכוונות  ו-. לכל קשת  נגדיר את פונקציית המשקל: 

כעת נריץ את אלגוריתם דייקסטרה על הגרף G' החל מצומת s. המסלול s-t שנקבל יהיה המסלול המבוקש.

האלגוריתם:

1 create direct graph G' from G

2 foreach edge  in G'

* 1. 

3 

4 return route(s,t)

נכונות:

טענה: סכום משקלי הקשתות במסלול s-t בגרף המכוון G' שווה לסכום הדרגות של הקדוקודים במסלול s-t ללא דרגת s בגרף הלא מכוון G.

הוכחה: יהי P מסלול בגרף המכוון :G' 

יהי  סכום הדרגות במסלול P ללא דרגת הצומת ההתחלתית s שתופיע בכל מסלול s-t.

ע"פ הגדרת הגרף G' לכל קשת  מתקיים: 

ולכן מתקיים: 

כלומר משקלי המסלולים זהים לפי שתי הפונקציות. ולכן הטענה מתקיימת.

נכונות האלגוריתם נובעת מנכונות אלגוריתם דייקסטרה המוצא את המסלול s-t הקצר ביותר בגרף G'. מנכונות הטענה שהוכחנו מסלול זה הוא גם בעל סכום הדרגות הקטן ביותר בגרף G. סכום הדרגות אינו מכיל את דרגת הצומת s אך אין זה משנה דבר מכוון שדרגת צומת s תופיע בכל סכום דרגות ולכן ניתן להתעלם ממנה.

סיבוכיות: סיבוכיות יצירת הגרף G' מהגרף הנתון G היא O(m+n) . לאחר מכן אנו מריצים דייקסטרה על הגרף G' בו 2m קשתות ולכן סיבוכיות הזמן היא: O(2mlgn)=O(mlgn).

מכאן שסיבוכיות האלגוריתם כולו היא O(mlgn).

**שאלה 4**

1. יהי  גרף קשיר ולא מכוון ויהי  תת גרף של G שמחזיר האלגוריתם. ע"פ הגדרת האלגוריתם לכל קשת  עבורה  מתקיים  שכן אחרת e היתה נוספת ל-H.

נניח בשלילה שהטענה לא נכונה כלומר קיים זוג צמתים  עבורם המסלול u-v הקצר ביותר ב-H גדול פי שלושה מאורכו של המסלול u-v הקצר ביותר ב-G כלומר: 

יהי P המסלול u-v הקצר ביותר ב-G: , 

 

אורך המסלול P הוא: 

המקרה הגרוע ביותר הוא כאשר המסלול u-v הקצר ביותר ב-H אינו מכיל אף קשת מ- P ואז מתקיים:



כלומר 

מצאנו כי לכל זוג צמתים  קיים מסלול u-v קצר יותר ב-H שאורכו לכל היותר פי שלושה מאורך המסלול u-v ב-G וזו סתירה למנימיליות שלו.

לכן הנחתנו שגויה כלומר הטענה מתקיימת לכל זוג צמתים בגרף.

1. יהי  תת הגרף של Gשמחזיר האלגוריתם.

טענה 1: כל מעגל בגרף H מכיל לפחות 5 קשתות.

הוכחה: נניח בשלילה שהטענה לא נכונה כלומר קיים מעגל C בגרף H המכיל פחות מ-5 קשתות. נניח כי C מכיל 4 קשתות (ההוכחה עבור 3 קשתות זהה).

בה"כ נניח כי  היא הקשת האחרונה במעגל שנוספה לגרף H. לפני הוספת הקשת e4 היה בגרף מסלול u-v שכלל את שלוש הקשתות: e1,e2,e3. בה"כ נניח כי . האלגוריתם הוסיף את e4 אחרונה לכן מתקיים:

1.  אחרת e4 היתה נוספת קודם ל-H.
2.  תנאי להוספת קשת כאשר קיים מסלול בין הצמתים שמקשרת.

נשים לב שמתקיים: 

וזה בסתירה ל-2, לכן הנחתנו שגויה כלומר לא קיים מעגל בגרףH בעל 4 קשתות. באופן דומה מוכיחים שלא קיים מעגל בעל 3 קשתות.

טענה 2: הדרגה המינימלית בגרף H בעל n צמתים היא לכל היותר .

הוכחה: נניח בשלילה שהטענה לא נכונה כלומר הדרגה המינימלית ב-H גדולה מ-. יהי  צומת בגרף H.

נסמן ב-S את קבוצת הצמתים השכנים ל-v. דרגת v גדולה מ-ולכן מספר השכנים של v גם הוא גדול מ- כלומר: 

ע"פ טענה 1 נסיק כי הצמתים בקבוצה S אינם מחוברים זה לזה שכן אחרת היה מתקבל מעגל באורך 3. נסמן ב-N(S) את קבוצת הצמתים השכנים שאינם v של כל צומת . דרגת כל צומת  גדולה גם היא מ- ולכן לכל צומת s יש לפחות  שכנים שאינם v. כל צומת שכן של  מחובר לצומת אחת בלבד מ-S שכן אחרת היה מתקבל מעגל באורך 4 בסתירה לטענה 1.

לכן מתקיים האי שוויון: 

וזאת בסתירה לכך שב-H יש n צמתים.

טענה 3: מספר הקשתות המקסימלי בגרף H בעל n צמתים הוא .

הוכחה: יהי f(n) מספר הקשתות המקסימלי בגרף H. ע"פ טענה 2 הדרגה המינימלית הגדולה ביותר שיכולה להיות בגרף H היא . אם נמחק את הצומת בעלת הדרגה הנמוכה ביותר בגרף נקבל גרף בעל n-1 צמתים וללא מעגלים קטנים מ-5 קשתות כמו גרף H. ולכן טענות 1 ו-2 מתקיימות גם עבור גרף זה.

לאחר מחיקת הצומת בעלת הדרגה המינימלית מתקיים:



נבצע n מחיקות ונקבל: 

ע"פ טענה 3 נקבל כי מספר הקשתות המקסימלי בגרף H הוא  ולכן מתקיים:



**שאלה 5**

יהי T עץ בינארי לחלוטין בעל n עלים ועומק d.

נגדיר d קבוצות עבור כל רמה בעץ:  

נגדיר פונקציית שכיחויות f באופן הבא:

לכל עלה בעץ יתקיים:  

לכל צומת אב בהכרח קיימים שני בנים שכן T בניארי לחלוטין .

כעת נוכיח כי עץ הופמן של סדרת השכיחויות Fהמוגדרת כפונקציית f שהגדרנו לכל צומת, הוא T.

תחילה נשים לב שסכום השכיחויות של n העלים הוא 1. נסתכל על זוגות העלים ברמה התחתונה בעץ . T בינארי לחלוטין, לכן מספר העלים בהכרח זוגי. כמו כן סכום השכיחויות של עלים אחים בעץ שווה לשכיחות עלה ברמה אחת מעל שכן:  והשכיחות של עלה ברמה  מוגדרת להיות .

כעת נבצע מיזוג של העלים בעץ כלומר נרשום את סכום השכיחויות של העלים בצומת האב ונמחק את העלים. כך יהפוך צומת האב לעלה ברמה אחת מעל בעל שכיחות של שני בניו שמוזגו. נמשיך ונבצע מיזוג עבור כל הרמות בעץ, וכאשר נגיע לרמה d=1 נקבל  כלומר שכיחותו של השורש בעץ שווה 1 וערך זה כאמור שווה לסכום כל בניו העלים.

בכל איטרציה של אלגוריתם הופמן ממזגים את זוג האיברים בעלי השכיחות הנמוכה ביותר. ע"פ פונקציית השכיחויות שהגדרנו הזוג שיבחר ראשון יהיה זה שבקבוצה העמוקה ביותר, כלומר זוג העלים הנמוך ביותר בעץ T המקורי.

לאחר המיזוג נוסף איבר לקבוצה בעל שכיחות השווה לסכום בניו. כפי שראינו קודם, שכיחות צומת האב (הצומת שנוסף) שווה לשכיחות העלים ברמה אחת מעל לזה שמוזגה. לכן בסיום מיזוג האיברים שהופיעו ברמה האלגוריתם ימזג את האיברים (העלים והאבות שנוספו) ברמה . בצורה זאת כל עלה שהופיע ברמה i בעץ המקורי יופיע באותה רמה גם בעץ הנוצר ע"י אלגוריתם הופמן ולכן נקבל עץ T זהה לחלוטין.