סריקות לעומק ורדוקציות

מפגש 2

שבוע שעבר ראינו

שבוע שעבר ראינו

- מבוא והגדרות בסיסיות

שבוע שעבר ראינו

- מבוא והגדרות בסיסיות
- חזרה מהירה על ניתוח זמני ריצה

שבוע שעבר ראינו

- מבוא והגדרות בסיסיות
- חזרה מהירה על ניתוח זמני ריצה
- מושגים בסיסיים מעולם הגרפים

התוכנית להיום

התוכנית להיום

- מעבר על BFS

התוכנית להיום

- מעבר על BFS
- תכנון אלגוריתם מבוסס רדוקציה

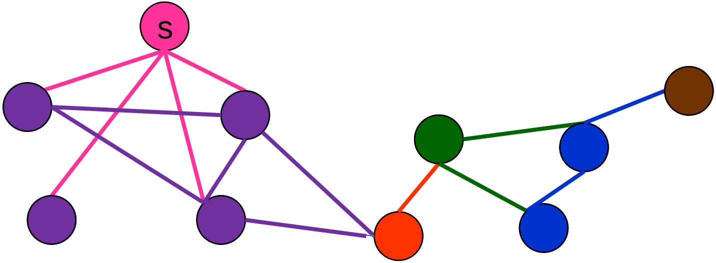
התוכנית להיום

- מעבר על BFS
- תכנון אלגוריתם מבוסס רדוקציה
- DFS -כמה שהזמן יאפשר לנו

חלק I

BFS - סריקה לרוחב

BFS

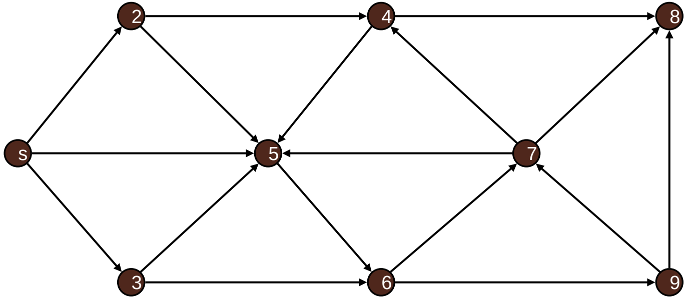


L_0 L_3

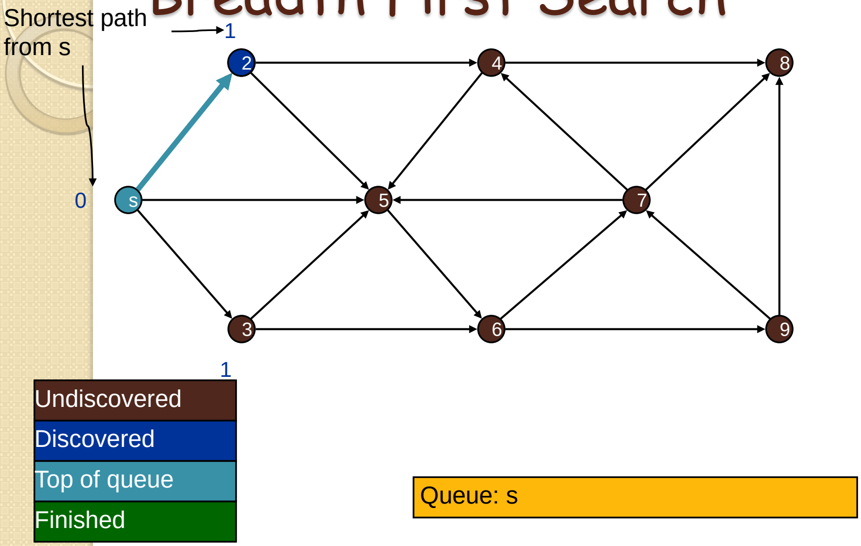
L_1 L_4

L_2 L_5

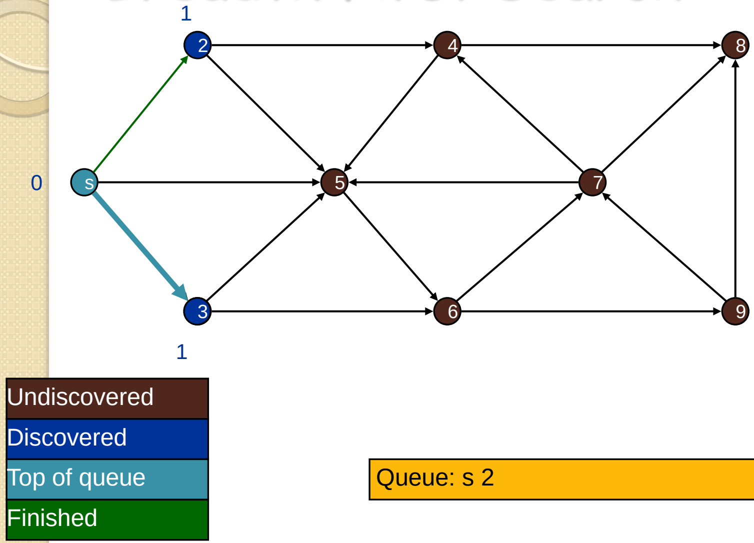
Breadth First Search



Breadth First Search

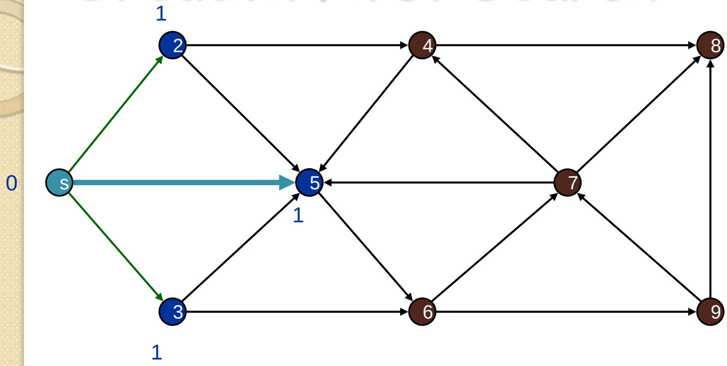


Breadth First Search



Queue: s 2

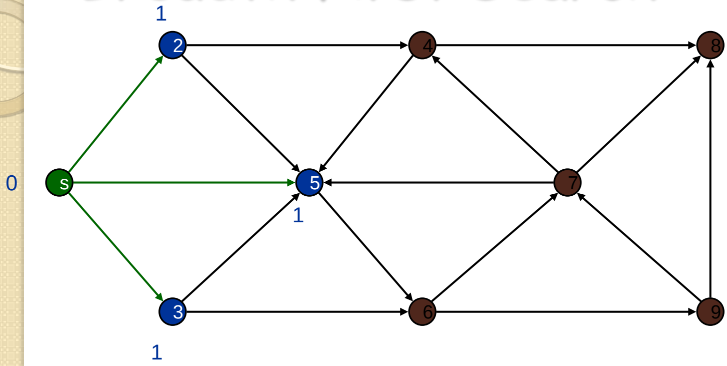
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: s 2 3

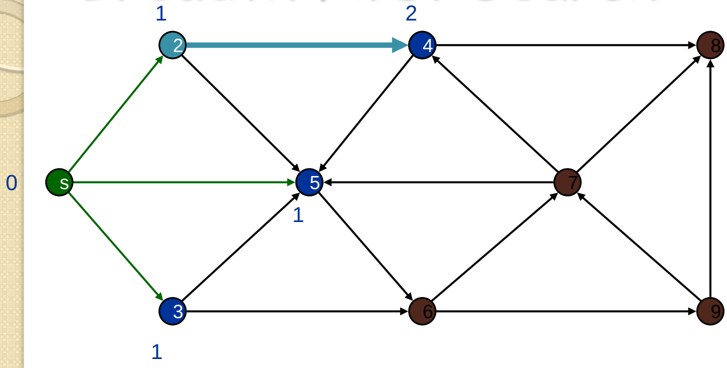
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 2 3 5

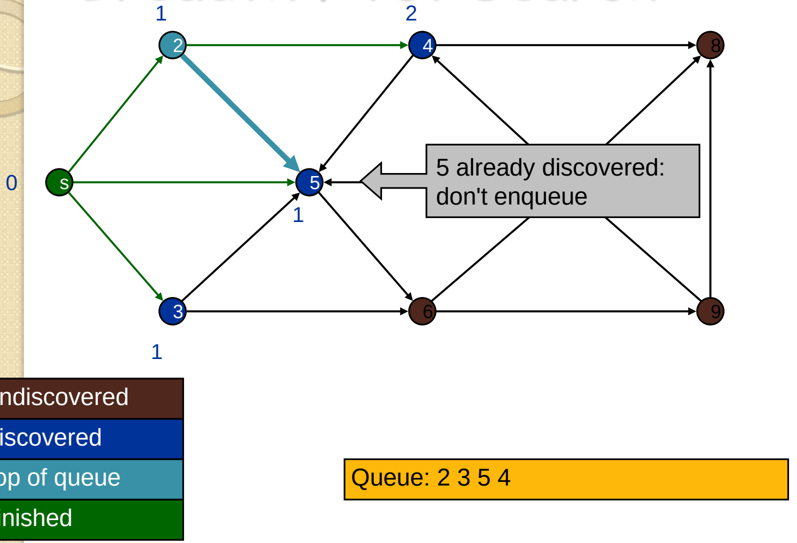
Breadth First Search



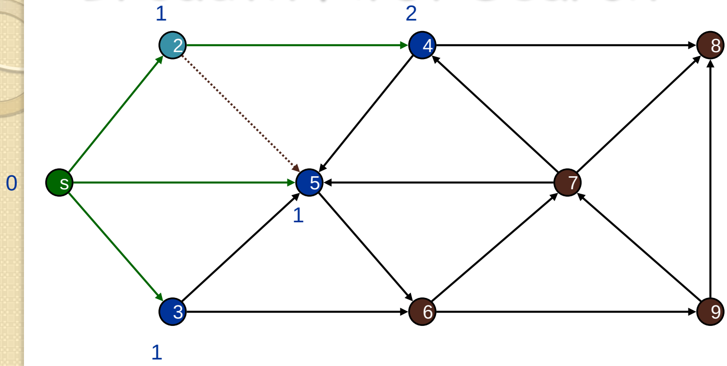
Undiscovered
Discovered
Top of queue
Finished

Queue: 2 3 5

Breadth First Search



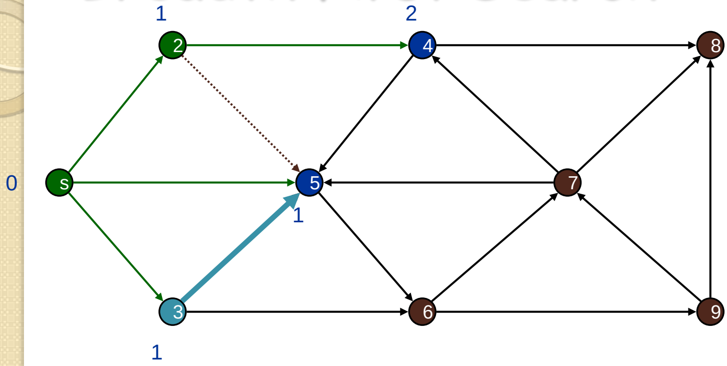
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 2 3 5 4

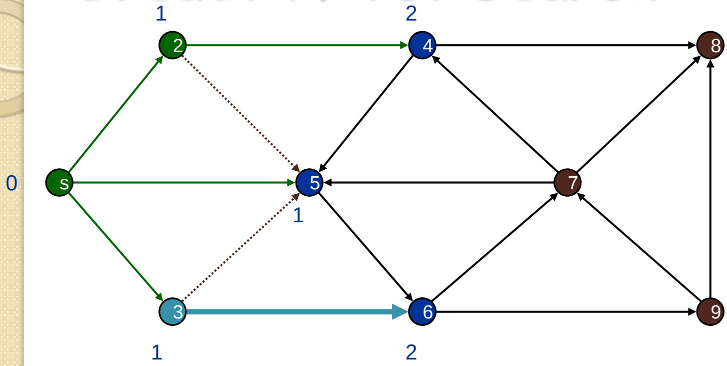
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 3 5 4

Breadth First Search



Undiscovered

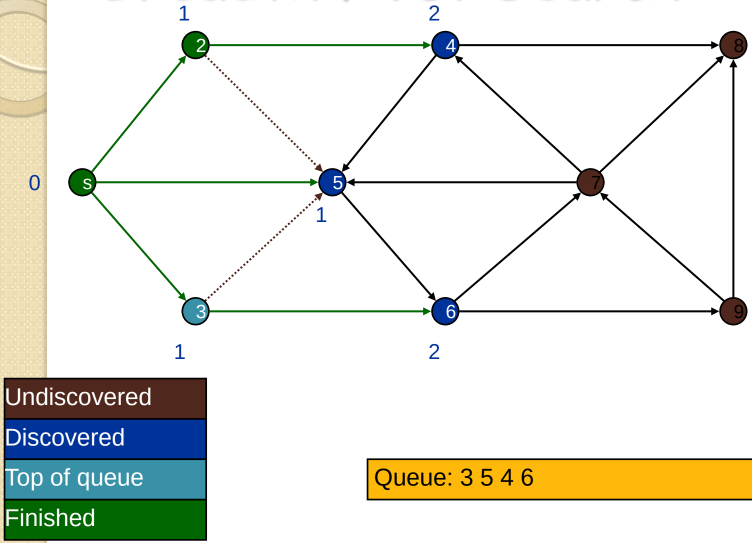
Discovered

Top of queue

Finished

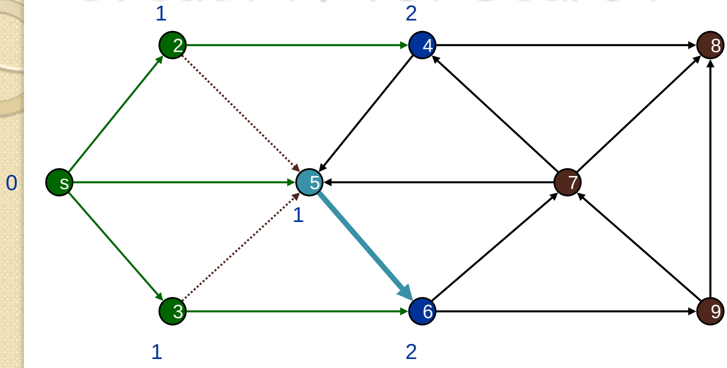
Queue: 3 5 4

Breadth First Search



Queue: 3 5 4 6

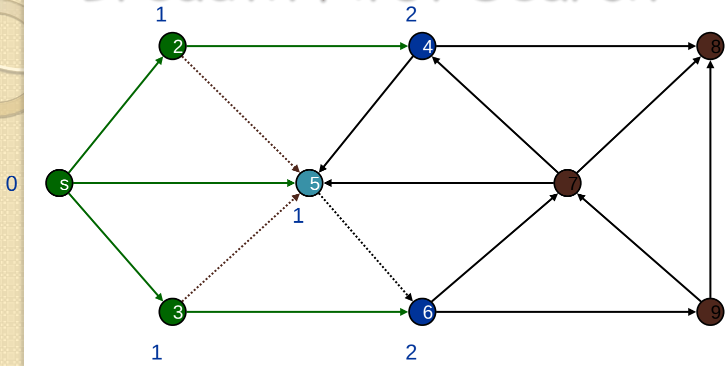
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 5 4 6

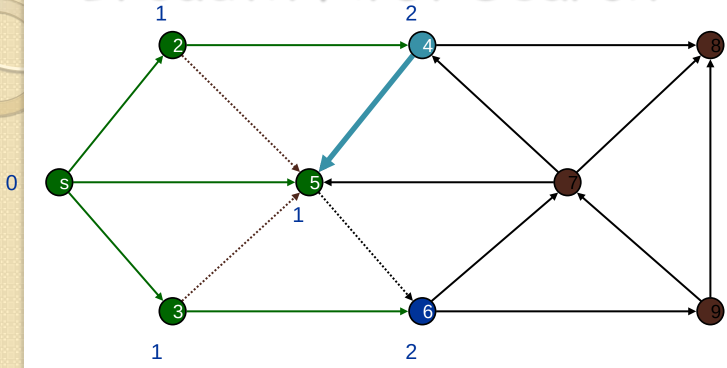
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 5 4 6

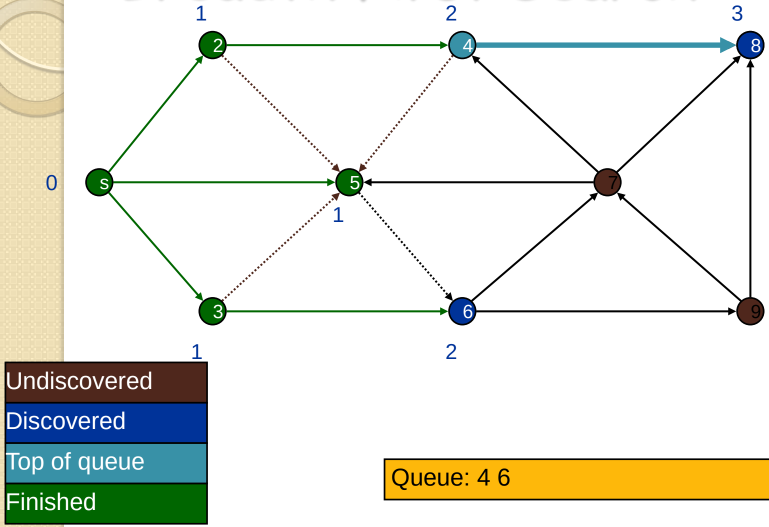
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

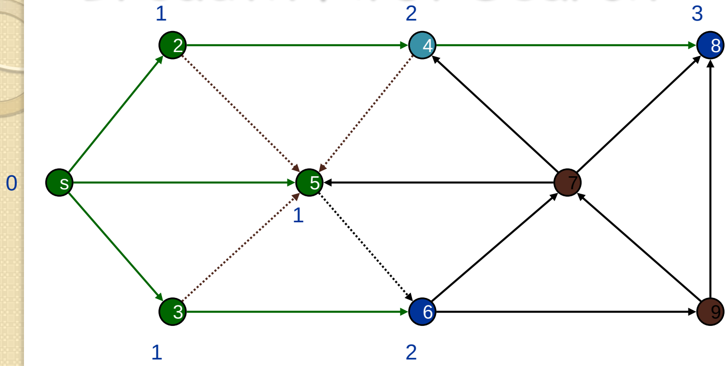
Queue: 4 6

Breadth First Search



Queue: 4 6

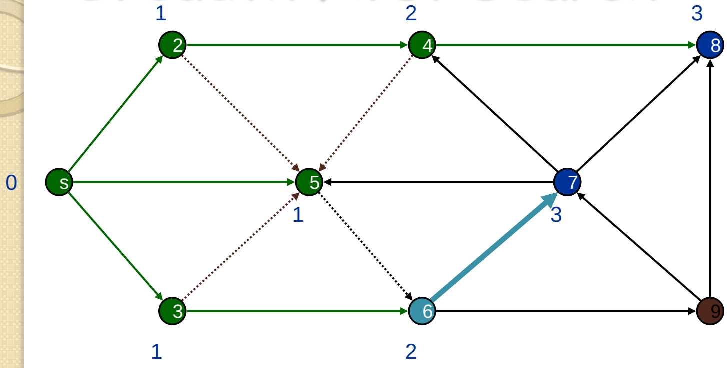
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 4 6 8

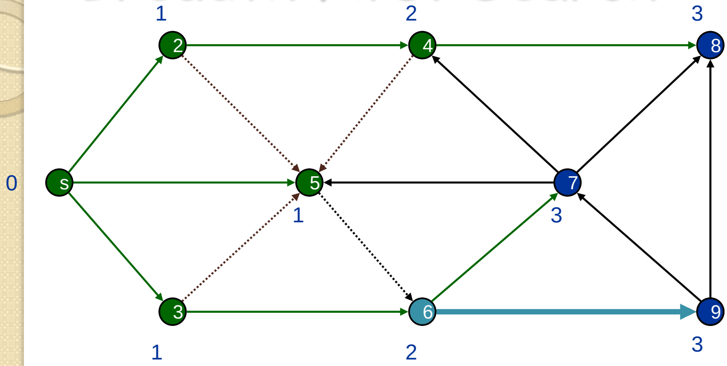
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 6 8

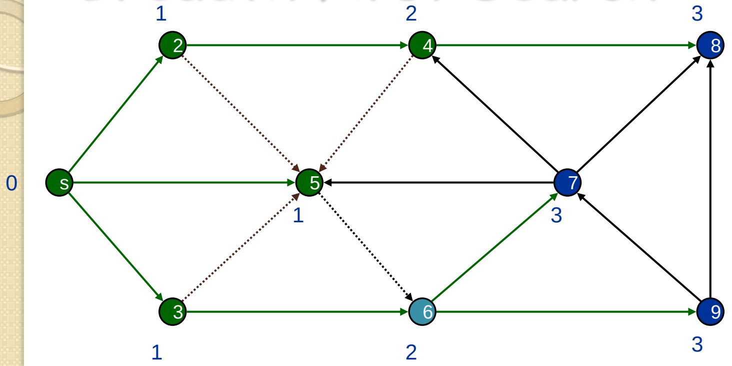
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 6 8 7

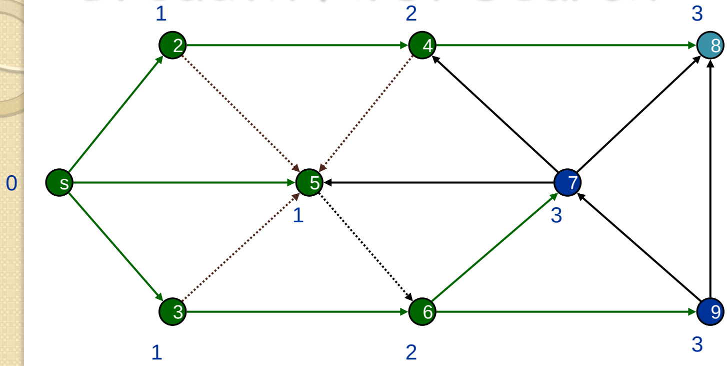
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 6 8 7 9

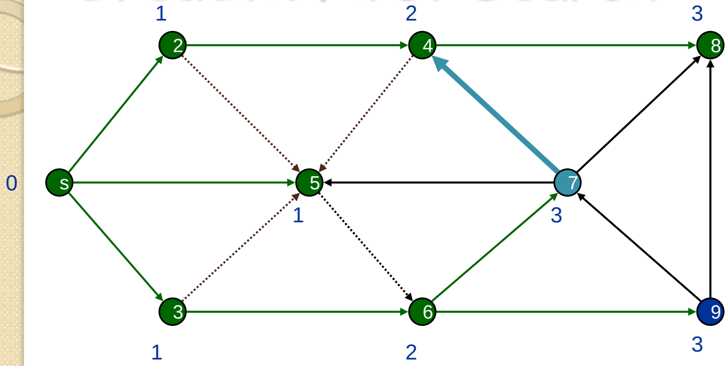
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 8 7 9

Breadth First Search



Undiscovered

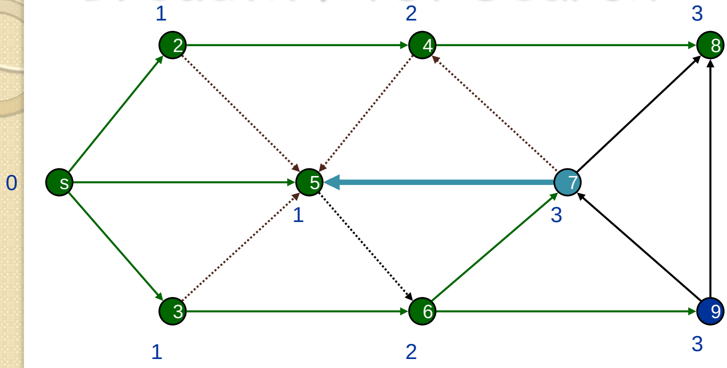
Discovered

Top of queue

Finished

Queue: 7 9

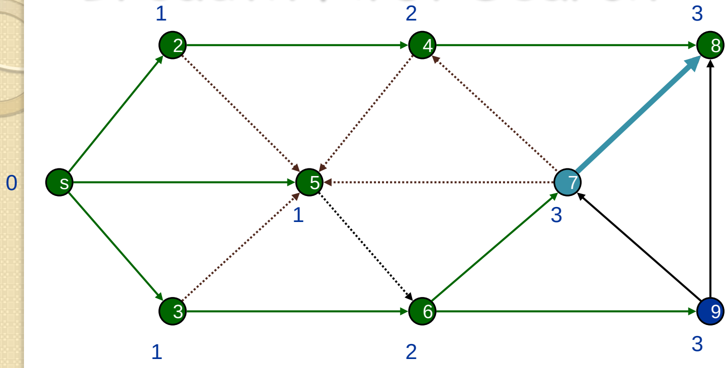
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 7 9

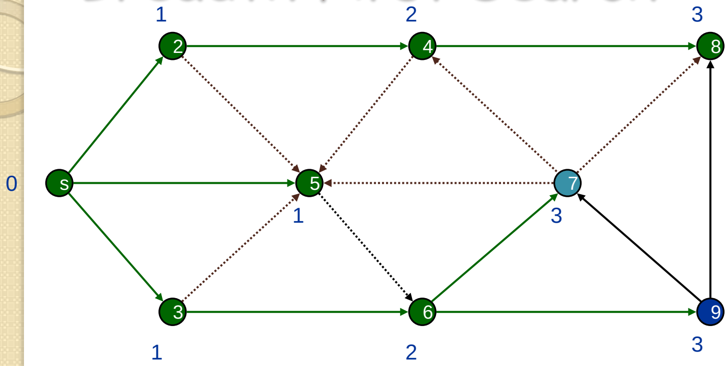
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 7 9

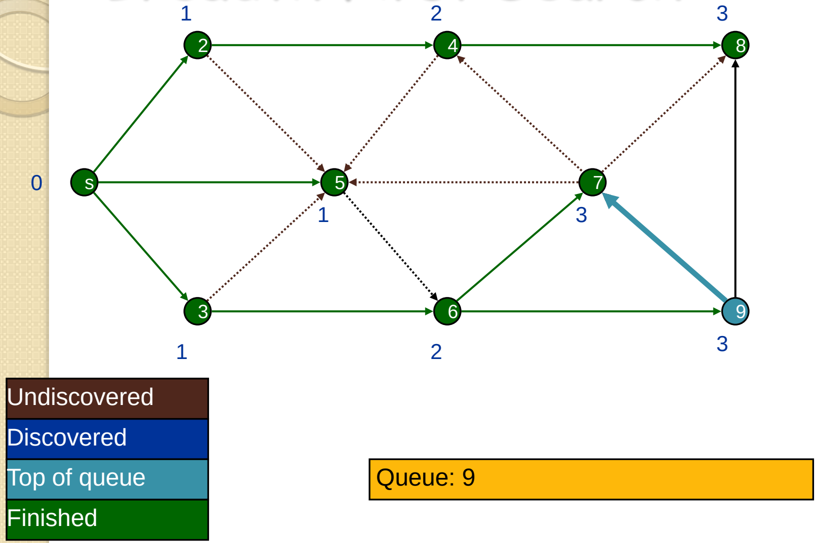
Breadth First Search



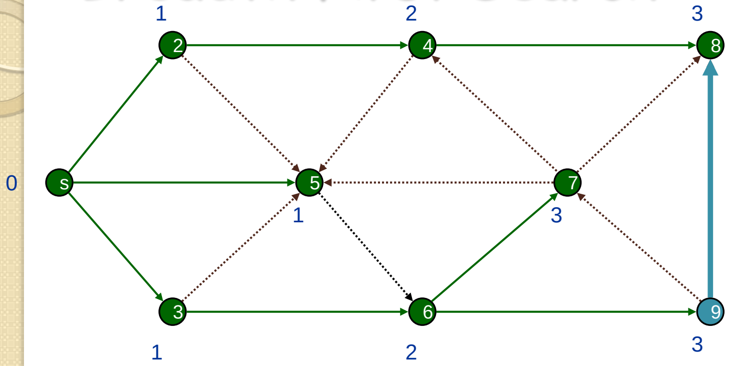
Undiscovered
Discovered
Top of queue
Finished

Queue: 7 9

Breadth First Search



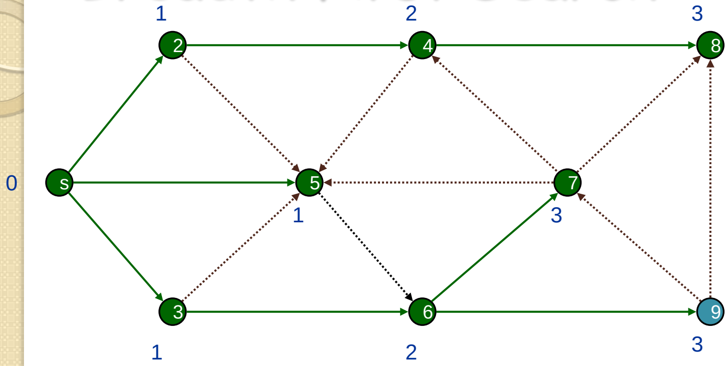
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 9

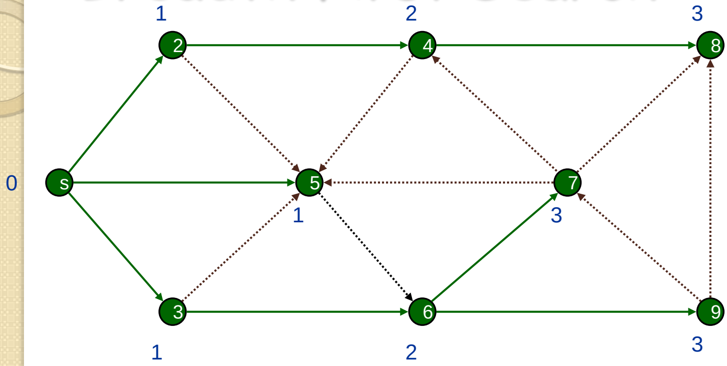
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 9

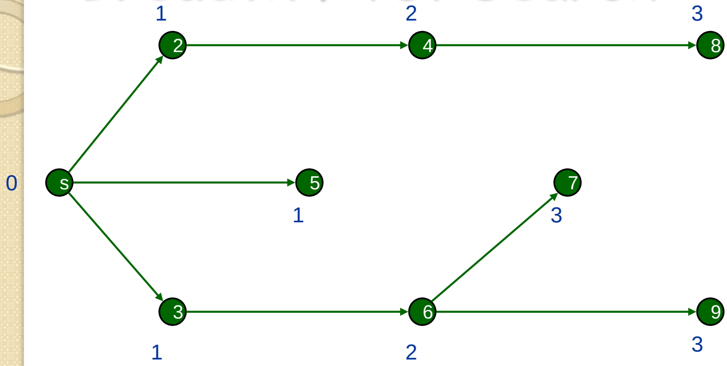
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue:

Breadth First Search



Level Graph

BFS- האלגוריתם

```
color[u]=white for all u ∈ V
BFS(s)
    color[s] = gray
    d[s] = 0
    ENQUEUE(Q, s)
    WHILE Q not empty:
        DEQUEUE(Q, u)
        FOR (u, v) ∈ E DO
            IF color[v] = white THEN
               color[v] = gray
               d[v] = d[u] + 1
               parent[v] = u
               ENQUEUE(Q, v)
            color[u] = black
```

שימושים של BFS

שימושים של BFS

- האם יש מסלול מ- u ל- v?

שימושים של BFS

- האם יש מסלול מ- u ל- v?
- מה ארכו של המסלול הקצר ביותר

שימושים של BFS

- האם יש מסלול מ- u ל- v?
- מה ארכו של המסלול הקצר ביותר
- מציאת רכיבי קשירות בגרף לא מכוון

BFS לגרף לא קשיר

```
FOR each vertex u ∈ V DO
        IF color[u] = white THEN BFS(u)
OD
```

100011101 מתחלק ב7?

חלק II

תכנון אלגוריתם באמצעות רדוקציה

דוגמא 1

הגדרה יהי $G=(V,E)$ גרף לא מכוון ויהיו $u,v\in V$ זוג קודקודים ב- G.

דוגמא 1

הגדרה יהי $G=(V,E)$ גרף לא מכוון ויהיו $u,v\in V$ זוג קודקודים ב- G.
נסמן ב- $\delta(u,v)$ את אורך המסלול המינימאלי בצלעות מקודקוד u לקודקוד v.

דוגמא 1

הגדרה יהי $G = (V, E)$ גרף לא מכוון ויהיו $u, v \in V$ זוג קודקודים ב- G.
נסמן ב- $\delta(u, v)$ את אורך המסלול המינימאלי בצלעות מקודקוד u לקודקוד v. בנוסף, נגדיר כי $\delta(u, v) = \infty$ אם לא קיים מסלול מ-u ל- v ב- G.

דוגמא 1

הגדרה יהי $G = (V, E)$ גרף לא מכוון ויהיו $u, v \in V$ זוג קודקודים ב- G.
נסמן ב- $\delta(u, v)$ את אורך המסלול המינימאלי בצלעות מקודקוד u לקודקוד v. בנוסף, נגדיר כי $\delta(u, v) = \infty$ אם לא קיים מסלול מ-u ל- v ב- G.

הגדרת בעיית SP (Single Source Shortest Paths):

דוגמא 1

הגדרה יהי $G = (V, E)$ גרף לא מכוון ויהיו $u, v \in V$ זוג קודקודים ב- G.
נסמן ב- $\delta(u, v)$ את אורך המסלול המינימאלי בצלעות מקודקוד u לקודקוד v. בנוסף, נגדיר כי $\delta(u, v) = \infty$ אם לא קיים מסלול מ-u ל- v ב- G.

הגדרת בעיית SP (Single Source Shortest Paths):
יהי $G = (V, E)$ גרף לא מכוון ו- $s \in V$ קודקוד ב- G.

דוגמא 1

הגדרה יהי $G=(V,E)$ גרף לא מכוון ויהיו $u,v \in V$ זוג קודקודים ב- G.
נסמן ב- $\delta(u,v)$ את אורך המסלול המינימאלי בצלעות מקודקוד u לקודקוד v. בנוסף, נגדיר כי $\delta(u,v)=\infty$ אם לא קיים מסלול מ-u ל- v ב- G.

הגדרת בעיית SP (Single Source Shortest Paths):
יהי $G=(V,E)$ גרף לא מכוון ו- $s \in V$ קודקוד ב- G. מצא $\delta(s,v)$ לכל $v \in V$.

דוגמא 1

הגדרה יהי $G=(V,E)$ גרף לא מכוון ויהיו $u,v\in V$ זוג קודקודים ב- G.
נסמן ב- $\delta(u,v)$ את אורך המסלול המינימאלי בצלעות מקודקוד u לקודקוד v. בנוסף, נגדיר כי $\delta(u,v)=\infty$ אם לא קיים מסלול מ-u ל- v ב- G.

הגדרת בעיית SP (Single Source Shortest Paths):
יהי $G=(V,E)$ גרף לא מכוון ו- $s\in V$ קודקוד ב- G. מצא $\delta(s,v)$ לכל $v\in V$.

הגדרת בעיית SEP (Single Source Shortest Even Paths):

דוגמא 1

הגדרה יהי $G=(V,E)$ גרף לא מכוון ויהיו $u,v\in V$ זוג קודקודים ב- G.
נסמן ב- $\delta(u,v)$ את אורך המסלול המינימאלי בצלעות מקודקוד u לקודקוד v. בנוסף, נגדיר כי $\delta(u,v)=\infty$ אם לא קיים מסלול מ-u ל- v ב- G.

הגדרת בעיית SP (Single Source Shortest Paths):
יהי $G=(V,E)$ גרף לא מכוון ו- $s\in V$ קודקוד ב- G. מצא $\delta(s,v)$ לכל $v\in V$.

הגדרת בעיית SEP (Single Source Shortest Even Paths):
יהי $G=(V,E)$ גרף לא מכוון ו- $s\in V$ קודקוד ב- G.

דוגמא 1

הגדרה יהי $G=(V,E)$ גרף לא מכוון ויהיו $u,v\in V$ זוג קודקודים ב- G.
נסמן ב- $\delta(u,v)$ את אורך המסלול המינימאלי בצלעות מקודקוד u לקודקוד v. בנוסף, נגדיר כי $\delta(u,v)=\infty$ אם לא קיים מסלול מ-u ל- v ב- G.

הגדרת בעיית SP (Single Source Shortest Paths):
יהי $G=(V,E)$ גרף לא מכוון ו- $s\in V$ קודקוד ב- G. מצא $\delta(s,v)$ לכל $v\in V$.

הגדרת בעיית SEP (Single Source Shortest Even Paths):
יהי $G=(V,E)$ גרף לא מכוון ו- $s\in V$ קודקוד ב- G. מצא את אורך המסלול (לפי צלעות) הזוגי המינאמלי מ- s לכל $v\in V$ או ∞ אם לא קיים כזה.

- איך פותרים את SP?

- איך פותרים את SP?
- בהינתן של-SP יש אלגוריתם כיצד נבנה אלגוריתם עבור SEP?

שאלה 1 - הראו אלגוריתם רדוקציה מ-SEP ל-SP

חישוב זמן ריצה לאלגוריתם רדוקציה

כפי שראינו ישנם שלושה רכיבים הדורשים זמן חישוב בעת הרצת האלגוריתם:

חישוב זמן ריצה לאלגוריתם רדוקציה

כפי שראינו ישנם שלושה רכיבים הדורשים זמן חישוב בעת הרצת האלגוריתם:

- חישוב תרגום הקלט,

חישוב זמן ריצה לאלגוריתם רדוקציה

כפי שראינו ישנם שלושה רכיבים הדורשים זמן חישוב בעת הרצת האלגוריתם:

- חישוב תרגום הקלט,
- הפעלת האלגוריתם לפתרון הבעיה המתורגמת,

חישוב זמן ריצה לאלגוריתם רדוקציה

כפי שראינו ישנם שלושה רכיבים הדורשים זמן חישוב בעת הרצת האלגוריתם:

- חישוב תרגום הקלט,
- הפעלת האלגוריתם לפתרון הבעיה המתורגמת,
- וחישוב תרגום הפלט.

חישוב זמן ריצה לאלגוריתם רדוקציה

חישוב זמן ריצה לאלגוריתם רדוקציה

- זמן חישוב האלגוריתם המבוסס על הרדוקציה הינו הזמן המקסימאלי הדרוש מבין השלושה.

חישוב זמן ריצה לאלגוריתם רדוקציה

- זמן חישוב האלגוריתם המבוסס על הרדוקציה הינו הזמן המקסימאלי הדרוש מבין השלושה.
- נתייחס לאלגוריתם שפותר את בעיה המתורגמת כאל "**קופסה שחורה**", מבלי להניח דבר על אופן פעולתו מלבד העובדה שהוא אכן פותר נכון את הבעיה השניה לכל מופע.

חישוב זמן ריצה לאלגוריתם רדוקציה

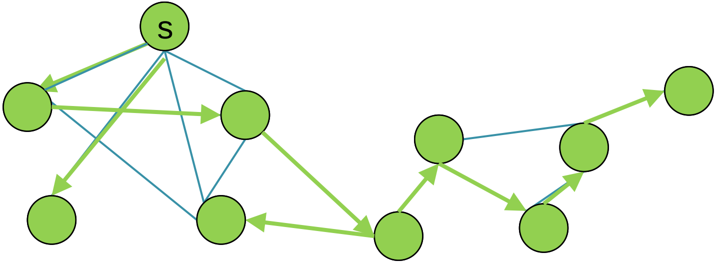
- זמן חישוב האלגוריתם המבוסס על הרדוקציה הינו הזמן המקסימאלי הדרוש מבין השלושה.
- נתייחס לאלגוריתם שפותר את בעיה המתורגמת כאל "**קופסה שחורה**", מבלי להניח דבר על אופן פעולתו מלבד העובדה שהוא אכן פותר נכון את הבעיה השניה לכל מופע.
- נשים לב שזמן הריצה של הקופסא השחורה הוא על גודל הקלט החדש (הפלט של ממיר הקלט).

שאלה 2 - אלגוריתם רדוקציה מ-SP ל-SEP

תרגיל למחשבה

הגדרת בעיית SOP (Single Source Shortest Odd Paths):
יהי $G = (V, E)$ גרף לא מכוון ו- $s \in V$ קודקוד ב- G. מצא את אורך המסלול (לפי צלעות) **האי-זוגי** המינאמלי מ- s לכל $v \in V$ או ∞ אם לא קיים כזה.

DFS



DFS - האלגוריתם

```
t = 0     color[u]=white for all u ∈ V
DFS(u)
    color[u] = gray
    d[u] = t
    t = t + 1
    FOR (u, v) ∈ E DO
        IF color[v] = white THEN
            parent[v] = u
            DFS(v)
        FI
    OD
    color[u] = black
    f[u] = t
    t = t + 1
```

דוגמא על הלוח

תכונות DFS

תכונות DFS

- סורק את כל הקודקודים

תכונות DFS

- סורק את כל הקודקודים
- מוצא מעגלים

תכונות DFS

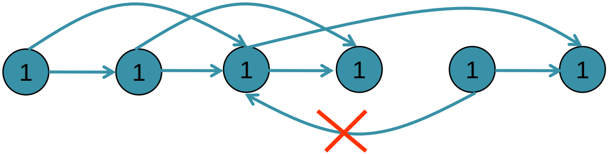
- סורק את כל הקודקודים
- מוצא מעגלים
- סיבוכיות $O(|V| + |E|)$

שאלה 3

הוכיחו או הפריכו את הטענה הבאה: יהי גרף קשיר ולא מכוון ויהי $u \in V$. אם קיימת הרצת DFS מ- u על G והרצת BFS מ- u על G הנותנות את אותו עץ T אז בהכרח $T = G$

מיון טופולוגי

- מיון טופולוגי – מיון טופולוגי של גרף מכוון $G=(V,E)$ הינו סידור
$(v_1,\ldots,v_n)$ של קודקודי הגרף, כך שלכל
$1 \le i,j \le n$
אם $j < i$ אז *אין* קשתות מ j ל i בגרף.



מיון טופולוגי

משפט: אם הגרף גמ"ל אזי יש מיון טופולוגי.

מוכיחים בבניה – נותנים אלגוריתם שמוצא מיון טופולוגי (עמוד 111 בספר).

- רכיב קשיר היטב: תת קבוצה מקסימאלית U כך שכל שני קודקודים בU ניתנים להגעה הדדית.

מיון טופולוגי

טענה: כל שני רכיבים קשירים היטב בגרף הם זרים.

הוכחה: תרגיל קל.

מסקנה(בסיסית וחשובה): אוסף הרכיבים הקשירים היטב מהווה חלוקה של הגרף.

הוכחה: ברור שכל קודקוד בגרף נמצא ברכיב קשיר שכן לפחות הוא בעצמו מהווה קבוצה קשירה היטב.

לכן איחוד כל הרכיבים הקשירים היטב הוא כל הקודקודים.

שנית, כל שני רכיבים קשירים היטב הם זרים.

וסיימנו.

שאלה

- **הגדרה: גרף מעורב** הוא גרף שבו כמה מהקשתות מכוונות והאחרות אינן מכוונות.
- הוכיחו שאם בגרף מעורב התת-גרף המתקבל מכל צמתי הגרף ומהקשתות המכוונות בלבד אינו מכיל מעגל מכוון, אז תמיד ניתן לכוון את הקשתות הלא מכוונות כך שבגרף המכוון המתקבל אין מעגל מכוון. הראו כיצד ניתן למצוא כיוון מתאים לקשתות בזמן $O(|V|+|E|)$.
- רמז: מצאו קודם את האלגוריתם המבצע את הכיוון ומהוכחת נכונותו הסיקו כי קיים כיוון כנדרש.

תשובה

- נפתור ע"י רדוקציה למיון טופולוגי. בשלב ראשון נתייחס לתת-גרף המכוון G', הכולל רק את הקשתות המכוונות. מאחר שזהו גרף מכוון חסר מעגלים אפשר להפעיל עליו מיון טופולוגי בזמן ליניארי, ולקבל סידור של הצמתים כך שכל קשתות הגרף מכוונות תמיד מצומת בעל מספר סידורי נמוך לצומת בעל מספר סידורי גבוה. כעת נוסיף את הקשתות הלא מכוונות, ונכוון אותן כך שכל קשת תהיה תמיד מצומת שמספרו הסידורי במיון הטופולוגי נמוך יותר אל צומת שמספרו הסידורי במיון הטופולוגי גבוה יותר.

תשובה

- התקבל גרף מכוון שבו כל הקשתות מכוונות תמיד מצומת בעל מספר סידורי נמוך לצומת בעל מספר סידורי גבוה. נניח שיש בגרף מעגל מכוון $v_1, \ldots, v_n, v_1$ ויהי $d(v_i)$ מספרו הסידורי של v_i במיון הטופולוגי.
- אז נקבל $d(v_1) < \ldots < d(v_n) < d(v_1)$, כלומר $d(v_1) < d(v_1)$. לכן, בהכרח אין מעגל בגרף.

תרגיל

תרגיל: נתון גרף מכוון.

כיצד יכול להשתנות מספר הרכיבים הקשירים היטב בגרף אם מוסיפים קשת חדשה?

תשובה:

נניח שהיו K רכיבים.

- המספר רק יכול לקטון : הרכיבים הקשירים היטב שהושרו מהגרף G נותרו קשירים היטב אך לאו דווקא מקסימליים בגרף G‘: הוספת הקשת יכולה לגרום לאיחוד של שני רכיבים כאלו (או יותר) .

- עבור כל i≤K ניתן למצוא דוגמא בה מספר הרכיבים הקשירים היטב הופך ל i. מהי?

תרגיל

$G = (V, E)$ גרף לא-מכוון נתון.

הראה שאם קיימת ריצת DFS על G בה צומת $v \in V$ הוא עלה, אז קיים ב- G מסלול פשוט העובר דרך כל שכניו של v באופן ש- v איננו על המסלול.

תשובה

- משפט המסלול הלבן- ביער DFS מכוון/לא מכוון צומת v הוא צאצא של u אם"ם כש-u התגלה קיים מסלול מ-u ל- v שמכיל רק צמתים שעוד לא התגלו.

תשובה

- נתבונן בריצת DFS בה v הינו עלה ונסמן את שכני v לפי סדר הופעתם בעץ ה DFS , $v_1,v_2,\ldots,v_k$. ממשפט המסלול הלבן v_i הוא אב קדמון של v_{i+1} (המסלול הלבן הוא פשוט v_i-v-v_{i+1} שכן v התגלה לאחר כל שכניו, אחרת לא היה עלה).

לכן נוכל ללכת בעץ ה- DFS מ- v_1 לצאצא v_2 ואחר כך לצאצא של v_2 שהוא v_3 וכיוצא בזה עד ל- v_k. מסלול זה אינו עובר דרך v, כיון ש- v הוא עלה.

פתרון נוסף שהוצע במפגש:

- נתבונן בריצת DFS בה v הינו עלה ונסמן את העץ הנוצר מריצה זאת ב-T.
- נסמן את השכנים של v ב-$v_1,v_2,\ldots,v_k$.
- לפי משפט 3.7 בעמוד 92 בספר (ההבחנה+ שהמשפט נכון גם עבור קשתות עץ),עבור כל זוג צמתים x ו-y המחוברים בקשת, מתקיים ש- או x או y , הוא אב קדמון של השני. בפרט מתקיים עבור כל v_i $(1 \le i \le k)$ ש- או ש-v אב קדמון של v_i או ש-v_i אב קדמון של v בעץ T. מכיוון שv עלה ב-T, נקבל שכל v_i הוא אב קדמון של v. מכך נובע שבמסלול הפשוט מהשורש s לאבא הישיר של v בעץ T מופיעים כל שכני v - $v_1,v_2,\ldots,v_k$, בנוסף מסלול זה לא מכיל את v, וסיימנו.

שאלה

- הוכח או הפרך:

אם בגרף מכוון יש קשתות הנכנסות לצומת u וגם קשתות היוצאות ממנו, אזי לא ייתכן שבהרצת DFS על הגרף הצומת u יימצא בעץ המכיל אותו בלבד.

תשובה

- הטענה אינה נכונה.
- . למשל נסתכל על גרף שבו שלושה צמתים 1, 2, ו-3 ושתי קשתות (1, 2) ו-(2, 3), ונניח שבייצוג הגרף מופיע קודם כל הצומת 3, אח"כ הצומת 2 ואח"כ הצומת 1. במקרה זה, הצומת 2 יהיה מבודד ביער ה-DFS.

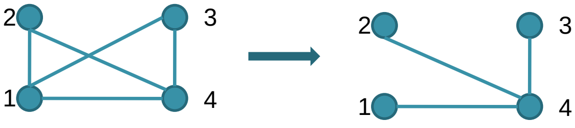
שאלה

- הוכיחו או הפריכו:

לכל גרף קשיר ולא מכוון , לכל מעגל פשוט C ב-G ולכל ריצת DFS על G, בהכרח יש ב-C בדיוק קשת אחורה אחת.

תשובה

הטענה אינה נכונה: למשל, גרף לא מכוון ובו ארבעה צמתים {1, 2, 3, 4} וקשתות (1, 2), (1, 3), (1, 4), (2, 4) ו-(3, 4) (זהו ריבוע עם אלכסון אחד). ריצת DFS חוקית על הגרף הזה מהצומת 1 יכולה ליצור את העץ המכיל את הקשתות (1, 4), (2, 4) ו-(3, 4). לכן המעגל הפשוט המורכב מארבע הקשתות (1, 2), (1, 3), (2, 4) ו-(3, 4) מכיל שתי קשתות אחורה (שתי הראשונות).



שאלה

- הוכיחו או הפריכו:
- יהי גרף קשיר ולא מכוון, יהי $s \in V$ ויהי T עץ המתקבל מהרצת DFS על G מ-s. אז עומקו של T הוא לפחות כעומקו של כל עץ המתקבל מהרצת BFS על G מ-s.

תשובה

- הטענה נכונה: נניח בשלילה שיש עץ המתקבל מהרצת BFS על G מ-s שעומקו גדול משל T. יהי v הצומת העמוק ביותר בעץ ה-BFS, ויהי d עומקו של v. מאחר שזהו עץ מסלולים קצרים הרי מרחקו של v מ-s הוא בדיוק d. אבל עומקו של T קטן מ-d, ובפרט עומקו של v ב-T קטן מ-d, כלומר המסלול בעץ T מ-s אל v קצר מ-d, בסתירה לנכונות BFS.

שאלה

- יהי $G=(V,E)$ גרף לא מכוון וקשיר. הוכח או הפרך:
- א. כל עץ המתקבל מריצת DFS על G ניתן לקבל על ידי ריצת BFS על G.
- ב. כל עץ המתקבל מריצת BFS על G ניתן לקבל על ידי ריצת DFS על G.

תשובה

- בשני הסעיפים הטענה אינה נכונה. נפריך ע"י גרף מלא (קליק) G שבו 4 צמתים. כלומר, כל צומת מחובר בקשת לכל צומת אחר. בגרף זה, כל ריצת DFS, לא משנה מאיזה צומת נתחיל, נראית כמו שרוך, כלומר, מסלול בן 4 צמתים. לעומת זאת, כל ריצת BFS, לא משנה מאיזה צומת נתחיל, היא עץ שבו צומת המקור ברמה 0 וכל שאר הצמתים ברמה 1. מכאן ברור כי על גרף זה אף אחת מהטענות אינה נכונה.