ברוכים הבאים לקורס אלגוריתמים!

מה אתם עושים פה?

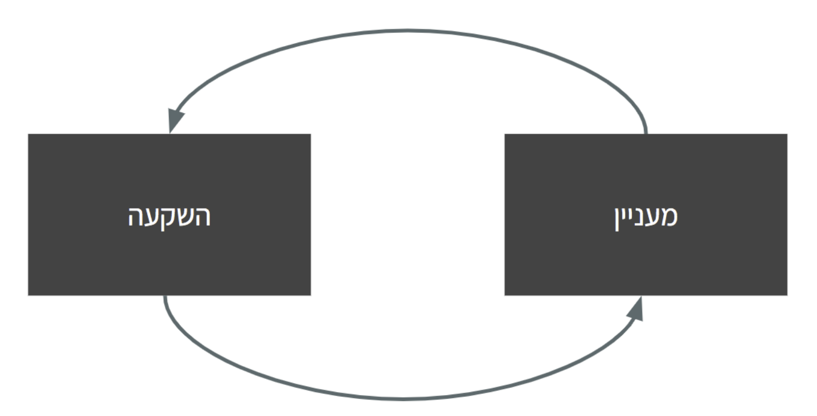
(או - מה אתם חושבים שנעשה פה)

ברוכים הבאים לקורס טעימות של אלגוריתמים!

מהי התרומה של הקורס עבורכם?

אנשים אומרים שאלגוריתמים הוא אחד הקורס *הקשים בתואר*

הם צודקים.


השקעה
מעניין

דברים שנגיד פעם אחת

דברים שנגיד פעם אחת

- oren.roth@openu.ac.il

דברים שנגיד פעם אחת

- oren.roth@openu.ac.il
- שעת הנחייה: שלישי 14:00-15:00, 0542244598

דברים שנגיד פעם אחת

- `oren.roth@openu.ac.il`
- שעת הנחייה: שלישי 14:00-15:00, 0542244598
- אתר הקורס

דברים שנגיד פעם אחת

- `oren.roth@openu.ac.il`
- שעת הנחייה: שלישי 14:00-15:00, 0542244598
- אתר הקורס
- ממ"ן ומבחן

דברים שנגיד פעם אחת

- `oren.roth@openu.ac.il`
- שעת הנחייה: שלישי 14:00-15:00, 0542244598
- אתר הקורס
- ממ"ן ומבחן
  - כל המידע הטכני מופיע בחוברת הקורס

דברים שנגיד פעם אחת

- oren.roth@openu.ac.il
- שעת הנחייה: שלישי 14:00-15:00, 0542244598
- אתר הקורס
- ממ"ן ומבחן
  - כל המידע הטכני מופיע בחוברת הקורס
  - לכל ממ"ן יהיה פורום באתר בו תוכלו לשאול שאלות

איך נגרום לזה לעבוד?

איך נגרום לזה לעבוד?

- לפני כל שיעור תקראו את החומר (לפי לוח הזמנים שמופיע בחוברת הקורס)

איך נגרום לזה לעבוד?

- לפני כל שיעור תקראו את החומר (לפי לוח הזמנים שמופיע בחוברת הקורס)
  - אנחנו נדבר בקצרה על חומרים מתוך פרקים 1-2

איך נגרום לזה לעבוד?

- לפני כל שיעור תקראו את החומר (לפי לוח הזמנים שמופיע בחוברת הקורס)
  - אנחנו נדבר בקצרה על חומרים מתוך פרקים 1-2
  - ב2-3 שבועות הראשונים נתחיל בפרק 3

איך נגרום לזה לעבוד?

- לפני כל שיעור תקראו את החומר (לפי לוח הזמנים שמופיע בחוברת הקורס)
  - אנחנו נדבר בקצרה על חומרים מתוך פרקים 1-2
  - ב2-3 שבועות הראשונים נתחיל בפרק 3
- תשלחו שאלות ונושאים שדורשים הבהרות למייל שלי לפני השיעור

איך נגרום לזה לעבוד?

- לפני כל שיעור תקראו את החומר (לפי לוח הזמנים שמופיע בחוברת הקורס)
  - אנחנו נדבר בקצרה על חומרים מתוך פרקים 1-2
  - ב2-3 שבועות הראשונים נתחיל בפרק 3
- תשלחו שאלות ונושאים שדורשים הבהרות למייל שלי לפני השיעור
- "מי שמבין חומר בצורה עמוקה יכול להסביר אותו בצורה פשוטה"

איך נגרום לזה לעבוד?

- לפני כל שיעור תקראו את החומר (לפי לוח הזמנים שמופיע בחוברת הקורס)
  - אנחנו נדבר בקצרה על חומרים מתוך פרקים 1-2
  - ב2-3 שבועות הראשונים נתחיל בפרק 3
- תשלחו שאלות ונושאים שדורשים הבהרות למייל שלי לפני השיעור
- "מי שמבין חומר בצורה עמוקה יכול להסביר אותו בצורה פשוטה"
- מה עוזר לכם ללמוד חומר מורכב?

התוכנית שלנו להיום

התוכנית שלנו להיום

- מבוא והגדרות בסיסיות

התוכנית שלנו להיום

- מבוא והגדרות בסיסיות
- חזרה מהירה על ניתוח זמני ריצה

התוכנית שלנו להיום

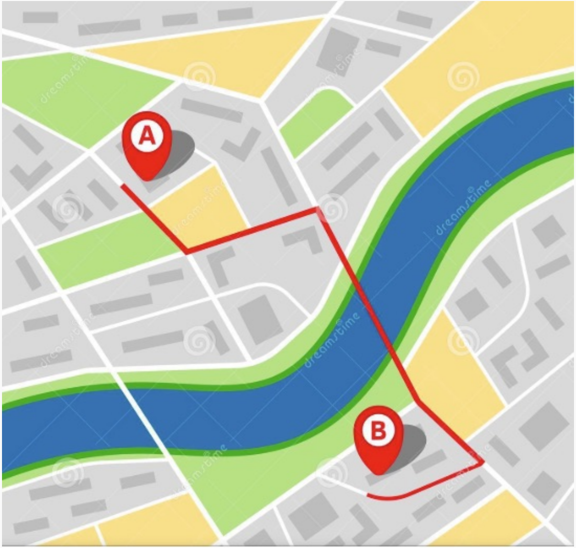
- מבוא והגדרות בסיסיות
- חזרה מהירה על ניתוח זמני ריצה
- מושגים בסיסיים מעולם הגרפים

#התחלנו

חלק I

מבוא והגדרות בסיסיות

בעיה, מופע ופתרון



מה זה אלגוריתם?

מה זה אלגוריתם?

מה זה אלגוריתם?

- סדרה של פעולות

מה זה אלגוריתם?

- סדרה של פעולות
- בסיומה מוחזר פלט

מה זה אלגוריתם?

- סדרה של פעולות
- בסיומה מוחזר פלט
- בקורס אנחנו מתרכזים רק באלגורתמים שתמיד מחזירים תשובה נכונה **לכל מופע**

הוכחת נכונות פורמאלית

דוגמא

נתון אלגוריתם אשר "לכל מפת כבישים, עיר מוצא ועיר יעד, האלגוריתם יחזיר מסלול קצר ביותר במפה בין שתי הערים".

דוגמא

נתון אלגוריתם אשר "לכל מפת כבישים, עיר מוצא ועיר יעד, האלגוריתם יחזיר מסלול קצר ביותר במפה בין שתי הערים". עלינו להוכיח כי האלגוריתם אכן מקיים את הטענה לעיל תחת שני מקרים:

דוגמא

נתון אלגוריתם אשר "לכל מפת כבישים, עיר מוצא ועיר יעד, האלגוריתם יחזיר מסלול קצר ביותר במפה בין שתי הערים". עלינו להוכיח כי האלגוריתם אכן מקיים את הטענה לעיל תחת שני מקרים:

- ישנו מסלול במפת הכבישים בין שתי ערים נתונות.

דוגמא

נתון אלגוריתם אשר "לכל מפת כבישים, עיר מוצא ועיר יעד, האלגוריתם יחזיר מסלול קצר ביותר במפה בין שתי הערים". עלינו להוכיח כי האלגוריתם אכן מקיים את הטענה לעיל תחת שני מקרים:

- ישנו מסלול במפת הכבישים בין שתי ערים נתונות.
- במפת הכבישים אין מסלול בין שתי ערים נתונות.

זמני ריצה

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע. בעיות שניתנות לפיתרון בזמן קבוע (כלומר אינו תלוי בגודל הקלט). לרוב, אינן מעניינות במיוחד.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר *n* מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע. בעיות שניתנות לפיתרון בזמן קבוע (כלומר אינו תלוי בגודל הקלט). לרוב, אינן מעניינות במיוחד.
- $O(\log n)$ - זמן לוגריתמי.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע. בעיות שניתנות לפיתרון בזמן קבוע (כלומר אינו תלוי בגודל הקלט). לרוב, אינן מעניינות במיוחד.
- $O(\log n)$ - זמן לוגריתמי.
- $O(n)$ - זמן לינארי.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע. בעיות שניתנות לפיתרון בזמן קבוע (כלומר אינו תלוי בגודל הקלט). לרוב, אינן מעניינות במיוחד.
- $O(log\ n)$ - זמן לוגריתמי.
- $O(n)$ - זמן לינארי.
- $O(n^c)$ - עבור $c \in \mathbb{N}$ - זמן פולינומי.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע. בעיות שניתנות לפיתרון בזמן קבוע (כלומר אינו תלוי בגודל הקלט). לרוב, אינן מעניינות במיוחד.
- $O(log\ n)$ - זמן לוגריתמי.
- $O(n)$ - זמן לינארי.
- $O(n^c)$ - עבור $c \in \mathbb{N}$ - זמן פולינומי.
- $O(2^{cn})$ - עבור $c \in \mathbb{N}$ - זמן אקספוננציאלי.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע. בעיות שניתנות לפיתרון בזמן קבוע (כלומר אינו תלוי בגודל הקלט). לרוב, אינן מעניינות במיוחד.
- $O(\log n)$ - זמן לוגריתמי.
- $O(n)$ - זמן לינארי.
- $O(n^c)$ - עבור $c \in \mathbb{N}$ - זמן פולינומי.
- $O(2^{cn})$ - עבור $c \in \mathbb{N}$ - זמן אקספוננציאלי.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע. בעיות שניתנות לפיתרון בזמן קבוע (כלומר אינו תלוי בגודל הקלט). לרוב, אינן מעניינות במיוחד.
- $O(log\ n)$ - זמן לוגריתמי.
- $O(n)$ - זמן לינארי.
- $O(n^c)$ - עבור $c \in \mathbb{N}$ - זמן פולינומי.
- $O(2^{cn})$ - עבור $c \in \mathbb{N}$ - זמן אקספוננציאלי. זמן זה נחשב ללא יעיל במיוחד, כאשר הזמן הדרוש לפתרון של קלטים קטנים יחסית יכול להיות עצום.

זמני ריצה

להלן כמה דוגמאות לזמני ריצה, כאשר n מסמל את גודל המופע:

- $O(1)$ - זמן ריצה קבוע. בעיות שניתנות לפיתרון בזמן קבוע (כלומר אינו תלוי בגודל הקלט). לרוב, אינן מעניינות במיוחד.
- $O(log\ n)$ - זמן לוגריתמי.
- $O(n)$ - זמן לינארי.
- $O(n^c)$ - עבור $c \in \mathbb{N}$ - זמן פולינומי.
- $O(2^{cn})$ - עבור $c \in \mathbb{N}$ - זמן אקספוננציאלי. זמן זה נחשב ללא יעיל במיוחד, כאשר הזמן הדרוש לפתרון של קלטים קטנים יחסית יכול להיות עצום.
- *הביטו בטבלה 2.1 בספר (עמוד 39)

כתיבת אלגוריתם בשלושה חלקים

בבואנו לכתוב אלגוריתם הפותר בעיה נתייחס לשלושה אספקטים:

כתיבת אלגוריתם בשלושה חלקים

בבואנו לכתוב אלגוריתם הפותר בעיה נתייחס לשלושה אספקטים:

1. תיאור דרך כללית להגעה לפתרון עבור מופע כלשהו של הבעיה (ע"י תיאור אלגוריתם).

כתיבת אלגוריתם בשלושה חלקים

בבואנו לכתוב אלגוריתם הפותר בעיה נתייחס לשלושה אספקטים:

1. תיאור דרך כללית להגעה לפתרון עבור מופע כלשהו של הבעיה (ע"י תיאור אלגוריתם).
2. הוכחת נכונות הדרך שתיארנו (על ידי הוכחה פורמאלית).

כתיבת אלגוריתם בשלושה חלקים

בבואנו לכתוב אלגוריתם הפותר בעיה נתייחס לשלושה אספקטים:

1. תיאור דרך כללית להגעה לפתרון עבור מופע כלשהו של הבעיה (ע"י תיאור אלגוריתם).
2. הוכחת נכונות הדרך שתיארנו (על ידי הוכחה פורמאלית).
3. ניתוח זמן הריצה הדרוש לקבלת הפתרון, בהתאם לדרך שתיארנו.

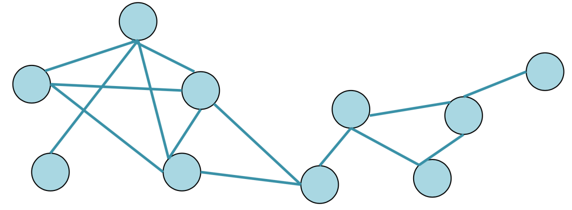
חלק II

מושגים בסיסיים מעולם הגרפים

מושגים בסיסיים

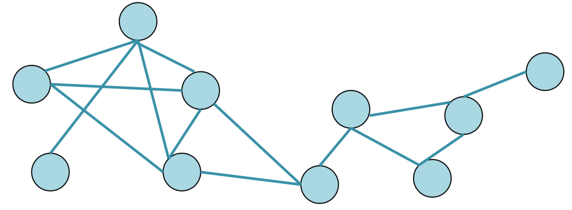
מושגים בסיסיים

- גרף לא מכוון הוא זוג $G = (V, E)$



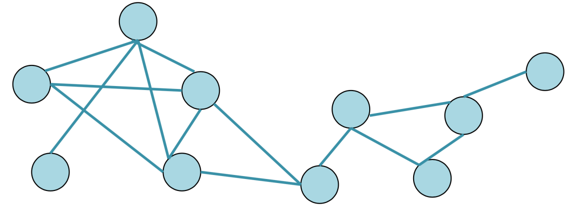
מושגים בסיסיים

- גרף לא מכוון הוא זוג $G = (V, E)$
  - V היא קבוצה סופית של איברים הנקראים צמתים או קודקודים.



מושגים בסיסיים

- גרף לא מכוון הוא זוג $G = (V, E)$
  - V היא קבוצה סופית של איברים הנקראים צמתים או קודקודים.
  - E היא קבוצה של זוגות לא סדורים מתוך V הנקראים קשתות.



מושגים בסיסיים

מושגים בסיסיים

- **שכן -** צומת v הוא שכן של צומת u, אם קיימת קשת בגרף $\{u, v\}$, היחס כמובן סימטרי.

מושגים בסיסיים

- **שכן -** צומת v הוא שכן של צומת u, אם קיימת קשת בגרף $\{u, v\}$, היחס כמובן סימטרי.
- **דרגה -** הדרגה של צומת u שווה למספר השכנים של u ומסומנת $deg(u)$.

מושגים בסיסיים

- **שכן -** צומת v הוא שכן של צומת u, אם קיימת קשת בגרף $\{u, v\}$, היחס כמובן סימטרי.
- **דרגה -** הדרגה של צומת u שווה למספר השכנים של u ומסומנת $deg(u)$.
- **מסלול -** מסלול בגרף הוא סדרה של צמתים $(v_1, \ldots, v_k)$ שבה כל זוג (v_i, v_{i+1}) היא קשת בגרף.

מושגים בסיסיים

מושגים בסיסיים

- **אורך מסלול -** מספר הקשתות במסלול. אורך המסלול $(v_1, \ldots, v_k)$ הוא $k - 1$

מושגים בסיסיים

- **אורך מסלול -** מספר הקשתות במסלול. אורך המסלול $(v_1, \ldots, v_k)$ הוא $k - 1$
- **מרחק בין צמתים -** אורך המסלול הקצר ביותר המחבר אותם. אם אין מסלול כזה, המרחק מוגדר להיות אינסוף.

מושגים בסיסיים

- **אורך מסלול -** מספר הקשתות במסלול. אורך המסלול $(v_1, \ldots, v_k)$ הוא $k-1$
- **מרחק בין צמתים -** אורך המסלול הקצר ביותר המחבר אותם. אם אין מסלול כזה, המרחק מוגדר להיות אינסוף.
- **מסלול פשוט-** מסלול בו שום צומת איננו מופיע יותר מפעם אחת.

מושגים בסיסיים

מושגים בסיסיים

- **מעגל -** מסלול המכיל לפחות שלושה צמתים שונים ובו הצומת הראשון והאחרון זהים, כל שאר הצמתים במסלול מופיעים בדיוק פעם אחת.

מושגים בסיסיים

- **מעגל -** מסלול המכיל לפחות שלושה צמתים שונים ובו הצומת הראשון והאחרון זהים, כל שאר הצמתים במסלול מופיעים בדיוק פעם אחת.
- **גרף קשיר -** גרף נקרא קשיר אם בין כל זוג צמתים בגרף קיים מסלול המקשר אותם.

מושגים בסיסיים

- **מעגל -** מסלול המכיל לפחות שלושה צמתים שונים ובו הצומת הראשון והאחרון זהים, כל שאר הצמתים במסלול מופיעים בדיוק פעם אחת.
- **גרף קשיר -** גרף נקרא קשיר אם בין כל זוג צמתים בגרף קיים מסלול המקשר אותם.
- **תת-גרף-** $H = (V', E')$ נקרא תת-גרף של הגרף $G = (V, E)$, אם H הוא גרף וגם, $V' \subseteq V \quad E' \subseteq E,$

מושגים בסיסיים

מושגים בסיסיים

- **עץ** – גרף קשיר ללא מעגלים.

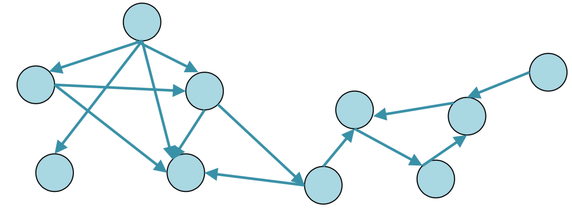
מושגים בסיסיים

- **עץ –** גרף קשיר ללא מעגלים.
- **עץ מושרש –** עץ עם שורש מיוחד הנקרא שורש. צומת v הוא צאצא של צומת u אם u מופיע על המסלול הפשוט (היחיד) המחבר את v לשורש.

מושגים בסיסיים - גרפים מכוונים

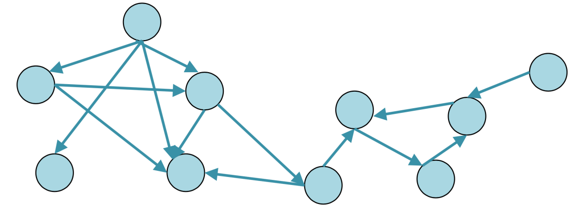
מושגים בסיסיים - גרפים מכוונים

- **גרף מכוון** הוא זוג $G = (V, E)$



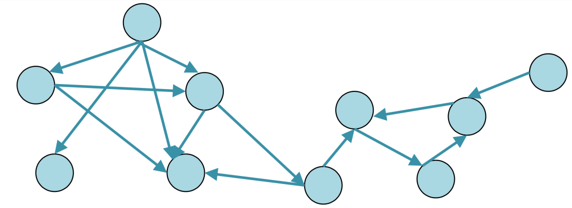
מושגים בסיסיים - גרפים מכוונים

- **גרף מכוון** הוא זוג $G = (V, E)$
  - V היא קבוצה סופית של איברים הנקראים צמתים או קודקודים.



מושגים בסיסיים - גרפים מכוונים

- **גרף מכוון** הוא זוג $G = (V, E)$
  - V היא קבוצה סופית של איברים הנקראים צמתים או קודקודים.
  - E היא קבוצה של זוגות **סדורים** מתוך V הנקראים קשתות.



מושגים בסיסיים - גרפים מכוונים

מושגים בסיסיים - גרפים מכוונים

- **דרגת כניסה של צומת** v **–** מספר הקשתות הנכנסות ל v .

מושגים בסיסיים - גרפים מכוונים

- **דרגת כניסה של צומת** v **–** מספר הקשתות הנכנסות ל v .
- **דרגת יציאה של צומת** v **–** מספר הקשתות היוצאות מ v .

מושגים בסיסיים - גרפים מכוונים

- **דרגת כניסה של צומת** v **–** מספר הקשתות הנכנסות ל v .
- **דרגת יציאה של צומת** v **–** מספר הקשתות היוצאות מ v .
- **DAG** גרף מכוון ללא מעגלים מכוונים בגרף.

מושגים בסיסיים - גרפים מכוונים

- **דרגת כניסה של צומת** *v* **–** מספר הקשתות הנכנסות ל *v* .
- **דרגת יציאה של צומת** *v* **–** מספר הקשתות היוצאות מ *v* .
- **DAG** גרף מכוון ללא מעגלים מכוונים בגרף.
- **קשירות היטב -** *u* ו- *v* קשירים היטב, אם קיים מסלול מכוון מ *u* ל *v* וכמו כן מ *v* ל *u* .

שאלה 1

שאלה 1

- נקרא למסלול ״מסלול פשוט קשתות״ אמ״מ אין בו קשת החוזרת פעמיים

שאלה 1

- נקרא למסלול ״מסלול פשוט קשתות״ אמ״מ אין בו קשת החוזרת פעמיים
- הוכח או הפרך ע״י מתן דוגמא נגדית

שאלה 1

- נקרא למסלול "מסלול פשוט קשתות" אמ"מ אין בו קשת החוזרת פעמיים
- הוכח או הפרך ע"י מתן דוגמא נגדית
  - האם כל מסלול פשוט הוא מסלול פשוט קשתות?

שאלה 1

- נקרא למסלול "מסלול פשוט קשתות" אמ"מ אין בו קשת החוזרת פעמיים
- הוכח או הפרך ע"י מתן דוגמא נגדית
  - האם כל מסלול פשוט הוא מסלול פשוט קשתות?
    - נכון. נניח בשלילה שלא. אזי במסלול קיימת קשת המופיעה פעמיים במסלול, ולכן הקודקדים הסמוכים לה מופעים פעמים, והמסלול אינו פשוט, בסתירה להנחה.

שאלה 1

- נקרא למסלול "מסלול פשוט קשתות" אמ"מ אין בו קשת החוזרת פעמיים
- הוכח או הפרך ע"י מתן דוגמא נגדית
  - האם כל מסלול פשוט הוא מסלול פשוט קשתות?
    - נכון. נניח בשלילה שלא. אזי במסלול קיימת קשת המופיעה פעמיים במסלול, ולכן הקודקדים הסמוכים לה מופעים פעמים, והמסלול אינו פשוט, בסתירה להנחה.
  - האם כל מסלול פשוט-קשתות הוא מסלול פשוט?

שאלה 1

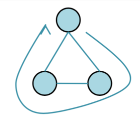
- נקרא למסלול "מסלול פשוט קשתות" אמ"מ אין בו קשת החוזרת פעמיים
- הוכח או הפרך ע"י מתן דוגמא נגדית
  - האם כל מסלול פשוט הוא מסלול פשוט קשתות?
    - נכון. נניח בשלילה שלא. אזי במסלול קיימת קשת המופיעה פעמיים במסלול, ולכן הקודקדים הסמוכים לה מופעים פעמים, והמסלול אינו פשוט, בסתירה להנחה.
  - האם כל מסלול פשוט-קשתות הוא מסלול פשוט?
    - לא נכון. להלן דוגמא נגדית

שאלה 1

- נקרא למסלול "מסלול פשוט קשתות" אמ"מ אין בו קשת החוזרת פעמיים
- הוכח או הפרך ע"י מתן דוגמא נגדית
  - האם כל מסלול פשוט הוא מסלול פשוט קשתות?
    - נכון. נניח בשלילה שלא. אזי במסלול קיימת קשת המופיעה פעמיים במסלול, ולכן הקודקדים הסמוכים לה מופעים פעמים, והמסלול אינו פשוט, בסתירה להנחה.
  - האם כל מסלול פשוט-קשתות הוא מסלול פשוט?
    - לא נכון. להלן דוגמא נגדית

שאלה 1

- נקרא למסלול "מסלול פשוט קשתות" אמ"מ אין בו קשת החוזרת פעמיים
- הוכח או הפרך ע"י מתן דוגמא נגדית
  - האם כל מסלול פשוט הוא מסלול פשוט קשתות?
    - נכון. נניח בשלילה שלא. אזי במסלול קיימת קשת המופיעה פעמיים במסלול, ולכן הקודקדים הסמוכים לה מופעים פעמים, והמסלול אינו פשוט, בסתירה להנחה.
  - האם כל מסלול פשוט-קשתות הוא מסלול פשוט?
    - לא נכון. להלן דוגמא נגדית



גרפים

גרפים

- מבני נתונים לשמירת גרפים:

גרפים

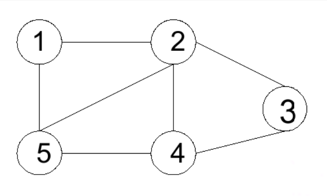
- מבני נתונים לשמירת גרפים:
  - מטריצת סמיכויות

גרפים

- מבני נתונים לשמירת גרפים:
  - מטריצת סמיכויות
  - רשימת סמיכויות

מטריצת סמיכויות - דוגמא

מטריצת סמיכויות - דוגמא



$$\begin{array}{c|ccccc|} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 1 & 0 & 0 & 1 \\ 2 & 1 & 0 & 1 & 1 & 1 \\ 3 & 0 & 1 & 0 & 1 & 0 \\ 4 & 0 & 1 & 1 & 0 & 1 \\ 5 & 1 & 1 & 0 & 1 & 0 \\ \hline \end{array}$$

מטריצת סמיכויות:

- מטריצה $A = (a_{ij})$ שמימדיה $|V| \times |V|$ וערכי איבריה:

$$a_{ij} = \begin{cases} 1, (i, j) \in E \\ 0, otherwise \end{cases}$$

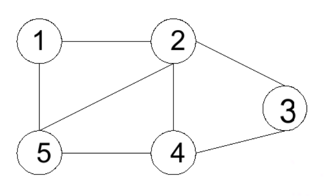
מטריצת סמיכויות:

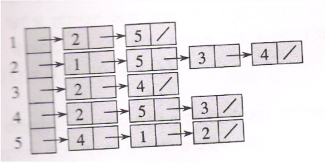
- מטריצה $A = (a_{ij})$ שמימדיה $|V| \times |V|$ וערכי איבריה:

$$a_{ij} = \begin{cases} 1, (i, j) \in E \\ 0, otherwise \end{cases}$$

- הייצוג ע"י מטריצת סמיכויות עשוי להיות עדיף כאשר הגרף צפוף או כאשר נדרשת היכולת לגלות במהירות אם קיימת קשת המחברת שני קודקודים נתונים.

רשימת סמיכויות - דוגמא





רשימת סמיכויות

רשימת סמיכויות

- הייצוג ע"י רשימות סמיכות של גרף $G = (V, E)$ מורכב ממערך Adj של $|V|$ רשימות, אחת עבור כל קדקוד ב- V.

רשימת סמיכויות

- הייצוג ע"י רשימות סמיכות של גרף $G = (V, E)$ מורכב ממערך Adj של $|V|$ רשימות, אחת עבור כל קדקוד ב- V.
- עבור כל u מ- V רשימת $Adj[u]$ מכילה מצביעים לכל הקודקודים v שעבורם קיימת קשת (u, v) .

רשימת סמיכויות

- הייצוג ע"י רשימות סמיכות של גרף $G = (V, E)$ מורכב ממערך Adj של $|V|$ רשימות, אחת עבור כל קדקוד ב- V.
- עבור כל u מ- V רשימת $Adj[u]$ מכילה מצביעים לכל הקודקודים v שעבורם קיימת קשת (u, v) .
- בד"כ קודקודים בכל רשימת סמיכות מאוחסנים בסדר שרירותי.

עלות מקום לייצוגים

עלות מקום לייצוגים

- עלות המקום כאשר מייצגים גרף בעזרת מטריצת סמיכויות: $\Theta(|V|^2)$

עלות מקום לייצוגים

- עלות המקום כאשר מייצגים גרף בעזרת מטריצת סמיכויות: $\Theta(|V|^2)$
- ובייצוג על ידי רשימת סמיכויות: $\Theta(|V| + |E|)$

גרפים

- תרגיל: נתון גרף מכוון G

שאלה	מטריצה(זמן ריצה)	רשימה (זמן ריצה)
(u,v) is in E ?		
קיימת קשת כלשהי בגרף?		
בהינתן צומת v מצא את כל שכניו		

גרפים

- תרגיל: נתון גרף מכוון G

שאלה	מטריצה(זמן ריצה)	רשימה (זמן ריצה)
(u,v) is in E ?	O(1)	O(|V|)
קיימת קשת כלשהי בגרף?	O(|V|*|V|)	O(|V|)
בהינתן צומת v מצא את כל שכניו	Θ(|V|)	O(|V|)

טכניקות בסיסיות ושימושיות מאוד

טכניקות בסיסיות ושימושיות מאוד

- אינדוקציה

טכניקות בסיסיות ושימושיות מאוד

- אינדוקציה
- עקרון שובך היונים: אם מכניסים $n+1$ ל n שובכים אז קיים שובך אחד בו לפחות 2 יונים.

טכניקות בסיסיות ושימושיות מאוד

- אינדוקציה
- עקרון שובך היונים: אם מכניסים $n+1$ ל n שובכים אז קיים שובך אחד בו לפחות 2 יונים.

טכניקות בסיסיות ושימושיות מאוד

- אינדוקציה
- עקרון שובך היונים: אם מכניסים $n+1$ ל n שובכים אז קיים שובך אחד בו לפחות 2 יונים.
  שימוש: כל מסלול בגרף בעלת $n+1$ קודקודים מכילה מעגל.

טכניקות בסיסיות ושימושיות מאוד

- אינדוקציה
- עקרון שובך היונים: אם מכניסים $n+1$ ל n שובכים אז קיים שובך אחד בו לפחות 2 יונים.
  שימוש: כל מסלול בגרף בעלת $n+1$ קודקודים מכילה מעגל.
- בגרף לא מכוון:

$$\sum_{v \in V} deg(v) = 2|E|$$

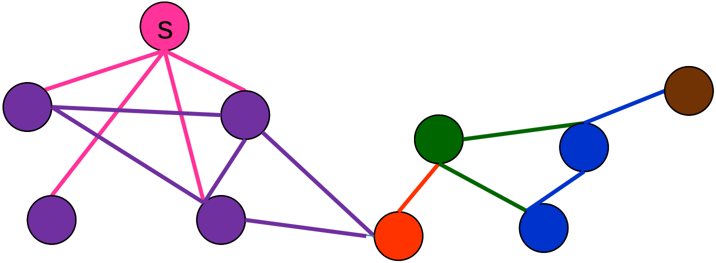
שאלה 2

הראה שבכל גרף בלתי מכוון וקשיר $|E| \geq |V| - 1$

שאלה 3

הראו שבכל DAG (גרף מכוון חסר מעגלים) יש קודקוד מקור (קודקוד שדרגת הכניסה אליו היא 0)

BFS

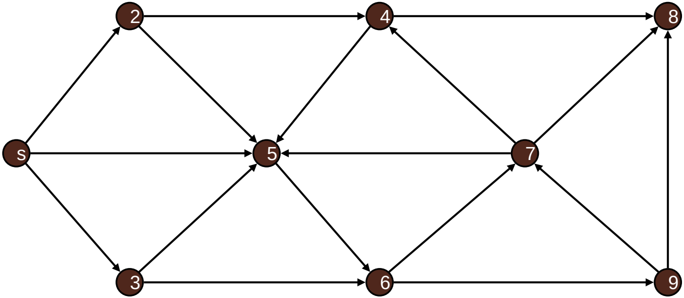


L_0 L_3

L_1 L_4

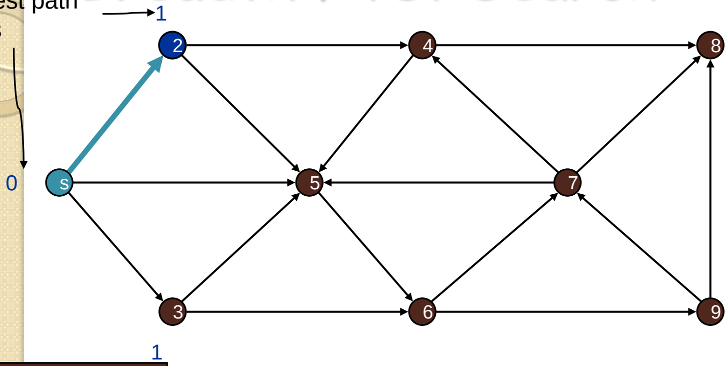
L_2 L_5

Breadth First Search



Breadth First Search

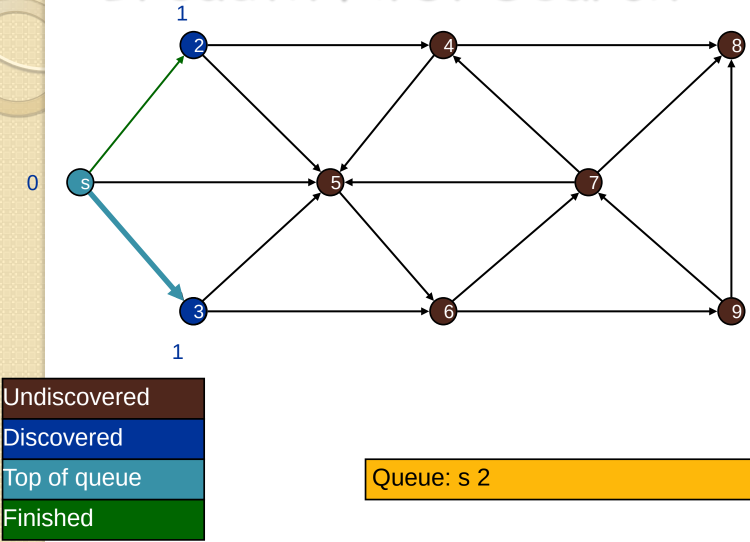
Shortest path from s



Undiscovered
Discovered
Top of queue
Finished

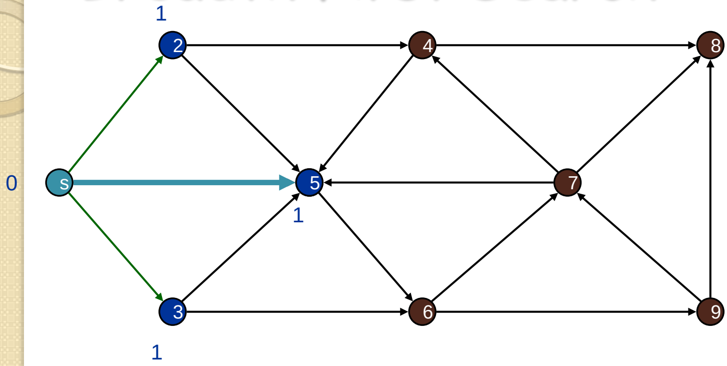
Queue: s

Breadth First Search



Queue: s 2

Breadth First Search



Undiscovered

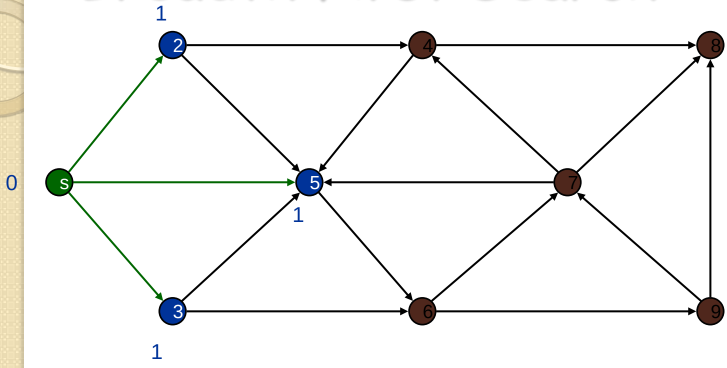
Discovered

Top of queue

Finished

Queue: s 2 3

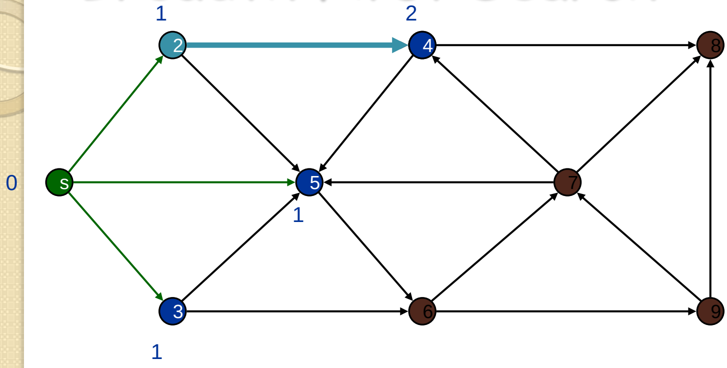
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 2 3 5

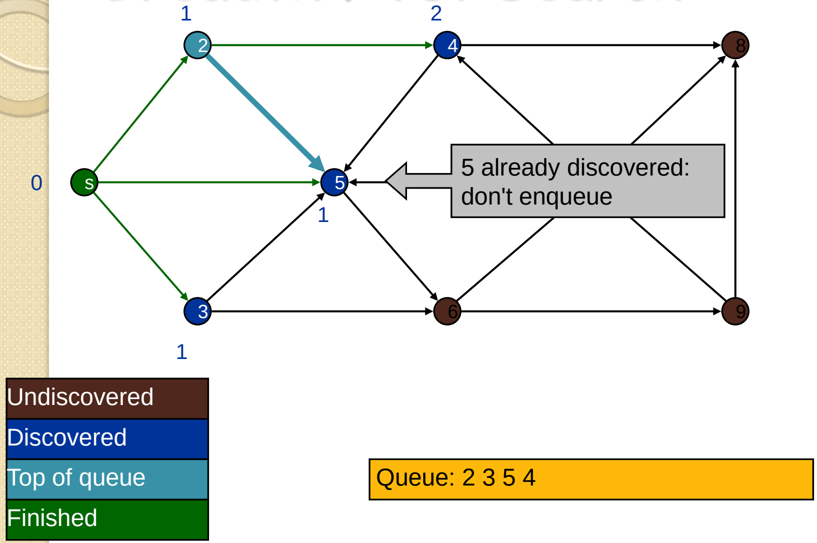
Breadth First Search



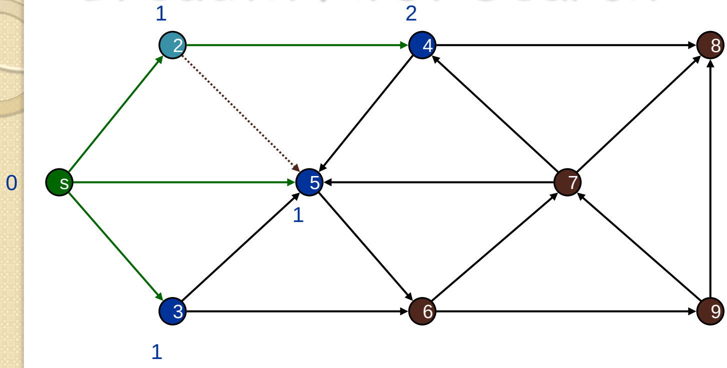
Undiscovered
Discovered
Top of queue
Finished

Queue: 2 3 5

Breadth First Search



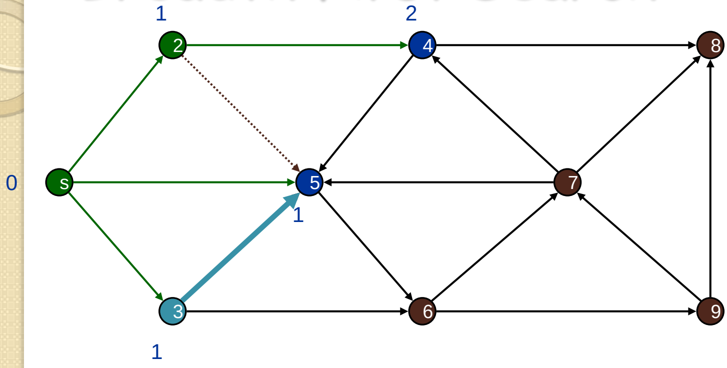
Breadth First Search



Undiscovered	
Discovered	
Top of queue	
Finished	

Queue: 2 3 5 4

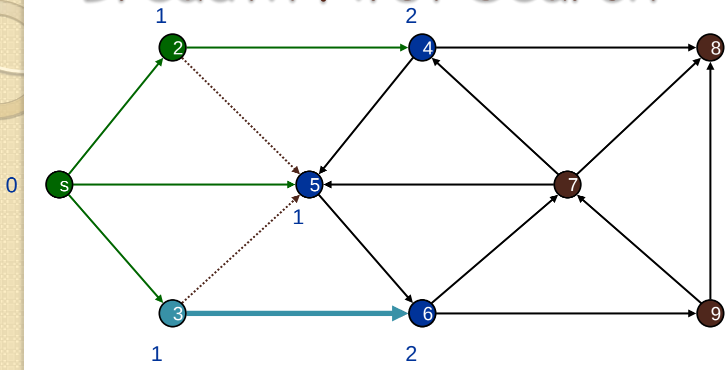
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 3 5 4

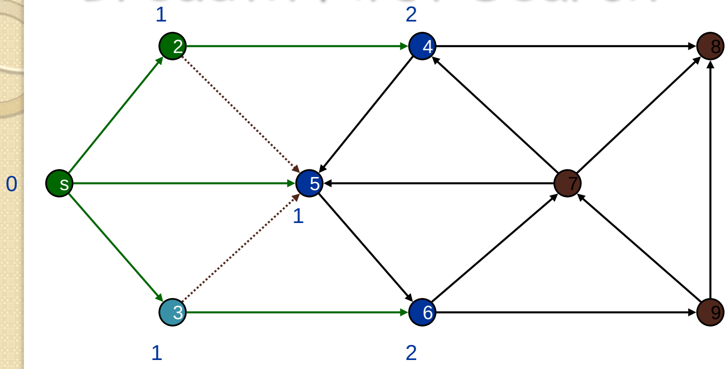
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 3 5 4

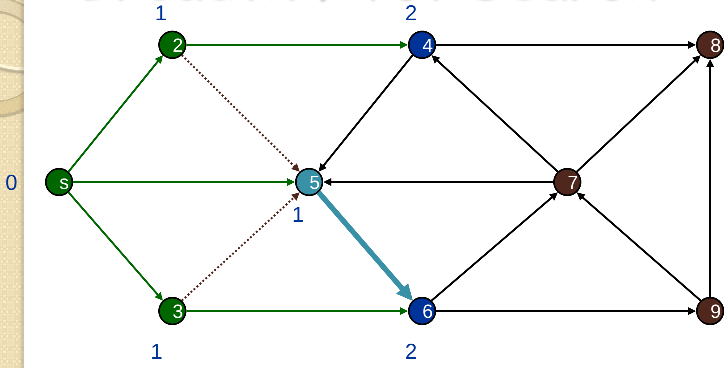
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 3 5 4 6

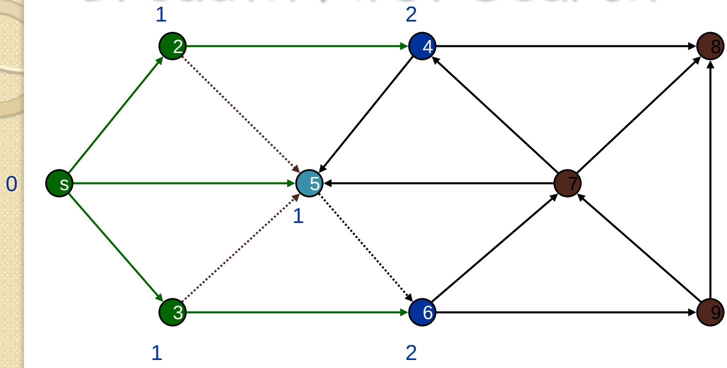
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 5 4 6

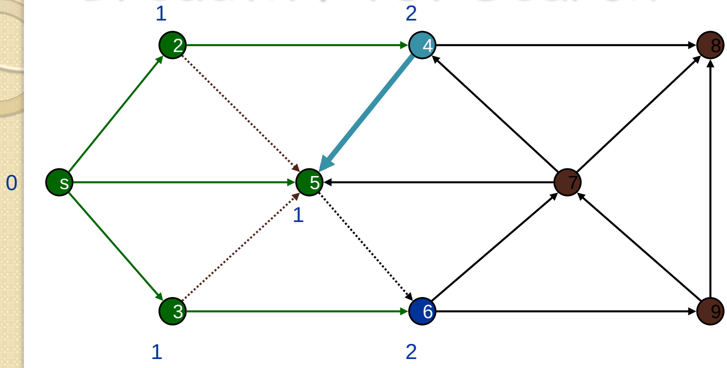
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 5 4 6

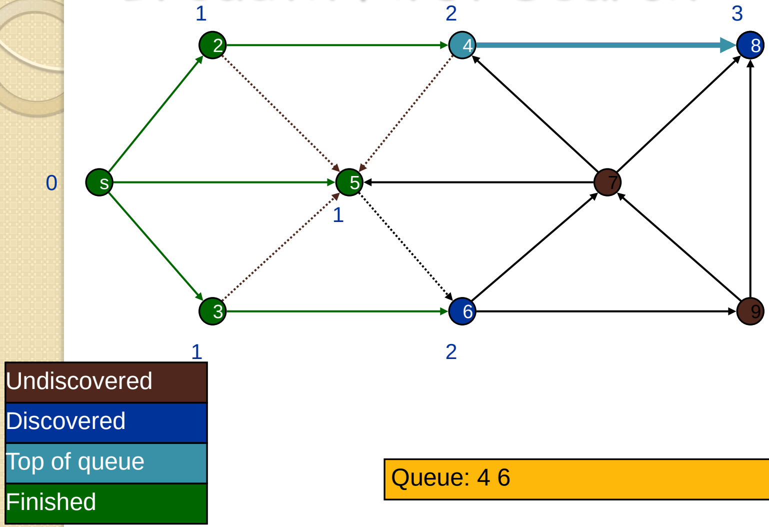
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

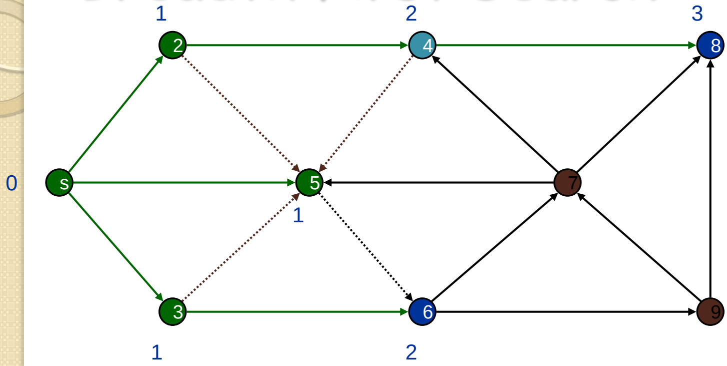
Queue: 4 6

Breadth First Search



Queue: 4 6

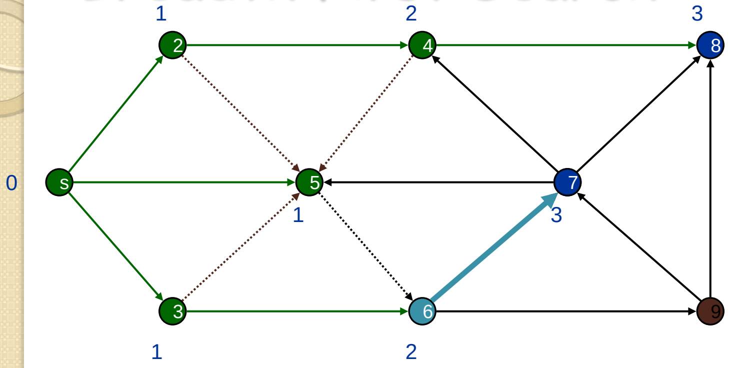
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 4 6 8

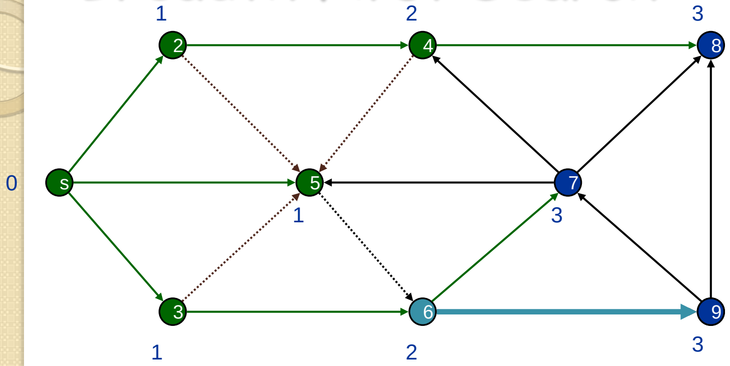
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 6 8

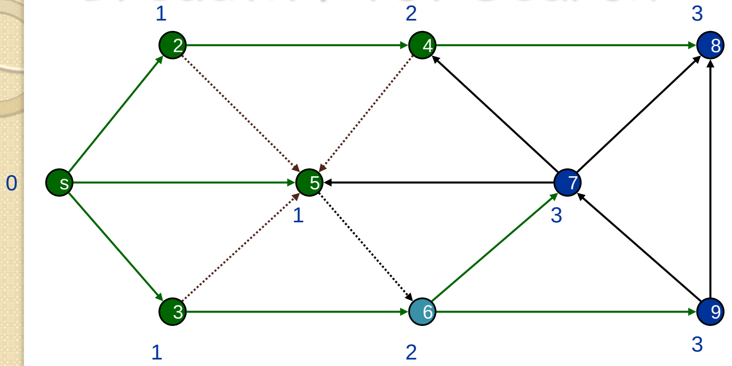
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 6 8 7

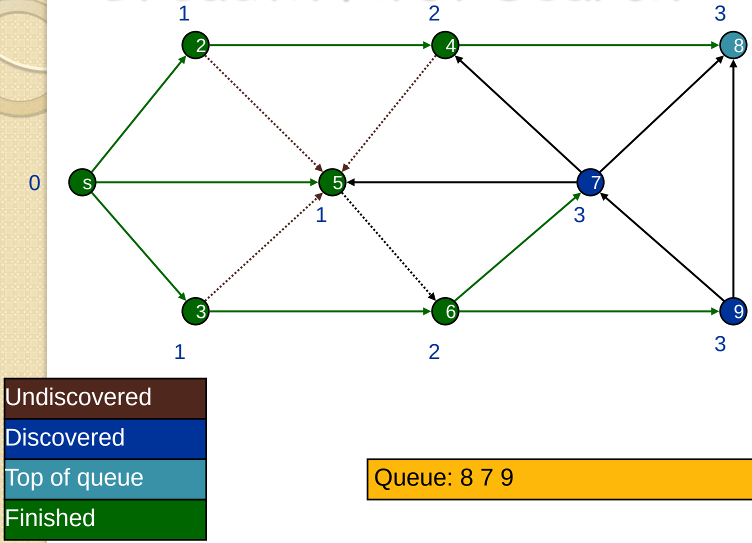
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

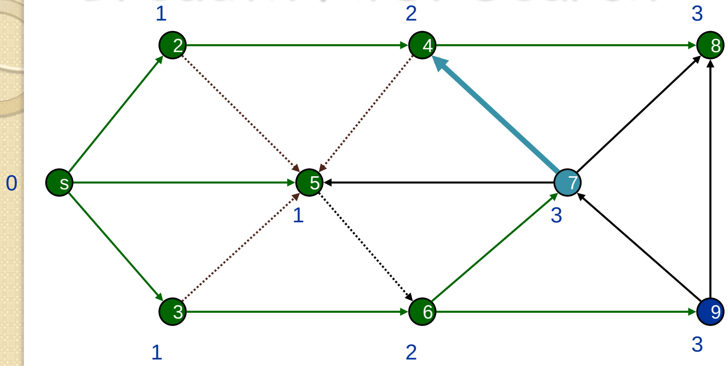
Queue: 6 8 7 9

Breadth First Search



Queue: 8 7 9

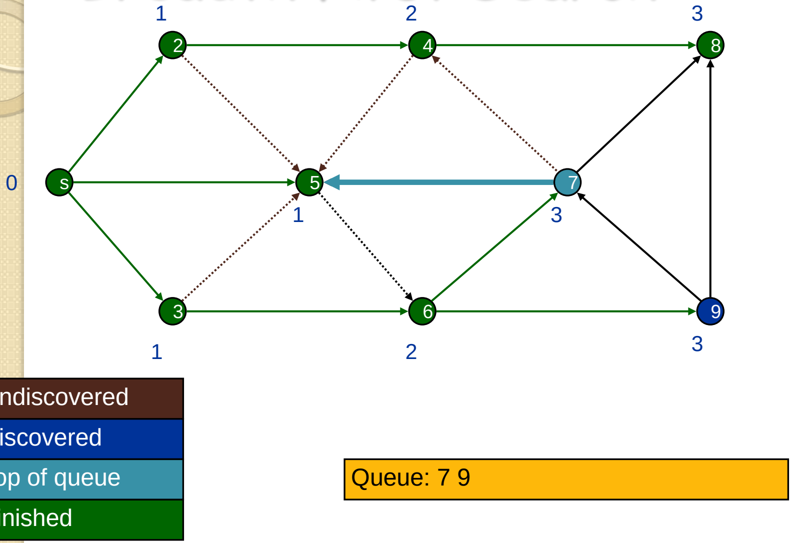
Breadth First Search



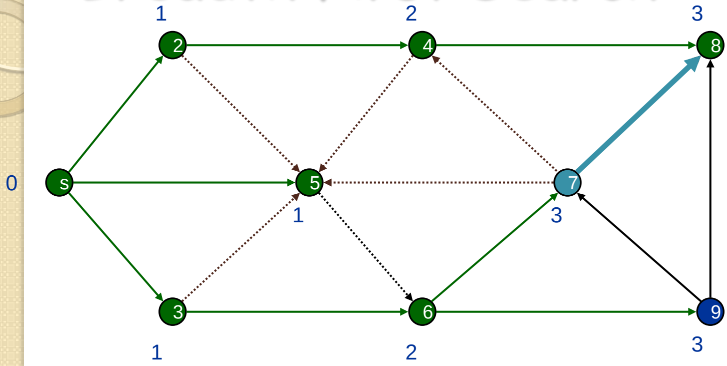
Undiscovered
Discovered
Top of queue
Finished

Queue: 7 9

Breadth First Search



Breadth First Search



Undiscovered

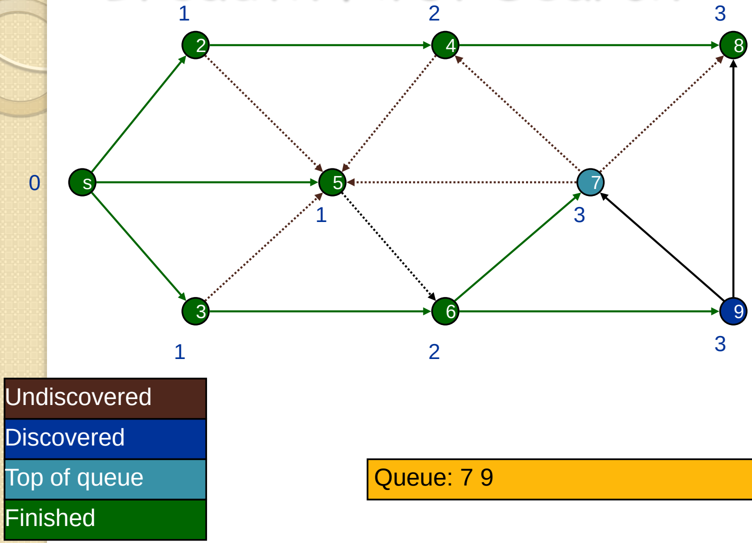
Discovered

Top of queue

Finished

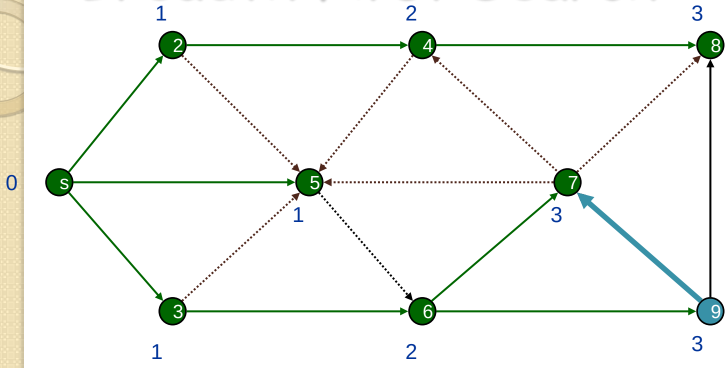
Queue: 7 9

Breadth First Search



Queue: 7 9

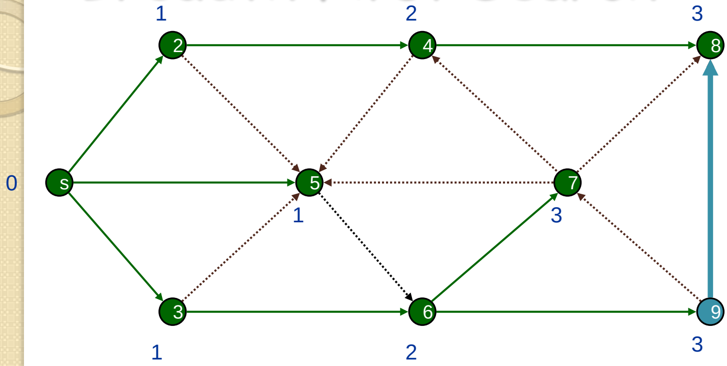
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue: 9

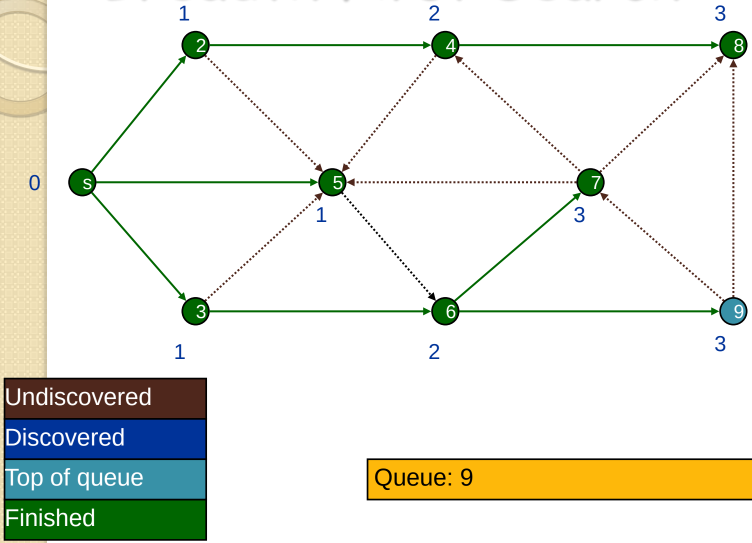
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

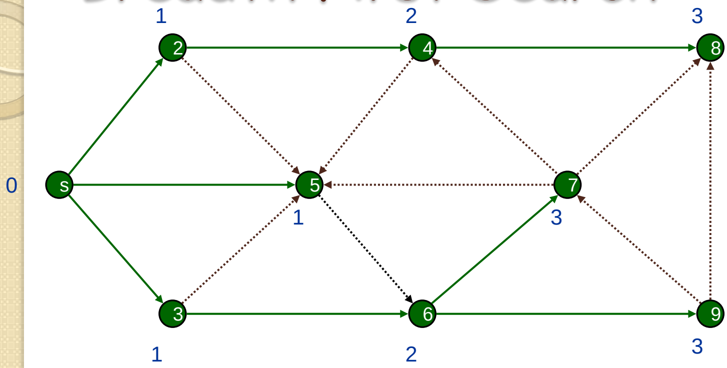
Queue: 9

Breadth First Search



Queue: 9

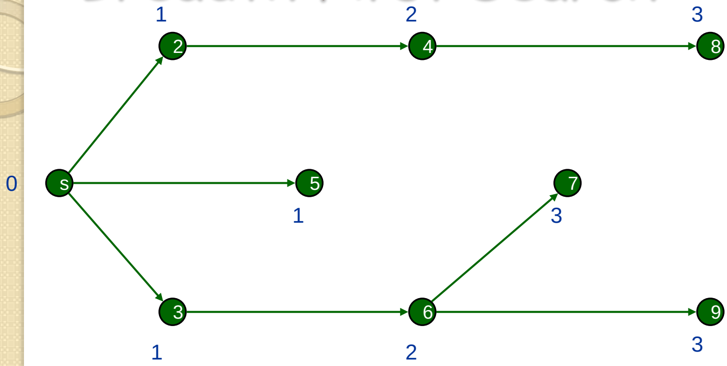
Breadth First Search



Undiscovered
Discovered
Top of queue
Finished

Queue:

Breadth First Search



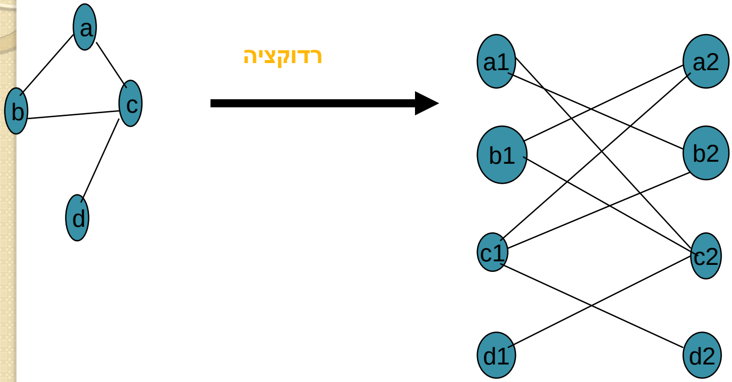
Level Graph

טכניקת הוכחה - רדוקציה

תרגיל

נתון גרף לא מכוון, קשיר וסופי $G(V,E)$, וצומת s ב V. תן אלגוריתם שמוצא עבור כל צומת v ב V, את אורך המסלול הקצר ביותר מ s ל v המכיל מספר זוגי של קשתות, או ∞ אם אין מסלול כזה.

דוגמא



פתרון

- נפתור ע"י רדוקציה. עקרון הרדוקציה: צמצום של בעיה חדשה לא ידועה לבעיה שפתרונה כבר ידוע.
- בהינתן הגרף $G=(V,E)$ נבנה ממנו גרף חדש $G'=(V',E')$ באופן הבא: מכל צומת v ניצור v1, v2. מכל קשת (a,b) ניצור שתי קשתות חדשות (a1,b2), (a2,b1). G' גרף דו-צדדי.
- נריץ BFS על G' החל מהצומת s1.
סימון: $\lambda(v1)$ את הערך בסיום ריצת ה BFS עבור כל צומת v1 השייך ל V1.

פתרון-המשך

- לכל קודקוד v1: החזר את $\lambda(v1)$.
- הוכחת נכונות:
  טענה1 - $\lambda(v1)$ הוא אורך המסלול הזוגי הקצר ביותר ב G בין s ל v.
  לא נוכיח טענה זו ישירות - אלא נוכיח טענה קלה יותר להוכחה.

נסמן –

- $\delta(s,t)$ – מרחק זוגי בין s ל-t ב-G.
- $\delta'(s_1,t_1)$ – מרחק בין s_1 ל-t_1 ב-G'.

- טענה 2:
א. יש מסלול באורך k בין $s1$ ל $v1$ ב G', אזי:
k זוגי, וגם יש מסלול באורך k בין s ל v ב G.
ב. יש מסלול באורך זוגי k בין s ל v ב G, אזי יש מסלול באורך k בין $s1$ ל $v1$ ב G'.

- שאלה:
מדוע טענה 2 מוכיחה את טענה 1?
תשובה:
- מסעיף א' בטענה 2 נסיק $\delta(s,t) \leq \delta'(s_1,t_1)$
ומסעיף ב' נסיק $\delta'(s_1,t_1) \leq \delta(s,t)$ ולכן יש שיוויון.

הוכחת טענה 2

- הוכחת א':
  - נביט במסלול באורך k בין s1 ל v1 ב G'.
  $P' = s1\text{-}a2\text{-}b1\text{-}c2\text{-}\ldots\text{-}t1$
  אם נמחק את כל האינדקסים נקבל את המסלול P
  $P = s\text{-}a\text{-}b\text{-}c\text{-}\ldots\text{-}t$ זהו מסלול חוקי בין s ל t ע"פ הצורה שבה בנינו את G'. כמו כן, אורך המסלול הזה כאורך המסלול P' שכן שינינו רק את שמות הקודקודים במסלול.
  -

הוכחת טענה 2

- הוכחת א':
 - המסלול P' באורך זוגי: זאת משום שהוא מתחיל בקודקוד עם אינדקס 1, נע לקודקוד עם אינדקס 2, חוזר לאינדקס 1, וכך נע לסירוגין ומסתיים באינדקס 1. (ובאופן פורמלי באינדוקציה על האי-זוגיים k, שאין מסלול באורך k בין $s1$ לקודקוד המסומן "1")

- הוכחת ב':
  - יהי $P=s-x-y-z-...-t$ מסלול באורך זוגי בין s ל v ב G. ע"פ האופן שבו בנינו את G' קיים בו המסלול: $P'=s1-x2-y1-z2-...-t1$. אורכו זהה לאורך המסלול P ולכן באורך זוגי. כמו כן, המסלול אכן מסתיים באינדקס 1 שכן אורכו זוגי.
- סיבוכיות: יצירת G' דורשת מעבר על כל הצמתים ב G פעמיים , ועל כל קשת של G פעמיים. לכן $\theta(|V|+|E|)$. הרצת BFS על G',
  $O(|E'|+|V'|)=O(|E|+|V|)$
  סה"כ: $\theta(|E|)$ (כי הגרף קשיר ולכן $|E|=\Omega(|V|)$).

- **תרגיל:**

  כתוב אלגוריתם יעיל ככל שתוכל אשר מקבל כקלט גרף מכוון $G=(V,E)$ עם פונקציית משקל $w:E\rightarrow\{1,2\}$ ושני צמתים s,t ומוצא משקל מסלול קצר ביותר בין s ל t.

פתרון:

פתרון:

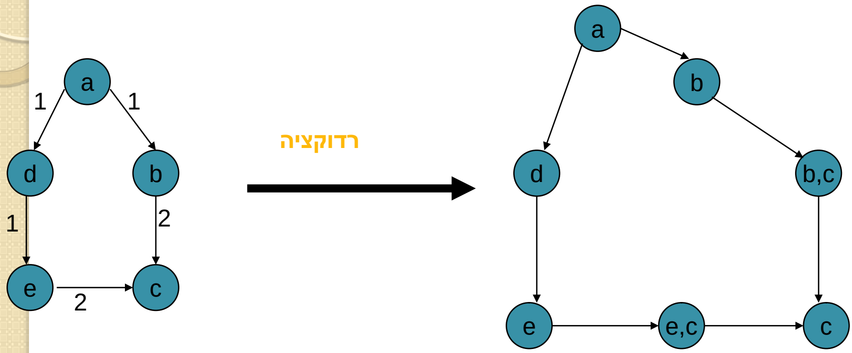
על ידי רדוקציה. נבנה גרף לא ממושקל בו נפעיל BFS מs ונחזיר בתור תשובה את d[t].

גרף הרדוקציה: G'=(V',E')

$$V' = V \cup \{v_e \mid e \in E \;\&\; w(e) = 2\}$$

$$E' = \{e \mid e \in E \;\&\; w(e) = 1\} \cup \{(x, v_{(x,y)}), (v_{(x,y)}, y) \mid (x,y) \in E \;\&\; w(x,y) = 2\}$$

דוגמא



- סיבוביות: ליניארית !

בניית גרף הרדוקציה $O(|V|+|E|)$

BFS על G' ליניארית בגודל G' שלינארי בגודל G

- הטענה המרכזית:

א. אם יש מסלול באורך k מ s ל t ב G אזי יש מסלול באורך k מs ל t ב G'

ב. להיפך: אם יש מסלול באורך k מ s ל t ב G' אזי יש מסלול באורך k מ s ל t ב G

הוכחה: תרגיל בית, נסו בעצמכם

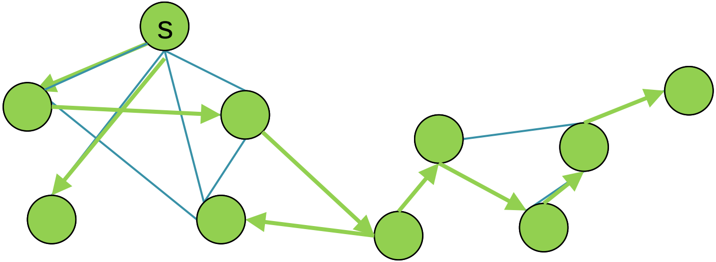
(בדומה להוכחת טענה 2 בתרגיל הקודם)

DFS –
Depth First Search Algorithm

- אלגוריתם רקורסיבי שמסמן צמתים.

```
DFS(v)
  mark v;
  for each neighbor u of v do
    if u is unmarked  then DFS(u)
```

DFS



DFS

- האלגוריתם מתחיל מצומת מקור s, ובכל פעם שהוא מוצא שכן שטרם ביקר בו,האלגוריתם ממשיך אליו רקורסיבית.
- אם בסוף הרקורסיה נשארו צמתים שלא בקרנו בהם - ניקח צומת כזה ונתחיל שוב
- כל הצמתית צבועים בלבן בתחילת האלג'

DFS
זמני כניסה ויציאה

Algorithm 3.1 DFS(u)

$b_u \leftarrow t; \quad t \leftarrow t+1$
Mark u as "Explored".
for each edge $\{u,v\}$ incident to u **do**
  if v is not marked "Explored" **then**
    Recursively invoke DFS(v)
$f_u \leftarrow t; \quad t \leftarrow t+1$

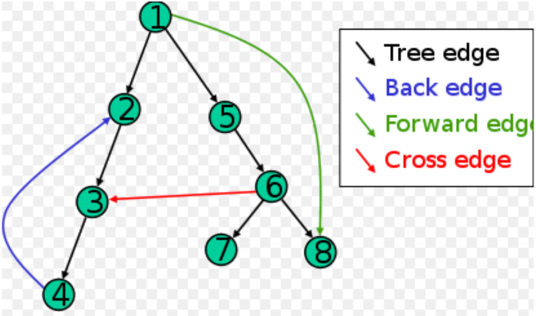
(עמוד 27 במדריך למידה)

- דוגמא על הלוח.

DFS

- נעבור על כל השכנים של u:
  - אם השכן v אליו מובילה הקשת (u,v) טרם התגלה, נעבור ל v והוא יהפוך לצומת הנוכחי. הקשת (u,v) תקרא קשת עץ.
  - עבור שכן v שכבר התגלה האלגוריתם לא עושה כלום, אך אנו נתייחס למקרים שונים:
    - אם טרם הושלם הטיפול ב v (צבעו אפור) נקרא לקשת (u,v) קשת אחורה – המצב הזה קורה כאשר u צאצא של v.
    - אם v צאצא של u בעץ – נקרא לקשת קשת קדימה
    - כל שאר הקשתות שאינן קשתות עץ הן קשתות חוצות

DFS



תכונות DFS

- עובר על כל הקודקודים
- ניתן להתייחס לעץ DFS כאל עץ מכוון, גם כשהגרף עליו רצים אינו מכוון
- DFS מוצא מעגלים
- DFS מסווג קשתות
  - עץ
  - אחורה
  - קדימה
  - חוצות
- סיבוכיות $O(|V| + |E|)$

שאלה 4

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 $G=T$

שאלה 4

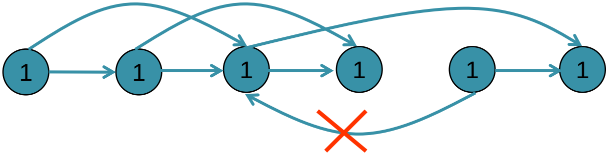
- **הטענה נכונה**.
- נניח בשלילה כי קיים גרף G כך שהרצת BFS וגם DFS עליו מקודקוד u נותנת T, אבל
  $G \neq T$
- מההנחה נובע כי קיימת קשת (x,y) ב G שאיננה ב T.
  נניח ללה"כ כי x נסרק לפני y, לכן (x,y) היא קשת אחורה ו x אב קדמון של y

שאלה 4

- **הטענה נכונה.**
- מכאן מרחק y בעץ ה DFS מהשורש גדול ממרחק x מהשורש + 2
- אבל כיוון שזהו גם עץ מסלולים קצרים ביותר וקיימת קשת בין x y וזוהי סתירה

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 $i < j$ 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 $d(v_1) < d(v_n) < \ldots < d(v_1)$, 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.

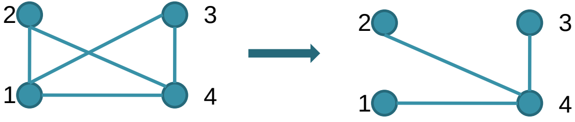
שאלה

- 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.