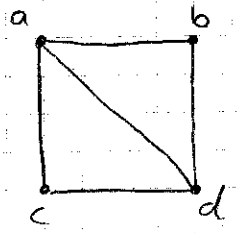
הערה: בהמשך נשתמש בגרפים המכוונים, נקרא לקורס "מבוא לתורת הגרפים"

גרף $G=(V,E)$

V קבוצה של קודקודים/נמתחים (node / vertex)

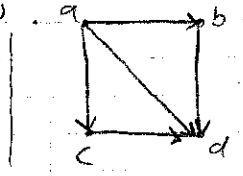
E קבוצה של קשתות (edge). קשת היא זוג (u,v) של נמתחים.



לדוגמא: בגרף יש 4 נמתחים a,b,c,d

ו-5 קשתות $(a,b), (a,d), (a,c)$
$(c,d), (b,d)$

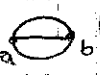
גרף מכוון directed - אם הקשתות מכוונות ע"י זוגות סדורים. לדוגמא, אם נתונים לגרף הקודם כמכוון הוא יהיה כך: ע"י הוספת הזוג הסדור (b,a) לקשתות. ניתן גם לדרוך קשת מ-b ל-a.



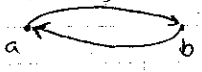
גרף לא מכוון undirected - הוא גרף בו אין משמעות לסדר בתוך הזוג הסדור של הקשתות.

לולאה loop הוא קשת הדומה ומצירת ל אותו הקודקוד. (a,a)

קשתות מקבילות הם 2 קשתות המחברות אותן 2 קודקודים (בגרף שאינו מכוון)



קשתות אינו מקבילות הן קשתות בגרף מכוון, המחברות 2 קודקודים אך בכיוונים שונים.



גרף פשוט: הוא גרף ללא לולאות וללא קשתות מקבילות.

לכל קשת (a,b) נקרא ל- a,b נקודות קצה (endpoints) ונגיד שהם שכנים בגרף.

זוג קשתות הן סמוכות אם יש להן נק' קצה משותפת.

בגרף מכוון, דרגת היציאה של צומת u הוא מס' הקשתות היוצאות מ-u $\leftarrow (u,v)$

דרגת הכניסה הוא מספר הקשתות הנכנסות ל u.

דרגת u הוא מספר הקשתות הסמוכות ל- u (נכנסות/יוצאות)

בגרף לא מכוון, יש רק דרגה.

קודקוד עם דרגה 0 נקרא מבודד (isolated), כלומר, אין צומת שאין לו אף קשת סמוכה.

בגרף מכוון, צומת עם דרגת כניסה 0 נקרא מקור source. צומת עם דרגת יציאה 0 נקרא בור sink.

תת גרף subgraph של $G=(V,E)$ הוא גרף $G'=(V',E')$ כך ש $V'\subseteq V$, $E'\subseteq E$ וכן $E'\subseteq V'\times V'$.

4.11.08 (2)

תת גרף מושרה induced הוא כזה שמכיל את כל הקשתות ב-E שמתבטאות בקודקודים ב-V.

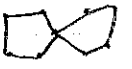
מסלול / מסלול path הוא סדרה של קודקודים $v_1, v_2, \dots, v_k$ כך שלכל i: $(v_i, v_{i+1}) \in E$ לדוגמה.

אורך המסלול הוא מספר הקשתות שבו. בפרט, $k=0$ הוא מסלול ריק.

מעגל cycle הוא מסלול ובו $v_k = v_1$ ניתן שבמעגל $k \geq 1$. לולאה היא מעגל באורך 1.

מסלול פשוט / מעגל פשוט simple הוא כשהוא אין חזרה על הקודקודים.

לדוגמה למעגל שאינו פשוט



בגרף לא מכוון, **קליק** clique הוא תת קבוצה של קודקודים, שכל אחד בקשת לכל, לדוגמה:

גרף שלם complete K_n, n הוא מספר הקודקודים ולכל זוג קודקודים מחוברת בקשת (לא מכוון). קיימים בו מקסימום $\binom{n}{2}$ קשתות, 0 קשתות = גרף ריק.

קבוצה בלתי תלויה independent set של קודקודים היא כזו שאין שום קשת שמחברת זוג קודקודים בקבוצה (לא מכוון). [כלומר שבין הקודקודים יש מסלולים אך בלי מקרה אין קשת ישירה].

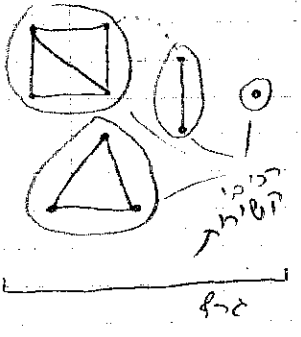
גרף דו-צדדי bipartite הוא גרף (שאינו נוטה שאינו מכוון) $G=(V,E)$ כך ש-V הוא איחוד זר של 2 תתי-קבוצות V_1, V_2 כאשר $E \subseteq V_1 \times V_2$.

קודקוד u **נגיש** reachable מקודקוד v, אם קיים מסלול מ-v ל-u. (מסלול מכוון - אם הגרף מכוון). כל קודקוד נגיש מעצמו.

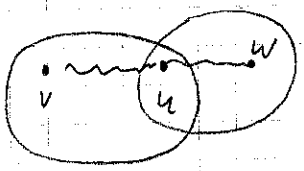
גרף לא מכוון הוא **קשיר** connected אם כל קודקוד נגיש מכל קודקוד.

אם הגרף הוא גרף לא קשיר, יש לו **רכיבי קשירות** Connected Components = תת קבוצה **מקסימלית** של קודקודים שתת הגרף המושרה שהם מגדירים הוא קשיר.

רכיב קשירות הוא תת גרף קשיר מקסימלי. כל גרף מתפרק לרכיבי קשירות **זרים** בקודקודים ובקשתות

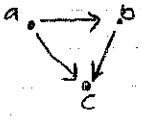


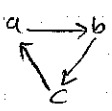
אלגוריתמים - שיעור 1



שרשור (concatenation).

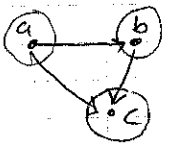
בגרף מכוון, נאמר שהיא קשיר חזק strongly connected, אם G דומה ונגיש מכל צמת, לכל מסלול מכוון.



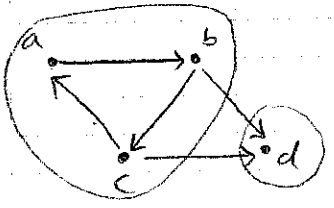
דוגמה לגרף לא קשיר חזק כי a לא נגיש מ- b. לעומת זאת,  כן קשיר חזק.

אם G (מכוון) לא קשיר חזק, אפשר לפרק אותו לרכיבי קשירות חזקה strongly connected components של צמתים שהיא קשירה חזק (או ליתר דיוק את הגרף שהיא משרה).

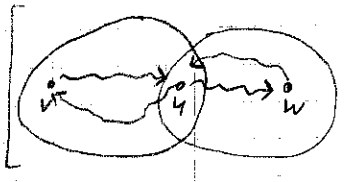
דוגמה: ①



②



שרשור מכוון



כאמור פירוק של V לקבוצות, בעיגול מקרה הלא-מכוון, מתבצע קשתות בין רקע"ים.

גרף אציקלי מכוון DAG הוא גרף מכוון מכוון חסר מעגלים. לדוגמה:

מכוון לא מכוון

גרף לא מכוון, חסר מעגלים נקשיר הוא עץ (tree). לדוגמה

אם הוא רק חסר מעגלים, הגרף נקרא יער (forest) וכל רכיב קשירות שלו הוא עץ. לדוגמה

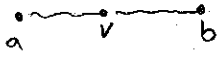
עלה הוא צומת בעץ / יער עם דרגה 1.

בעץ עם n קודקודים יש בדיוק n-1 קשתות.

ניתן לראות באינדוקציה על n: (לכל עץ יש לפחות עלה אחד).

n=2 קשת אחת.

אם נבונה ל n-1 צמתים ונתון עץ T עם n צמתים, נקח עלה V ונמחק אותו ואת הקשת היחידה שמחברת לו ונקבל את גרף T' עם n-1 צמתים, ללא מעגלים ונקשיר, ולכן T' עץ לפי הנחת האינדוקציה. ב- T' n-2 קשתות.



4.11.08 (4)

כל 2 מהתכונות הבאות גוררת את השלישית:

1. קשיר
2. חסר מעגלים
3. בעל $|V|-1$ קשתות.

בעץ כזה כל צומת במעגלים קיים מעגל אחד וייחד.

<u>עץ מכוון:</u>

עלקתים צומת אחד ← מגבר כשרש העץ (root) ואז מכוונים את הקשתות מהשורש והלאה או לכיוון השורש.

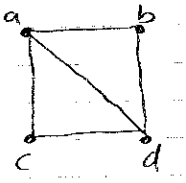
<u>ייצוג של גרפים:</u>

1) ע"י רשימת שכנות adjacency lists.
2) ע"י מטריצת שכנות adjacency matrix.

(1) <u>ע"י רשימת שכנות</u>

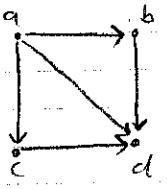
הגדרתית מוגדרים איך שהו לכל צומת יש מערך עם רשימה מקושרת של כל שכניו.

לדוגמה: I



$a \to (b \to c \to d)$
$b \to (a \to d)$
$c \to (a \to d)$
$d \to (a \to b \to c)$

II



$a \to (b \to c \to d)$
$b \to (d)$
$c \to (d)$
$d \to (\ )$

<u>יתרון:</u> חסרון במקום: $O(|V|+|E|)$.

<u>חסרון:</u> יקר להגיע לקשת ספציפית.

(2) <u>ע"י מטריצת שכנות</u>

מטריצה A בגודל $|V|$ שורות $|V|-1$ עמודות.
$A_{ij} = 1$ אם קיימת קשת המחברת צומת מס' i לצומת j, 0 אם לא.

לדוגמה: (תיאור של דוגמה II).

	a	b	c	d
a	0	1	1	1
b	0	0	0	1
c	0	0	0	1
d	0	0	0	0

<u>חסרון:</u> בגלל הכרחי, בעקר אם הגרף גדול sparse כלומר $E << \binom{N}{2}$

<u>יתרון:</u> גישה בזמן קבוע לקשת.

4.11.08 ⑤

BFS Breadth-First-Search (חיפוש לרוחב)

נתון גרף $G=(V,E)$, מכוון או לא ורשימת התחלה s. מטרתנו לבקר בדיוק פעם אחת בכל הקודקודים הנגישים מ-s ובסוף לחשב לכל קודקוד u את אורך המסלול (מס' הקשתות) הקצר ביותר מ-s ל-u.

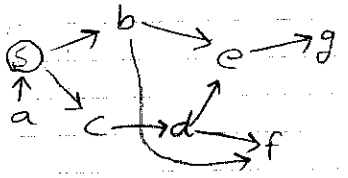
לכל קודקוד u מתאימים את השדות הבאים (פרט לבסיס):

i. $d[u]$ - האורך הקצר ביותר של מסלול מ-s אל u שהתגלה עד עתה.

ii. $\pi[u]$ - מצביע לקודקוד הקודם במסלול הקצר ביותר מ-s ל-u שהתגלה עד עתה.

האלגוריתם משתמש בתור Q, G נתון ע"י רשימות שכנות.

צבעים	~~s~~	~~b~~	~~c~~	~~e~~	~~d~~	~~a~~	~~g~~
$d[\cdot]$	0	1	1	2	2	2	3
$\pi[\cdot]$	nil	s	s	b	b	c	e



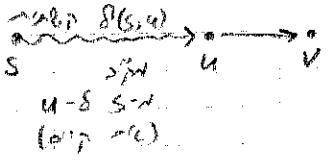
באמצעות מצביעי π ניתן לשחזר את המסלול הקצר ביותר לכל קודקוד.

$\delta(s,v)$ = אורך המסלול הקצר ביותר מ-s ל-v, או $+\infty$ אם אין מסלול כזה.

<u>נכונות</u>: לכל v $d[v]=\delta(s,v)$ בסיום האלגוריתם.

$d[s]=\delta(s,s)=0$ בכל האתחול (נכון עבור s)

1) אם $(u,v)\in E$ אז $\delta(s,v)\le\delta(s,u)+1$ (מקבל - מסלול הקצר ביותר)

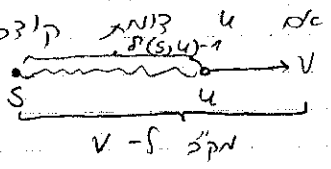


אם אין מסלול מ-s ל-u אי השוויון נכון ($\delta(s,u)\le\infty$).

אם u קודקוד קודם של v על מקבל מ-s ל-v אז $\delta(s,v)=\delta(s,u)+1$.

כי $\delta(s,v)\le\delta(s,u)+1$ ואם $\delta(s,v)<\delta(s,u)+1$ זה סתירה.

כלומר $\underbrace{\delta(s,v)-1}_{\text{מסלול מ-s ל-u}}<\underbrace{\delta(s,u)}_{\text{מקבל מ-s ל-u}}$



אם u קודקוד ל-v על מקבל מ-s אז חלק המסלול מ-s עד u גם הוא מקבל.

סיבוכיות האלגוריתם היא $O(|V|+|E|)$. לוגי סבירות אינטואיטיבית.