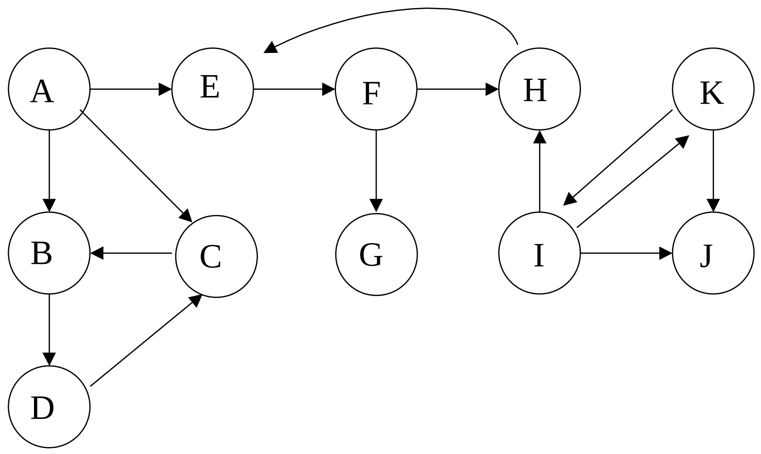
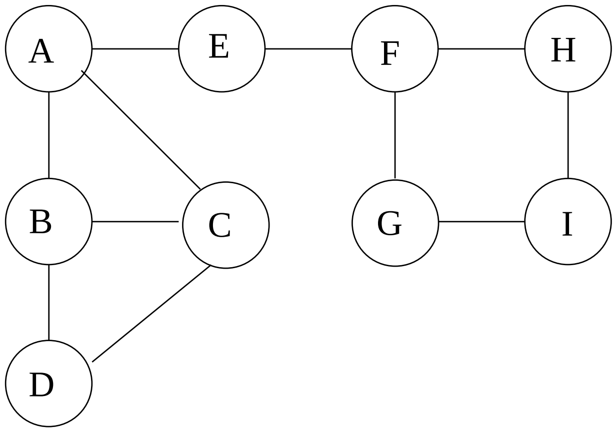
נושא: חיפוש עומק- DFS

שאלה 1
נתונים הגרף המכוון והגרף הלא-מכוון הבאים. הניחו שהקדקודים נתונים לפי סדר א״ב.





א. הפעילו DFS על הגרפים האלה. לכל גרף עליכם: לצייר את תוצאת ה-DFS (עצים + קשתות אחרות תוך ציון סוגיהן); לרשום את סדר גילוי הקדקודים ואת סדר הסיום.

ב. הפעילו את האלגוריתם למציאת רכיבי קשירות חזקה על הגרף המכוון. ציירו את התוצאות של שני מעברי ה-DFS, ובסוף רשמו את רכיבי הקשירות שנמצאו (אפשר ע״י הקפתם בציור ה-DFS עצמו).

ג. הפעילו את האלגוריתם למציאת גשרים על הגרף הלא-מכוון. ציירו את התוצאות של שני מעברי ה-DFS, ובסוף סמנו את הגשרים שנמצאו (אפשר לסמן בציור ה-DFS עצמו).

פתרון:

סעיף א', תשובה חלקית *(לבדיקה עצמית)*

בגרף המכוון, סדר הגילוי הוא (משמאל לימין):

A B D C E F G H I J K

וסדר הסיום:

C D B G H F E A J K I

בגרף הלא-מכוון, סדר הגילוי הוא (משמאל לימין):

A B C D E F G I H

וסדר הסיום:

D C B H I G F E A

***סעיף ב', תשובה חלקית** (לבדיקה עצמית)*

ה-DFS הראשון בוצע כבר בסעיף א'. רכיבי הקשירות החזקה הם (בסדר מציאתם באלגוריתם):

{I, K} {J} {A} {E, H, F} {G} {B, C, D}

***סעיף ג', תשובה חלקית** (לבדיקה עצמית)*

ה-DFS הראשון בוצע כבר בסעיף א'. הגשרים הם: AE, EF.

שאלה 2

רשמו פסאודו-קוד של אלגוריתם DFS המשיג את המטרות הבאות: בסוף הריצה, לכל קדקוד v נמצא בתא D[v] את המספר הסידורי של v לפי סדר הגילוי ("מספר DFS") ובתא F[v] את המספר הסידורי של v לפי סדר הסיום ("מספר סיום"). כמו-כן, על האלגוריתם להציב בתא Parent[v] את זהות הקדקוד שהינו אביו של v בעץ ה-DFS. לקדקודים שהם שורשים יכיל תא זה את המספר של עצמם.

פתרון:

(התוספות מודגשות בצהוב)

DFS(Graph G)
// INIT
time ← 0
for each vertex u **do**
    Color[u] ← white

// MAIN LOOP
for each vertex u **do**
    if Color[u] = white **then**
        Parent[u] ← u
        VISIT(u)

VISIT(Vertex u)
time ← time + 1
D[v] ← time
Color[u] ← gray   // begin processing of u
for each v ∈Adj[u] **do**
    if Color[v] = white **then**
        Parent[v] ← u
        VISIT(v)
Color[u] ← black   // end processing of u
time ← time + 1
F[v] ← time

<u>שאלה 3</u>

א. יהי G גרף לא מכוון, ותהי K קליקה ב- G (כלומר קבוצה חלקית של קדקודים של G כך שכל שני קדקודים ב- K שכנים). נבצע DFS על G. הוכיחו או הפריכו את הטענות הבאות:
<u>טענה 1 -</u> כל הקדקודים ב- K יהיו בהכרח על מסלול מכוון אחד בעץ ה- DFS המתקבל.
<u>טענה 2 -</u> כל קדקודי הקליקה יופיעו בהכרח <u>ברציפות</u> על מסלול כנ"ל.

ב. כעת, יהי G גרף מכוון, ויהי C מעגל מכוון פשוט ב- G. הוכיחו או הפריכו את הטענות הבאות:
<u>טענה 1 -</u> כל קדקודי המעגל יופיעו בהכרח על מסלול מכוון אחד בעץ ה- DFS המתקבל.
<u>טענה 2 -</u> כל קדקודי המעגל יופיעו בהכרח <u>ברציפות</u> על מסלול כנ"ל.

<u>פתרון:</u>

<u>סעיף א</u>

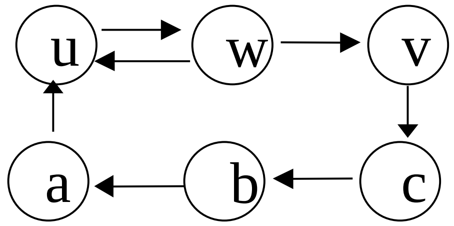
(1) הטענה נכונה. <u>הוכחה</u>: ב-DFS של גרף לא מכוון יש רק קשתות עץ וקשתות חוזרות. אם נניח בשלילה ששני קדקודים מתוך K: u,v שייכים לשני תת עץ שונים (כך שאף אחד מהם אינו אב קדמון של השני) אזי הקשת ביניהם תהיה קשת חוצה – סתירה.
יש כמובן דרכים אחרות להוכחה, למשל באינדוקציה על הגודל של K.

*(2) הטענה שגויה. ייתכן שהקדקודים לא יופיעו ברצף. זה יקרה אם ה- DFS יבקר בקדקוד שלא שייך ל- K באמצע (**תשובה מלאה צריכה לכלול דוגמה. מבחינה לוגית, הפרכת טענה כזו היא <u>אך ורק</u> על ידי דוגמה נגדית קונקרטית, כלומר גרף והדגמת הריצה עליו**).*

<u>הערה</u>: למרות ש-(1) נכון, עדיין ניתנו לו לעיתים הוכחות שגויות. אחד הסימנים המובהקים להוכחה שגויה של (1) הוא – שבאותה דרך אפשר "להוכיח" גם את (2)...

<u>חלק ב</u>

(1)+(2) הטענות שגויות והנה דוגמה נגדית. נתבונן בגרף הבא ונניח ש- w הוא הקדקוד שמתגלה ראשון, ותחילה עוברים על הצלע (w,u). מכאן, u,v יהיו צאצאים של w בעץ ה- DFS ויהיו אחים זה של זה בעץ. כלומר המעגל (המכיל את כל קדקודי הגרף) אינו מקיים את הטענה.



<u>שאלה 5</u>

יהי G גרף לא מכוון פשוט בעל n קדקודים.

א. כתבו אלגוריתם המוצא בו מעגל פשוט (או מודיע שאין כזה) בזמן O(n) (אם יש מעגל אז הפלט יהיה רשימת הקדקודים המשתתפים במעגל). הניחו שהגרף נתון במבנה של רשימות סמיכויות.

ב. אם הגרף נתון כרשימת קדקודים ורשימת צלעות (בלי סדר מסוים), האם ניתן עדיין להשיג זמן של O(n)?

<u>פתרון:</u>

<u>סעיף א'</u>

<u>תאור הרעיון של האלגוריתם</u>

האלגוריתם הוא DFS עם השינויים הבאים:
אם מוצאים קשת חוזרת עוצרים מיד את תהליך ה-DFS ומוצאים מעגל כפי שיתואר להלן.
אחרת – מודיעים שאין מעגל בגרף.

איתור קשת חוזרת הוא קל, כי כל קשת שאינה קשת עץ היא כזו. לכן אין צורך לשם כך בשום תוספת מיוחדת לאלגוריתם.
למציאת המעגל עצמו נחזיק באלגוריתם את המערך Parent[n] שבו Parent[v] יהיה שם הקדקוד שהוא ההורה של v בעץ ה- DFS (את עדכון המערך הוספתם לאלגוריתם הבסיסי כבר בשאלה 2). ברגע שנזהה קשת חוזרת (u,v), נקרא לפונקציה PrintCycle שתדפיס את המעגל בעזרת המערך Parent (הפונקציה "תטייל" במעלה העץ תוך הדפסת הקדקודים על המסלול, עד אשר תחזור שוב לקדקוד v).

האלגוריתם מסתמך על הטענות הבאות:
<u>טענה 1</u>
לאחר הרצת DFS על G, בכל מעגל תהיה קשת חוזרת.
<u>הוכחה</u>
יהי v הקדקוד האחרון, מקדקודי המעגל, שהתגלה ב-DFS. יהיו u_1, u_2 שכניו משני הצדדים במעגל. אחד מהם יכול להיות אביו בעץ אבל לא שניהם. לכן נניח בלי הגבלת הכלליות ש- u_1 אינו אביו של v. מכאן שהקשת (v,u_1) אינה קשת עץ - אלא קשת חוזרת (בגרף לא מכוון אין אפשרויות אחרות).

<u>טענה 2</u>
אם (u,v) קשת חוזרת, אז קשת זו, בתוספת המסלול המחבר את u ל-v דרך העץ, מהווים מעגל.
<u>הוכחה</u>
טריוויאלי.

<u>נכונות האלגוריתם</u>
יש להוכיח את הטענות הבאות:

I. ***אם בגרף יש מעגל אז האלגוריתם ידפיס מעגל כלשהו בגרף.***
מטענה 1, אם יש מעגל נגלה קשת חוזרת. מטענה 2, הפונקציה PrintCycle באמת מדפיסה מעגל.
II. ***אם בגרף אין מעגל, אז האלגוריתם יודיע זאת.***
נוכיח שאם בגרף אין מעגל אז בהכרח אף אחת מקשתותיו לא תסווג כקשת חוזרת על-ידי האלגוריתם. נניח בשלילה שקיימת קשת שסווגה כקשת חוזרת. אזי מטענה 2 נסיק כי קיים מעגל בגרף – סתירה.

<u>יעילות</u>
הטענה, שזמן ביצוע האלגוריתם הוא O(n), נובעת מהעובדה הידועה הבאה:
בעץ עם n קדקודים יש n–1 קשתות.
מעובדה זו נסיק, שלאחר בדיקת n קשתות חייבים להיתקל בקשת חוזרת - או לסיים את הסריקה (אם אין כזו).

<u>סעיף ב'</u>
אם הגרף נתון כרשימת קדקודים ורשימת קשתות אז מרשימת הקדקודים ורשימת הקשתות אפשר לבנות מבנה של רשימות סמיכות בזמן $\Theta(n+m)$ עבור n קדקודים ו-m קשתות. ברגע שבנינו את מבנה הנתונים ניתן להפעיל DFS באופן יעיל. כדי לשמור על זמן ריצה של O(n), נכניס למבנה הנתונים רק n מן הקשתות (במקרה שיש יותר). הניתוח למעלה מראה, שזה מספיק כדי שנמצא מעגל אם קיים.

<u>שאלה 7</u>
כתבו אלגוריתם אשר בהינתן גרף מכוון פשוט G=(V, E) מייצר ומחזיר את גרף העל של G.
<u>דרישות</u>:
1. על האלגוריתם לרוץ בזמן ליניארי.
2. על האלגוריתם לייצר גרף **פשוט** (ללא לולאות וללא קשתות כפולות בין רכיבי הקשירות החזקה).

<u>פתרון:</u>
קיימות 2 גישות שונות לפתרון הבעיה הנתונה.
גישה ראשונה: *שימוש באלגוריתם שריר-קוסרז'ו כקופסה שחורה.*
גישה שנייה: *ביצוע שינויים באלגוריתם שריר-קוסרז'ו על מנת שייצר את גרף העל (במקום להחזיר את חלוקת הקדקודים לרכיבי קשירות חזקה).*

נתאר את <u>הרעיון הכללי</u> של האלגוריתם בכל אחת מהגישות:

גישה ראשונה:
1. *נריץ אלגוריתם שריר קוסרז'ו על הגרף G ונקבל כפלט מערך Root באורך n (מס' הקדקודים ב-G), כך ש-Root[v] הוא הנציג של רכיב הקשירות החזקה של קדקוד v.*
2. *נעבור על כל קשתות הגרף G, ולכל קשת (u,v), אם הקשת מחברת בין 2 רק"חים שונים נכניס את הקשת (Root[u], Root[v]) למערך חדש ("מערך הקשתות").* *(הבחנה: במערך הקשתות תיתכנה כפילויות.)* $d(a,b)$
3. *נבצע מיון של הערכים ב"מערך הקשתות" באופן הבא: תחילה נבצע מיון לפי הערך של קדקוד היציאה, ולאחר מכן קשתות בעלות קדקוד יציאה זהה תמוינה על פי קודקוד הכניסה. כל אחד מהמיונים יתבצע בעזרת Bucket-Sort (זה אפשרי מאחר ששמות הקדקודים ב-G הם בין 1 ל-n). למעשה ביצענו כאן סוג של Radix-Sort כאשר מספר הספרות הוא 2 ובסיס הספירה הוא n+1 (יעילות: O(n+m)).*
4. *נייצר גרף חדש G* ריק מקשתות, שבו נמצאים רק הקדקודים המהווים נציגים של רכיבי קשירות חזקה ב-G (חשבו כיצד ניתן לעשות זאת בזמן O(n)).*
5. *נעבור על "מערך הקשתות" הממוין מתחילתו ועד סופו, ונוסיף ל-G* כל קשת ששונה מקודמתה במערך (וגם את הקשת הראשונה).*

גישה שנייה:
האלגוריתם ייצר גרף חדש (זמני) G' בעל n קדקודים הזהים לקדקודי הגרף G, וללא קשתות (הגרף ייוצג כרשימות שכנויות). גרף זה יאפשר לנו להוסיף קשתות השייכות לגרף העל תוך כדי הריצה של שריר-קוסרז'ו, עוד בטרם אנחנו יודעים מהו מספר הקדקודים בגרף העל.
תוך כדי ביצוע ה-DFS השני של שריר קוסרז'ו (המתבצע על G^T), האלגוריתם יאתר קשתות חוצות בין עצים (חשבו כיצד ניתן לזהות קשת כזו?). אם זוהתה קשת שכזו (u,v) ב-G^T, יש להוסיף את הקשת (v,u) לגרף G'. אך תחילה יש לוודא שהקשת לא התווספה עוד קודם לגרף.
פתרון נאיבי ולא יעיל, יעבור על כל הקשתות ברשימת השכנים של v ב-G' ויחפש את הקשת.
על מנת לייצר אלגוריתם ליניארי, האלגוריתם יבדוק האם הקשת <u>הראשונה</u> ברשימת השכנים של v בגרף G' היא הקשת (v,u). אם לא – האלגוריתם יוסיף את הקשת (v,u) כקשת ראשונה ברשימה. (נכונות האלגוריתם נובעת מכך שאם הקשת (v,u) אינה ראשונה ברשימה, אז בהכרח היא לא נמצאת בגרף. חשבו כיצד ניתן להוכיח טענה זו).
בסיום הריצה של שריר-קוסרז'ו, האלגוריתם ייצג גרף חדש G שבו מספר הקדקודים זהה למספר רכיבי הקשירות החזקה ב-G ויוסיף את הקשתות מ-G' בהתאם.*