ממ"ן 11 אלגוריתמים גיל וסר 205519739

אלגוריתמים שאלה 1

על מנת להוכיח את שתמיד ניתן למצוא לוחות זמנים מקוצצים נשתמש באלגוריתם הזהה במהותו לאלגוריתם הזיווג היציב. הפעם הזיווג הוא בין הנמלים לספינות והזיווג ביניהם שקול לזיווג בין זוגות נשואים.

רשימת העדפות של הספינות:

ספינה רוצה לסיים את עבודה שלה כמה שיותר מהר ולהיכנס לטיפול מוקדם ככל האפשר לכן ספינה תעדיף נמל שהיא צריכה לעבור בו מוקדם יותר מנמל שהיא אמורה לעבור בו מאוחר יותר.

רשימת העדפות של נמלים:

נמל רוצה להישאר פתוח כמה שיותר זמן על מנת שיוכל לקלוט כמה שיותר ספינות ,

לכן נמל יעדיף שתעגון אצלו ספינה העוברת מאוחר יותק מספינה שעוברת בו מוקדם יותר.

תיאור האלגוריתם:

נסמל

1. נבנה רשימת העדפות לכל ספינה בכך שנסרוק את כל הימים בלוח הזמנים מ 1 עד M.

1. נבנה רשימת העדפות לכל הנמלים כך:
   1. נעבור על כל 1 עד M
      1. בכל יום iנעבור על כל ספינה s
      2. אם לספינה s יש נמל d בו היא אמורה לעבור ביום i .
      3. נוסיף את הספינה sלתחילת רשימת העדפות של הנמל p
2. בתחילת האלגוריתם כל הספינות והנמלים לא מצוותים.
3. כל עוד קיימת שלא מצוותת וגם לא שלחה בקשה לעגון בכל :
   1. נבחר s כזו
   2. s זו תשלח בקשה לנמל הנמצא בראש רשימת העדפות , שלא שלחה לה בקשה בעבר נסמנו d.
      1. אם d לא מצוות הוא יסכים להצעה וכעת s,d מצוותים
      2. אם d מצוות אך s גבוהה יותר מ s' ברשימת העדפות של d , הציוות s',d יתפרק ו s,d יצוותו זה לזה
      3. אם d מצוות אך s נמוך או שווה ל s' ברשימת העדפות של d נמשיך.
4. בסיוף הלולאה כל הציוותים(אירוסין) יהפכו להסכמי עגינה(נישואין) .

נכונות האלגוריתם:

נראה כי מתקיימים כל הדרישות קבוצת לוחות זמנים מקוצצת .

ראשית כל ברשימת העדפות של כל ספינה מופיעים כל הנמלים, זה מתקיים מכיוון שבכל לוח זמנים יש את כל הנמלים.

שנית כל ברשימת העדפות של כל נמל מופיעות כל הספינות, זה מתקיים מכיוון שעברנו על כל יום והכנסנו לכל רשימה של כל נמל את הספינות המתוכננות לאותו יום לכן וברשימות הזמנים כל הספינות עוברות בכל הנמלים לכן הטענה מתקיימת.

ולבסוף נוכיח כי לא ייתכן ש2 ספינו יהיו יחדיו באותה ספינה .

כלומר שאם ל s,d , יש הסכם עגינה ו s מגיע ל d ביום x לd לא יתכן כי לאחר יום d תגיע s' כלשהי.

נניח כי ל s,d יש הסכם עגינה ו s מגיעה ל d ביום x . וגם s' מגיעה ל d ביום y כך ש y>x מכך ש s' עוברת ב d נובע כי היא הציעה לה הסכם עגינה , אך מכך של y>x נובע כי d הייתה מוותרת על הסכם העגינה עם s ומקבלת את הסכם העגינה עם s' ובכך הגענו לסתירה .כלומר לא יכול להיות מצב בו 2 ספינות נמצאות באותו נמל באותו יום.

זמן הריצה:

לחישוב כל רשימת העדפות של ספינה זמן הריצה הוא O(m)

ישנן n ספינות לכן לחשוב כל רשימות העדפות של כל הספינות זמן הריצה הוא O(n\*m)

כמו כן בחישוב רשימות העדפות של כל הספינות אנחנו עוברים בכל יום על כל הספינות כלומר זמן הריצה הוא שוב O(n\*m)

בזמן הציוות בכל ריצה של הלולאה נשלחת בקשת עגינה אחת שלא תחזור שוב כי בכל פעם שולחים בקשה שלא נשלחה מעולם, לכל ספינה יש n נמלים לשלוח אליהן בקשות יש n ספינות ולכן הלולאה רצה לכל היותר פעמים כלומר זמן הריצה של האלגוריתם הוא

שאלה 2

במידה ואין מעגל ברכיב קשירות כלשהו בגרף אז לרכיב קשירות זה בן n קדקודים יש לכל היות n-1 קשתות ולכן קיים לפחות קדקוד אחד שדרגת היציאה שלו היא 0 .

במידה ובכל רכיבי הקשירות יש מעגלים יש אפשרות לכוון את הגרף

אלגוריתם

לכל רכיב קשירות G נבצע סריקת BFS מקדקוד כשלהו ונקבל בחזרה עץ T

אם כל צלעות G נמצאות ב T נחזיר FALSE

אם קיימת קשת ב G שלא נמצאת ב T הרי זוהי קשת חוזרת נסמנה {u , v }

נכוון את הקשת מ u ל v

רקורסיבית לכל צומת החל מ v נכוון את כל הצלעות הלא מכוונון כלפי חוץ

עבור כל צומת שאנו מכוונים נכניס אותה למערך

אם עברנו על כל רכבי הקשירות בלי להחזיר FALSE נדפיס מערך הקשתות

נכונות:

אם כל הצלעות של G נמצאות ב F הרי שאין קשתות הסוגרות מעגל ברכיב הקשירות כי לא הגענו במהלך הסריקה לאותו קדקוד ולכן יש במעגל לכל היותר n-1 קשתות וכן מתחייב שיש לפחות קדקוד אחד שדרגת הכניסה שלו היא 0.

אם קיימת קשת שלא נמצאת ב F הרי שיש קשת המחברת בין קדקוד כלשהו לקדקוד שכבר נסרק ולא תורם לגילוי קדקוד חדש כלשהו . כשאר נבצע את כיוון הצלעות באמצעות BFS נעבור בכל צומת (נובע מנכונות BFS ) ואנחנו מכוונים אליו קשת . כמו ן לצומת v השורש כיוונו קשת אליו ולכן נובע כי לכל קדקוד יש לפחות קשת שנכנסת אליו .ולכן דרגת הכניסה של כל צומת היא לפחות 1.

סיבוכיות:

אנחנו מבצעים לכל רכיב קשירות פעמיים BFS 2\*O(n)=O(n) וכמו כן אנו עוברים על כל הקשתות בG לראות שהן נמצאות בT O(n) וגם מכוונים קשת יחידה מחוץ לBFS O(1) כלומר סיבוכיות זמן הריצה הינה O(n)

שאלה 3

רעיון האלגוריתם : להחליף את כל הפסוקים מהצורה לפסוקים מהצורה

ו .

להרכיב גרף שקדקודיו הם כל הליטרליים ושלילותיהם . כלומר לכל ליטרל v יש קדקוד v וגם קדקוד .

אלגוריתם:

נבנה את הגרף המכוון כך :

לכל ליטרל v נבנה 2 קדקודים v ו .

עבור כל פסוק מהצורה נבנה את הקשת ואת הקשת .

נבחר קדקוד v שעוד לא סומן ונבצע עליו DFS נקבל חזקה עץ .

אם לא נמצא בעץ נסמן כל קדקוד בעץ ב T וכל קדקוד שהוא שלילתו של הקדקוד ב F .

נכניס למערך S את הערכים.

אחרת אם נמצא בעץ נבצע על DFS ונקבל עץ נוסף בחזרה

אם v לא חלק מהעץ החדש נסמן כל קדקוד בעץ ב T וכל קדקוד שהוא שלילתו של הקדקוד ב F . ונכניס למערך S את הערכים

אחרת נחזיר FALSE

נחזיר את S .

בכל פעם אנחנו בודקים האם קיים מסלול מ v ל אם יש מסלול כזה הרי קיבלנו סתירה שהרי אם v נכון מחייב שלאורך כל המסלול גם שאר הקדקודים ברכיב הקשירות הם נכונים .

אך v לא בהכרח נכון ולכן אנחנו גם בודקים את המצב השני בו נכון ועושים DFS ממנו שוב אם v נמצא ברכיב הקשירות הרי שינה סתירה v→ v'→ vולכן נחזיר FALSE אם אין סתירה סימן שלא קיבלנו סתירה בין הפסוקים ולכן נוכל לסמן את רכיב הקשירות כנדרש .

נוכיח זאת : נניח שעשינו DFS מ v אבל במסלול קיימים u ו u' שלילתו כך ש u→u' אז כדי ששניהם יהיו יחד בעץ v חייב להתחבר ל u תחילה ולאחר מכן מכיוון ש v→ j → v = u'→ j' → v' נובע כי בעץ גם ימצא v' כלומר אם v' לא נמצא בעץ אין שום סתירה .

לאחר שעשינו זאת על כל רכבי הקשירות נקבל פסוק אמת.

סיבוכיות :

עבור כל צומת אנחנו מבצעים פעמיים DFS לכל היותר ואם צריך אנחנו מסמנים את כל צמתי העץ ושלילתם כלומר O(n) לכל היותר עבור כל DFS. אלא שבכל איטרציה אנחנו מסמנים את כל צמתי העץ וכל צומת שנסמן לא נבצע עליה יותר פעולות כי נסמנה פעם אחת בלבד כלומר סה"כ הפעולות שאנחנו עושים הוא O(n) .

שאלה 4

לגרף G נבצע רדוקציה : ניצור את G' כך:

נמחק כל צלע שכוונה את קדקוד מועדף .

נכפיל את הגרף ב 3 כלומר שיהיו לנו 1G 2G 3G העתקים של G' .

עבור כל צלע שמחקנו נבנה 2 צלעות נסמן את קדקוד היציאה שלה ב G v ואת קדקוד הכניסה שלה u ניצור צלעות דומות (v1,u2) בהתאם למספרי העתקים .

נבצע BFS מ s1 ל t3 ונחזיר את המסלול המתקבל.

קיימת התאמה חד ערכית בין מסלול כלשהו המכיל 2 קדקודים מועדפים לבין מסלול מהגרף G' .

כל קשת רגילה (לא לקדקוד מועדף קיימת בגרף הנוכחי שלה ) כל קשת לקדקוד מועדף נעבור בקשת הדומה (כמו בבניית הרדוקציה ) לגרף הגבוהה יותר .

כמו כן כל מ s1 ל t3 מכיל בדיוק 2 קדקודים מועדפים.

כל מסלול יכול לעבור לכל היותר פעמיים בקדקוד מועדף . מפני שהקשתות הנכנסות לקדקודים המועדפים עוברים רק מגרף נמוך לגרף גבוה וישנם רק 2 מעברים בין גרף נמוך לגרף גבוהה הטענה נכונה.

כמו כן כל מסלול בין s1 ל t3 מכיל לפחות 2 קדקודים מועדפים כי בכל מעבר מגרף נמוך לגרף גבוהה חייבים לעבור דרך קדקוד מועדף מפני שהחיבור בין הגרפים נעשה רק מקדקוד מסוים אל קדקוד מועדף כמו כן יש 2 מעברים כאלה לכן הטענה נכונה.

זמן ריצה בניית הרדוקציה O(n + m) + מעבר BFS O(n + m) סה"כ0 O(n + m)