1 אלגוריתמים – דף נוסחאות. כללי

טענה: כל 2 מהתכונות הבאות גוררת את השלישית:
(1) גרף קשיר **(2)** חסר מעגלים **(3)** בעל $V-1$ קשתות.
(*) בעץ בין כל זוג צמתים קיים מסלול אחד ויחיד.
(*) בגרף עם מעגל אוילר אין גשר.

מסלול אוילר –מסלול בגרף אשר עובר על כל קשת בדיוק פעם אחת. **(1)** גרף לא מכוון וקשיר מכיל מסלול אוילר $\Leftrightarrow$ בגרף 2 צמתים מדרגה אי-זוגית והיתר מדרגה זוגית. **(2)** גרף מכוון וקשיר מכיל מסלול אוילר $\Leftrightarrow$ קיימים 2 צמתים w,v כך ש- $d_{in}(v)=d_{out}(v)+1$, $d_{in}(w)+1=d_{out}(w)$ ולכל שאר הצמתים $u\neq v,w$ מתקיים $d_{in}(u)=d_{out}(u)$.

מעגל אוליר - **(1)** גרף לא מכוון וקשיר מכיל מעגל אוילר $\Leftrightarrow$ כל הדרגות זוגיות. **(2)** גרף מכוון וקשיר מכיל מעגל אוילר $\Leftrightarrow$ לכל $v\in V$: $d_{in}(v)=d_{out}(v)$.
(*) בגרף לא מכוון, אם נוסיף לגרף עם מסלול אויילר קשת בין 2 הצמתים מדרגה אי-זוגית אז נקבל מעגל אויילר.

גשרים – גשר הוא קשת שהסרתה תהפוך את הגרף ללא קשיר. **(1)** בגרף בעל גשר אין מעגל אויילר. **(2)** קשת אינה גשר $\Leftrightarrow$ היא נמצאת על מעגל פשוט בגרף. גרף לא מכוון וקשיר ניתן לכיוון לגרף קשיר בחוזקה $\Leftrightarrow$ אין בו גשרים.

גרף דו"צ – גרף לא מכוון נקרא דו"צ $\Leftrightarrow$ אפשר לחלק את V ל-2 קב' זרות $V_1\cup V_2=V$ כך שאין $u_1\in V_1, u_2\in V_2$ המקיימים $\{u_1,u_2\}\in E$.

רשימת שכנויות – לכל צומת מצביע לרשימת השכנים שלו. יתרון: חסכוני במקום. חיסרון: הרבה זמן להגיע לקשת ספציפית.

מטריצ שכנויות – $A_{ij}=1\Leftrightarrow(i,j)\in E$. יתרון: קצת זמן להגיע לקשת ספציפית. חיסרון: מבזבז הרבה מקום.

2 BFS – חיפוש לרוחב. סיבוכיות: $O(E+V)$

טענה: אם u נגיש מ-s, BFS מבקר ב-u. (הוכחה באינדוקציה).
טענה: בכל רגע בתור Q נמצאים צמתים עם לכל היותר 2 ערכי d עוקבים בסדר עולה. (הוכחה באינדוקציה).
משפט: לכל צומת u, $d[u]=\delta(s,u)$. **הוכחה:** צמתים עם ערך אינסוף נוכיח כמעט ישירות ושאר הצמתים באינדוקציה.
(*) בגרף מכוון, כל קשת שה-BFS לא השתמש בה לבניית d, מחברת צומת בשכבה j כלשהי לצומת בשכבה $i\le j+1$. בגרף לא מכוון כל קשת מחברת צמתים באותה שכבה או בשכבות סמוכות.
(*) כדי לחלק את הגרף לקבוצות, נגדיר V_1 להיות כל הצמתים בשכבות הזוגיות ו-V_2 בשכבות האי זוגיות.
טענה: לכל קשת (u,v) מתקיים בסיום האלגוריתם: $\delta(s,v)\le\delta(s,u)+1$ (אם (u,v) במק"ב מ-s ל-v יש שיוויון).
צבעים: לבן-עדיין לא ביקרנו בו. אפור-נכנס לתור. שחור-יצא מהתור.

3 DFS – חיפוש לעומק. סיבוכיות: $O(E+V)$

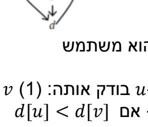
$d[u]$ - הזמן בו נתקלנו ב-u לראשונה. $f[u]$ - הזמן בו סיימנו את הטיפול ב-u. $\pi[u]$ - הצומת הקודם שגילה את u.

עקרון הקינון - לכל צומת u מתאים מרווח זמן חי $[d(u),f(u)]$. אם ל-2 צמתים יש מרווחים מקוננים אז הצומת עם הרווח הקטן צאצא של האחר. ואם יש להם מרווחים זרים אז אין קשר של צאצא בינהם.

עקרון המסלול הלבן – נניח ב-G יש מסלול $v_1\to\cdots\to v_n$ ונניח שהקודקוד הראשון במסלול שה-DFS מגלה הוא v_1. אזי כל שאר הצמתים יהיו צאצאים של v_1 ב-DFS (הוכחה באינדוקציה).

שימושים: **(1) מציאת רכיב קשירות בגרף לא מכוון** - נריץ DFS, כל עץ בפלט שלו הוא רכיב קשירות. **(2) האם 2 צמתים u,v נמצאים באותו רכיב קשירות?** אם הם באותו עץ אחרי ה-DFS אז כן. נבדוק ע"י הליכה אחורה עם מצביעי π ובדיקה שיש אותו אב. **(3) האם גרף נתון מכיל מעגל?** קיים מעגל $\Leftrightarrow$ קיימת קשת אחורית. **(4) מיון טופולוגי של גרף מכוון א-ציקלי (DAG)** – מסדר את הצמתים בסדר $v_1,\dots,v_n$ כך שלכל קשת (v_i,v_j)מתקיים $i<j$. נריץ DFS וסדר הצמתים יהיה הפוך לערכי $f(\cdot)$ (v_1 עם ערך f מקסימלי). **(5) מציאת גשרים, קודקודים מנתקים ורכיבי קשירות** – גרף לא מכוון וקשיר ניתן לכוון לגרף קשיר בחוזקה $\Leftrightarrow$ אין בו גשרים. **(6) מציאת רכיבי קשירות חזקה (רק"ח)**

בגרף מכוון - (1) נריץ DFS על G ונסדר את הצמתים לפי סדר $f(\cdot)$ יורד ונקרא לרשימה L. (2) נבנה את הגרף ההפוך G^T. (3) נריץ DFS על G^T, כאשר הפרוצ' הראשית משתמשת ב-L בתור רשימת הצמתים. (4) כל עץ ב-DFS השני הוא רק"ח. (*) גרף העל הוא DAG.
(*) כדי לבדוק אם גרף הוא **קשיר חזק**, נריץ DFS מ-s על G ועל G^T. אם בשני המקרים קיבלנו עץ יחיד, G קשיר חזק.



סיווג קשתות - לקשתות המשמשות להגדרת π נקרא **קשתות עץ**. האחרות מתחלקות ל: **(1) קשת קדמית**- מחברת אב קדמון לצאצא. **(2) קשת אחורית**- מצאצא לאב קדמון. **(3) קשת חוצה**- מחברת 2 צמתים שאינם צאצא/אב קדמון.
(*) ה-DFS עובר על כל קשתות הגרף, בחלק מהן הוא משתמש להגדרת π, לבניית עץ ה-DFS, ובשאר לא.
(*) בגרף מכוון, מסתכלים על קשת (u,v) ברגע ש-u בודק אותה: (1) v לבן $\Leftarrow$ קשת עץ. (2) אפור $\Leftarrow$ אחורית. (3) שחור $\Leftarrow$ אם $d[u]<d[v]$ אז קדמית, אם $d[u]>d[v]$ אז חוצה.
(*) בגרף לא מכוון יש רק 2 סוגי קשתות – עץ ואחורית.
צבעים: לבן-עדיין לא ביקרנו בו. אפור-נתקלנו בצומת אך טרם סיימנו. שחור-סיימנו את הביקור בצומת.

4 MST - עץ פורש מינימאלי (עפ"מ)

עץ פורש: מכיל את כל הצמתים ו-$|V|-1$ קשתות מ-E.
קלט: גרף לא מכוון עם פונ' משקלים. **תכונות:** (1) פורש את הגרף. (2) סכום המשקולות של הקשתות בעץ הוא המינימאלי מבין הפורשים.

גישה כללית – תת-קב' A של E נקראת קב' מבטיחה אם קיים עפ"מ T המכיל את A. אם A קב' מבטיחה ו-$e\notin A$ אז נקרא ל-e קשת בטוחה עבור A אם $A\cup\{e\}$ עדיין קב' מבטיחה. **אלגו':** $A:=\emptyset$. כל עוד $A<V-1$, מצא קשת בטוחה e עבור A ובצע $A:=A\cup\{e\}$. בסוף נחזיר את A.

טענה: אם A קב' מבטיחה ו-(V_1,V_2) חתך שמכבד את A (אין ב-A קשת חוצה), ו-e קשת חוצה קלה של החתך (משקלה מינימלי מבין הקשתות החוצות) אז e קשת בטוחה עבור A.

Prim: יש רכיב קשירות אחד C, ומוסיפים אליו בכל פעם צומת היושב על הקשת הקלה ביותר המחברת אותו ל-C. משתמשים בתור עדיפות $Q=V\backslash C$, שלכל צומת מפתח המייצג את משקל הקשת הקלה ביותר המחברת אותו ל-C. **סיבוכיות:** בערימה רגילה - $O(E\cdot logV)$. במערך לא ממוין - $O(V^2+E)$. **בערימת פיבונאצ'י** - $\boldsymbol{O(V\cdot logV+E)}$.

Kruskal: בכל שלב נחפש קשת במשקל מינימלי שמחברת 2 רכיבי קשירות שונים של A, נחבר אותם ונוסיף את הקשת ל-A. הקשתות ממוינות ע"פ משקל בסדר עולה. **סיבוכיות:** $O((E+V)\cdot logV)$.
(*) הפלט של קרוסקל תלוי אך ורק בסדר הממוין של הקשתות ויכול לחשב כל עפ"מ. **מסקנה:** אם כל המשקלים שונים, אז יש עפ"מ יחיד.
(*) אם פונ' המשקל היא $w:E\to\{0,\dots,k\}$, אז ניתן לבצע מיון בזמן לינארי, ונקבל $O(E+(E+V)\cdot a(V))$ (a - פונ' אקרמן ההפוכה, $(O(sort\backslash extract.min+(\#union\&find+\#make.set)\cdot a(V))$.

טענה: תהי $f:\mathbb{R}\longrightarrow\mathbb{R}$ פונקציה עולה ממש, אם נגדיר משקל חדש $w^*(e)=f(w(e))$ אז: T עפ"מ לפי w $\Leftrightarrow$ T עפ"מ לפי w^*.
טענה: T_1,T_2 עפ"מים באותו גרף עם אותה פונ' w, אז אם נסדר את משקלי הקשתות בצורה ממוינת נקבל: $x_1=y_1,x_2=y_2,\dots$.
טענה: עבור 2 עפ"מים T_1 ו-T_2, תהי $E_{1,r}$ קב' הקשתות של T_1 שמשקלן לכל היותר r, והקב' $E_{2,r}$ מוגדרת באותה דרך, אז הגרפים $(V,E_{1,r})$ ו-$(V,E_{2,r})$ מחזיקים בדיוק באותם רכיבי קשירות
טענה: אם נוסיף קשת לעץ פורש היא תסגור מעגל יחיד.

5 Single Source Shortest Paths - מק"בים מ-s

מסלול קצר ביותר מצומת s לכל צומת אחרת בגרף (בהתחשב במשקל על הקשתות). מניחים שאין מעגלים שליליים.

תכונות מק"בים: **(1)** תת מסלול של מק"ב הוא מק"ב. **(2) א"ש המשולש**: אם $(u,v)\in E$ אז $\delta(s,v)\le\delta(s,u)+w(u,v)$. אם u צומת אחד לפני v על מק"ב מ-s ל-v אז יש שיוויון. **(3)** אם כל המשקולות הן 1, אז משקל מסלול=מס' הקשתות ונפתור עם BFS.

פעולת Relax – מבוצע על קשת (u,t). נבצע $d[t]=min\{d[t],d[u]+w(u,t)\}$. אם $d[t]$ השתנה אז $\pi(t)=u$.

Dijkstra: לא עובד עם משקלים שליליים. משתמשים בתור עדיפות $Q:=V$ שמנוהל ע"י המפתח $d[]$ ובקב' S שהיא קב' הצמתים שהוצאנו מהתור Q. אחרי שמוציאים מ-Q את הצומת המינימלי u מבצעים $Relax$ על כל שכניו v: (1) $v\notin Q$ $\Leftarrow$ לא צריך $Relax$, אין לו השפעה. (2) $v\in Q$ $\Leftarrow$ יגרור פעולת $decrease\_key(v)$. **סיבוכיות:** ערימה רגילה - $O((E+V)\cdot logV)$. **בערימת פיבונאצ'י** – $\boldsymbol{O(V\cdot logV+E)}$. ובאופן כללי – $O(V+V\cdot extract\_min\_Q\ +E\cdot decrease\_key)$.
(*) אם פונ' המשקל היא $w:E\to\{0,\dots,k\}$, אז ניתן להשתמש במערך במקום ערימה ולשמור מצביע לתא המינימלי שלא ריק במערך, ונקבל סיבוכיות של $O(E+k\cdot V)$ ($k\cdot(V-1)$ זה המסלול הכי ארוך שיכול להיות לנו ולכן זה גודל המערך. כל תא במערך מתאים לאורך מק"ב).

Bellman–Ford: עובד גם עם משקלים שליליים. מאתחל $d[s]=0$ ולכל שאר הצמתים $d[u]=\infty$. מבצע $Relax$ על כל הקשתות בגרף $V-1$ פעמים. אם לאחר מכן יש לקשת כלשהי $Relax$ משפר אז יש מעגל שלילי בגרף ונחזיר שגיאה. **סיבוכיות:** $O(V\cdot E)$.
טענה: לאחר ביצוע k $Pivot$-ים באלג' BF אז מצאנו את כל המק"בים מ-s שאורכם $\le k$.

טופולוגי: בגרף מכוון א-ציקלי ניתן לבצע מיון טופולוגי ולרוץ על הצמתים אחד אחד לפי הסדר, ומכל צומת נבצע $Relax$ על כל הקשתות שיוצאות ממנו. **סיבוכיות:** $O(E+V)$.

6 All-Pairs Shortest Paths – מק"ב בין כל u,v

חיקוי Bellman–Ford: אם $i=j$ אז $d_{ij}^{(0)}=0$, ואחרת $d_{ij}^{(0)}=\infty$.
$d_{ij}^{(m)}$ - המשקל המינימלי מבין כל המסלולים מ-i ל-j שמכיל לכל היותר m קשתות. נסמן $w_{ij}=w(i,j)$. עבור כל i,j נבצע:
$d_{ij}^{(m)}\ =\ min\ (d_{ij}^{(m-1)},\min_{1\le k\le n}\ \{d_{ik}^{(m-1)}\ +\ w_{kj}\})$. וזה שקול להכפלת מט': $d_{ij}^{(m)}=\sum_{i=1}^{n}d_{ik}^{(m-1)}\cdot w_{kj}$ כאשר: $\cdot\to+,\ \ +\to min$.
נבצע $V-1$ איטרציות כאלה. $d_{ij}^{(n-1)}$ זה משקל המק"ב מ-i ל-j.
סיבוכיות: $O(V^4)$, אבל את הכפלת המט' ניתן לבצע $logV$ פעמים במקום V [במקום $(..((\dots(w\cdot w)\cdot w)\cdot w)..)$ נכפיל $[(..(w\cdot w)(w\cdot w)..)$ ונקבל $O(V^3\cdot logV)$.
(*) π_{ij} מצביע לצומת הקודם ל-j במק"ב שהתגלה עד כה מ-i. נעדכן: אם $k\neq j$ נתן לנו את הערך המינימלי החדש ל-d_{ij} אז נבצע $\pi_{ij}=k$.
(*) אם קיים i כך ש-$d_{ii}^{(n)}<0$ אז יש מעגל שלילי בגרף.

Floyd-Warshall: נמספר את הצמתים. "גובה" המק"ב=האינדקס המקסימלי של צמתי המק"ב. נבנה מק"ב לפי גובה עולה – לכל i,j,k נחשב את אורך המק"ב מ-i ל-j בגובה לכל היותר k (נסמנו ב- $d_{ij}^{(k)}$), ונחשב כך: $d_{ij}^{(k)}=min\{d_{ij}^{(k-1)},d_{ik}^{(k-1)}=d_{kj}^{(k-1)}\}$. **סיבוכיות**: $O(V^3)$.

Johnson: נהפוך את המשקולות לאי-שליליים ונריץ את דייקסטרה v פעמים עם פונ' המשקלים w^*. **סיבוכיות:** $O(V^2\cdot logV+V\cdot E)$.
למה: תהא $h:V\to\mathbb{R}$ פונ' על הצמתים ונגדיר משקולות חדשים $w^*(u,v)=w(u,v)+h(u)-h(v)$ לכל קשת, אז מק"ב תחת w $\Leftrightarrow$ מק"ב תחת w^*.
בניית פונ' h כך ש-$w^*\ge0$: נוסיף לגרף צומת חדש s וקשתות מ-s לכל שאר הצמתים עם משקל 0. נריץ בלמן-פורד מ-s ונקח $h(v)=\delta(s,v)$.

7 Network Flow - זרימה ברשתות

תכונות זרימה חוקית: **(1)** אילוצי קיבול $f(u,v)\le c(u,v)$. **(2)** אנטי סימטריה $f(u,v)=-f(v,u)$. **(3)** שימור זרימה: לכל צומת $u\neq s,t$ מתקיים $\sum_v f(u,v)=0$. **(4)** $f(u,u)=0$.

ערך זרימה – סה"כ הזרימה הנכנסת ל-t = סה"כ הזרימה היוצאת מ-s: $|f|=\sum_u f(u,t)=\sum_u f(s,u)$. **(*)** נרצה למצוא: $|f|$ max.

רשת שיורית - G_f: עותק של G שבו $c_f(u,v)=c(u,v)-f(u,v)$.

זרימה בין קבוצות - הזרימה מכל צומת בקב' $A\subseteq V$ לכל צומת בקב' $B\subseteq V$. $f(A,B)=\sum_{a\in A,b\in B}f(a,b)$. **תכונות:** **(1)** $f(A,B)=-f(B,A)$. **(2)** $f(X,X)=0$. **(3)** אם X,Y קב' זרות אז $f(X\cup Y,Z)=f(X,Z)+f(Y,Z)$ וכנ"ל עבור $f(Z,X\cup Y)$.

Ford-Fulkerson: כל עוד קיים מסלול משפר, נגדיל את f ע"י הזרמה נוספת לאורך המסלול. מסלול משפר הוא מסלול מ-s ל- t ב-G_f (קשתות עם קיבול 0 לא יופיעו ב-G_f). $C_f(p)$ הוא הקיבול המינימאלי מבין הקשתות במסלול משפר p. נגדיר זרימה חדשה:

$$f^*(u,v)=\begin{cases}f(u,v) & (u,v)\notin p\\ f(u,v)+C_f(p) & (u,v)\in p\\ -f^*(v,u) & (v,u)\in p\end{cases}$$

ו- מתקיים: $|f^*|=|f|+C_f(p)$. **סיבוכיות:** תלוי בקיבולים. כאשר הזרימה היא בשלמים, יקח לכל היותר צעדים השווים לזרימה המקסימלית. מחפשים מסלול משפר קצר ביותר (מס' הקשתות) על הרשת השיורית ע"י BFS. הסיבוכיות תהיה מס' האיטרציות כפול עלות BFS. מס' האיטרציות הוא $O(E\cdot V)$ והגרף קשיר לכן עלות BFS היא $O(E)$, ובסה"כ $O(E^2\cdot V)$.
(*) אם מחפשים לדעת האם ערך זרימה גדול מ-k אז נקבל: $O(k\cdot E)$.

MinCut-MaxFlow: נגדיר חתך – חלוקה של V ל-2 קב' זרות (S,T) כך ש- $s\in S,t\in T$. הזרימה המקסימאלית שווה לקיבול החתך המינימאלי. ע"פ המשפט, התנאים הבאים שקולים: **(1)** f זרימה מקסימאלית. **(2)** הרשת השיורית G_f לא מכילה מסלול מ-s אל t. **(3)** $|f|=C(S,T)$ לאיזשהו חתך (S,T).
טענה: לכל זרימה f ולכל חתך (S,T) מתקיים $|f|=f(S,T)$.
טענה: לכל זרימה f ולכל חתך (S,T): $|f|=f(S,T)\le C(S,T)$.

Edmonds-Karp: נאתחל את הזרימה לזרימת אפס. נבנה רשת שיורית. נריץ BFS על הרשת ונחפש מסלול קצר בין s ל-t. נשפר את הזרימה לפי המקסימום שאפשר להזרים במסלול המשפר שמצאנו. נחזור על התהליך (בניית רשת שיורית מחדש וכו'). **סיבוכיות**: עלות של BFS כפול מס' האיטרציות. עלות BFS $O(E)$ כי הגרף קשיר ומספר האיטרציות חסום ע"י $O(E\cdot V)$ ולסיכום העלות היא $O(E^2\cdot V)$.

Dinic: הרעיון הוא לבנות רשת שכבתית מהרשת השיורית (ע"י BFS), לחפש בה כמה שיותר מסלולים משפרים ורק כאשר t לא נגיש יותר, לבנות רשת שיורית חדשה. ברשת השכבתית מתעלמים מכל הקשתות ההפוכות. שיטת איתור המסלולים ברשת השכבתית דומה ל-DFS. כאשר "מסיימים" עם צומת מוחקים אותו וכל הקשתות הנכנסות אליו. כאשר מגיעים ל-t מזרימים זרימה מקסימאלית במסלול, מוחקים קשתות רוויות וממשיכים להתקדם מהאב של הקשת הרוויה הקרובה ביותר לצומת s. **סיבוכיות:** $O(E\cdot V^2)$ (בכל פאזה לכל היותר $O(E\cdot V)$ צעדים כפול מס' פאזות המקסימליות, שזה נגזר ממספר השכבות המקסימלי – $O(V)$).

רשתות 0/1: קיבול כל קשת היא 1 (או 0). ברשתות כאלו כל פאזה של דיניץ' מתבצעת ב-$O(E)$ (כי במחיקה כל הקשתות במסלול רוויות). לכן, ריצת דיניץ כולה לוקחת $O(E\cdot V)$ (מדויק יותר: $O(E^{1.5})$). לעיתים ניתן למצוא חסם הדוק יותר ע"פ טיפוס.
רשת מטיפוס 1 – רשת בה אין קשתות מקבילות (או אנטי מקבילות). **סיבוכיות:** $O(E\cdot V^{2/3})$.
רשת מטיפוס 2 – רשת בה לכל צומת יש דרגת כניסה 1 או דרגת יציאה 1. **סיבוכיות:** $O(E\cdot V^{0.5})$.
(*) טיפוס הרשת (1 או 2) נשמר גם ברשת השיורית.
טענה: נניח זרימה נוכחית היא 0 וערך זרימה מקסימאלית M. אז מספר השכבות הוא $k\le\frac{E}{M}$ ברשת כללית ו- $k\le\frac{V-2}{M}+1$ בטיפוס 2.

זיווג מושלם: **משפט Hall** - יהיה G גרף דו"צ $V=X\cup Y$, $E\subseteq X\times Y$, $|X|=|Y|$. לכל $A\subseteq X$ נסמן ב-$\Gamma(A)$ את קב' השכנים של צמתים של A ב-Y. אז ל-G יש זיווג מושלם (כלומר זיווג בגודל $|X|=|Y|=0.5\cdot|V|$) $\Leftrightarrow$ לכל $A\subseteq X$ מתקיים $|\Gamma(A)|\ge|A|$.

משפט מנגר - $G=(V,E)$ גרף מכוון. $s,t\in V$. מס' הקשתות המינימלי שצריך להסיר כדי לנתק את s מ-t שווה למס' המקסימלי של מסלולים זרים בקשתות מ-s ל-t (והם שווים לזרימה המקסימלית).
(*) כל הקשתות בחתך המינימלי רוויות (קיבול=0).
(*) אם נסיר את הקשתות בחתך המינימלי ננתק את הגרף.

8 String Matching - התאמת מחרוזות

קלט: מחרוזת טקסט T ($|T|=n$), ומחרוזת תבנית P ($|P|=m$).
מטרה: למצוא את כל המקומות ב-T בהם מופיע P.
בשיטה נאיבית נוכל לבצע השוואה מכל מקום במחרוזת T. זה מבוצע בזמן $O(m\cdot n)$. ראינו שיטת השוואות באמצעות טבלת π שמבצעת את ההשוואה בזמן $O(m+n)$.

KMP: מחקה את האוטומט "ללא תלות" באות הקלט הבאה. (i - מיקום ב-T ועליו מתבצע לולאת ה-for, j - כמה התאמה יש. $\pi[j]$ – לאן להזיז את $P[1,\dots,j]$ אם הבדיקה נכשלה). קוראים את האות הבאה של T $\Leftarrow$ $T[i]$: (1) כל עוד אין התאמה $P[j+1]\neq T[i]$ ו-$j>0$ $\Leftarrow$ $j:=\pi[j]$. (2) אם יש התאמה $P[j+1]=T[i]$ $\Leftarrow$ $j:=j+1$. (3) אם $j=m$=אורך P $\Leftarrow$ נודיע שנמצאה התאמה ו-$j:=\pi[j]$. (ואז נקרא את האות הבאה ב-T $\Leftarrow$ $T[i+1]$). (3*) בפונ' של חישוב ה-π: במוקם T יהיה P ובשלב 3 יתבצע $\pi[i]:=j$. **סיבוכיות**: בניית טבלת π – $O(m)$, פרוצדורה ראשית - $O(n)$, סה"כ $O(m+n)$.
(*) $\pi(i)$ – גודל הרישא ממש המקסימלית של $P[1..i]$ שהוא גם סיפא ממש של $P[1..i]$, כלומר, ה-$i>k$ המקסימלי כך ש-$P[1..k]$ הוא סיפא (ובהכרח גם רישא) של $P[1..i]$.
(*) π מחושבת רק מתוך P. לדוגמא אם $P=ABABAC$ אז $\pi(4)=2$, ואם התאמנו 4 תווים נזוז 2 מקומות. (גודל התזוזה שלנו הוא $i-\pi(i)$).
(*) KMP יכול לייצר גם את $\sigma(i)$ שערכה הוא גודל הרישא המקסימלית של P שהיא סיפא של $T[1\dots i]$ ($\sigma[i]:=j$ בסיום ה-for).
(*) מס' הקריאות המקסימלי ל-π: $n-min(m-1,1)$ (n כאשר ב-P רק תו אחד-$m=1$, ו-T מורכב רק מתו זה, ו-$n-1$ עבור P גדול יותר).

9 Linear Programming - תכנות לינארי

תרגום בעיות לצורה סטנדרטית – **(1)** הפיכת min ל-max של המינוס. **(2)** הפיכת $\ge$ ל- $\le$ של המינוס. **(3)** הפיכת = ל- $\le$ וגם ל- $\ge$. **(4)** הפיכת x_j שלא חייב להיות חיובי ל- $x_j=x_j'-x_j''$ כאשר נוסיף את התנאים $x_j',x_j''\ge0$.

$standart\ form$	$slack\ form$
$\max\sum_{j=1}^{n}c_j\cdot x_j$ $\sum_{j=1}^{n}a_{ij}\cdot x_j\le b_i\quad 1\le i\le m$ $x_i\ge0\quad 1\le i\le n$	$\max\sum_{j=1}^{n}c_j\cdot x_j$ $x_{n+i}=b_i-\sum_{j=1}^{n}a_{ij}\cdot x_j$ $1\le i\le m$ $x_i\ge0\quad 1\le i\le n+m$

(*) לאחר ביצוע $pivot$ על צורת ה-$slack$, יכול להיות בפונ' המטרה מס' ללא מקדם, כלומר $maxc_0+\sum_{j=1}^{n}c_j\cdot x_j$.
(*) כל צורת $slack$ מכילה n משתנים לא בסיסיים ו-m משתנים בסייסים, ולכן סה"כ יש $\binom{m+n}{m}<2^{m+n}$ צורות $slack$ שקולות.

משתנה בסיסי בצורת $slack$ - המשתנה בצד שמאל של האילוץ (והוא גם לעולם לא יופיע בפונ' המטרה), כל שאר המשתנים הם לא-בסיסיים.
פיתרון פיזיבילי - הצבה כלשהי שמקיימת את כל האילוצים הדרושים.
פיתרון בסיסי - הצבת 0 בכל המשתנים הלא-בסיסיים(אלה שבצד ימין).

Simplex: נבדוק שהפיתרון הבסיסי הוא פיזיבילי, במידה ולא נמצא צורת $slack$ שבה הפיתרון הבסיסי כן פיזיבילי. לאחר מכן, בכל שלב נבצע $Pivot$: נבחר משתנה בסיסי עם מקדם חיובי מפונ' המטרה (זהו המשתנה הנכנס - למשוואות) שאותו אנחנו רוצים להגדיל, ונחליף אותו מול אחד המשתנים הלא-בסיסיים (זהו המשתנה היוצא). לאחר ה-$Pivot$ נקבל צורת $slack$ חדשה. נמשיך לבצע $Pivot$-ים עד שכל המקדמים של כל המשתנים בפונ' המטרה שליליים, ואז הפיתרון האופטימלי הוא הפיתרון הבסיסי. **סיבוכיות**: $Pivot=O(m\cdot n)$, מס' הצעדים $2^{m+n}>$, וסה"כ נקבל: $O(m\cdot n\cdot2^{m+n})$.

<u>בחירת המשתנה היוצא</u> - x_j הוא המשתנה הנכנס. נבחר את המשתנה היוצא ע"י המשוואה עם המשתנה הלא-בסיסי שהכי מגביל את הגדלת המשתנה הנכנס, כלומר המשתנה היוצא יהיה בעל המשוואה עם הערך החיובי הקטן ביותר של $b_i/-a_{ij}$.
(*) אם קיים פיתרון פיזיבילי והפיתרון האופטימלי חסום אז הפיתרון האופטימלי=לפיתרון האופטימלי של הבעיה הדואלית.
(*) אם X פיתרון לבעיה הרגילה ו-Y פיתרון לבעיה הדואלית שלה ומתקיים $\sum_{j=1}^{n} c_j x_j = \sum_{i=1}^{m} b_i y_i$ אז $X = Y$ וזהו פיתרון אופטימלי.
וזה בגלל שתמיד מתקיים: $\sum_{j=1}^{n} c_j x_j \le \sum_{i=1}^{m} b_i y_i$ (לפי **משפט הדואליות החלש**: דואלית-עם מינימום≤רגילה-עם מקסימום).

הבעיה הרגילה	הבעיה הדואלית
$\max \sum_{j=1}^{n} c_j \cdot x_j$	$\min \sum_{i=1}^{m} b_i \cdot y_i$
$\sum_{j=1}^{n} a_{ij} \cdot x_j \le b_i \quad 1 \le i \le m$	$\sum_{i=1}^{m} a_{ij} \cdot y_i \ge c_j \quad 1 \le j \le n$
$x_i \ge 0 \quad 1 \le i \le n$	$y_i \ge 0 \quad 1 \le i \le m$

הבעיה הרגילה	הבעיה הדואלית
$\max C_{1\times n} \cdot X_{n\times 1}$	$\min {B^T}_{1\times m} \cdot Y_{m\times 1}$
$A_{m\times n} \cdot X_{n\times 1} \le B_{m\times 1}$	${A^T}_{n\times m} \cdot Y_{m\times 1} \ge {C^T}_{n\times 1}$
$X_{n\times 1} \ge 0$	$Y_{m\times 1} \ge 0$

<u>מציאת פיתרון בסיסי פיזיבילי</u> (בצורת $slack$) – פונ' המטרה תהיה $max - x_0$, נוסיף את התנאי $x_0 \ge 0$ ולכל משוואה נוסיף בצידה הימני $+x_0$. המשתנה היוצא ה-1 יהיה זה שבעל הערך הכי קטן בפיתרון הבסיסי (הערך חייב להיות שלילי, כי אין פיתרון בסיסי פיזיבילי). לאחר מכן נבצע $Pivot$-ים עד שנמצא פיתרון אופטימלי, ובסוף נחזור לתוכנית הלינארית בצורת ה-$slack$ שהתחלנו איתה (זאת שללא x_0) ונציב בפונ' המטרה שלה את כל המשתנים הבסיסיים (כלומר, נציב את החלק הימני של המשוואות שלהם) שקיבלנו בפיתרון האופטימלי של התוכנית הלינארית הסופית (זאת עם ה-x_0) וכן נציב $x_0 = 0$. וכעת קיבלנו צורת $slack$ עם פיתרון בסיסי פיזיבילי.

10 תכנות דינאמי

(*) כפל מטריצות יעיל - נתונה סדרה של מט' $A_1, \dots, A_n$, כאשר A_i בגודל $p_{i-1} \times p_i$. רוצים למצוא סדר הכפלה עם מס' מינימלי של פעולות. למשל כפל מט' $q \times r$ במט' $p \times q$ הוא pqr. נסמן $M[i,j]$ = המחיר המינימלי של כפל המט' $A_i \cdot \dots \cdot A_j$. נגדיר $M[i,i] = 0$ ו-$M[i,j] = min_{i \le k < j}\{M[i,k] + M[k+1,j] + p_{i-1}p_k p_j\}$. נחשב את $M[i,j]$ לפי סדר עולה של $(j-i)$ כאשר $j > i$. ובסוף $M[1,n]$ יהיה המחיר האופטימלי. סיבוכיות: $O(n^3)$.

(*) חישוב תת-סידרה משותפת מקסימלית – מקבלים 2 מחרוזות X,Y $(|X| = m, |Y| = n)$ ורוצים למצוא סדרה המופיעה באותו סדר אך לא בהכרח באותו רצף ב-2 המחרוזות באורך מקסימלי. נסמן ב-$X^{(i)}$ את הרישא של i האיברים הראשונים של X, וב-X_i את האיבר ה-i של X.
$LCS[X,Y]$:
$$if\ X_m = Y_n\ then\ \ LCS\left(X^{(m-1)}, Y^{(n-1)}\right) \parallel (X_m)$$
$$else\ \ return \max\{LCS(X^{(m-1)}, Y), LCS(X, Y^{(n-1)})\}$$
בטבלה נחשב את הביטוי לפי סדר שורות עולה, וכל שורה לפי סדר עולה של טורים. נחזיר את $LCS[X,Y]$. סיבוכיות: $O(m \cdot n)$.

(*) מס' המסלולים מ-s ל-t – נסמן ב-$M[u]$ את מס' המסלולים מ-u ל-t. $M[t] = 1$. נריץ BFS על G^R. אם $d[u] = \infty$ אז $M[u] = 0$ ועבור כל הצמתים האחרים נחשב: $M[u] = \sum_{w \in V, (u,w) \in E} M[w]$ ונחזיר בסוף את $M[s]$. סיבוכיות: $O(E+V)$.

(*) רמת כיפיות מקס' במסיבה – עובד אינו כיפי שהבוס שלו נמצא. נגדיר: $S^+(v)$ - הכיפיות המקס' כש-v מוזמן. $S^-(v)$ - הכיפיות המקס' כש-v לא מוזמן. אם v עלה: $S^+(v) = fun(v)$ ו-$S^-(v) = 0$. אם v לא עלה: $T(v)$ – קב' הבנים של v. $S^+(v) = fun(v) + \sum_{u \in T(v)} S^-(u)$ ו-$S^-(v) = \sum_{u \in T(v)} max\,\{S^-(u), S^+(u)\}$. נחזיר $max\,\{S^-(r), S^+(r)\}$ כאשר r זה השורש. סיבוכיות: $O(n)$.

(*) בעיית הסוכן הנוסע – צריך לבקר ב-n ערים ולעבור מרחק מינימלי. נסמן ב-$p(i,j)$ את המרחק מ-i ל-j. נגדיר $M(A,k)$ להיות המרחק המינימלי מ-1 אל k שעובר בצמתים שב-$A\backslash\{k\}$. נחשב כך: $M(\emptyset, k) = p(1,k)$ ו-$M(A,k) = min_{i \in A}\{M(A\backslash\{k\}, i) + p(i,k)\}$. נחזיר את $min_k\{M(\{2,\dots,n\},k)\}$. סיבוכיות: $O(2^n \cdot n^2)$.

(*) סכום מקס' של תת-סידרה מונוטונית לא יורדת בת$\ge k$ איברים מ-$a_1, \dots, a_n$ - נגדיר לכל $1 \le i \le n$ ולכל $1 \le j \le k$ את $S(i,j)$ להיות הסכום המקס' לתת-סידרה לא יורדת בת j איברים שמסתיימת ב-a_i. נאתחל $S(i,1) = a_i$. ונחשב לפי סדר עולה של j את הביטוי: $S(i,j) = a_i + \max_{m<i, a_m \le a_i}\{S(m, j-1)\}$, ונחזיר את $\max_{1 \le i \le n, 1 \le j \le k}\{S(i,j)\}$. סיבוכיות: $O(k \cdot n^2)$.

(*) <u>שאלה:</u> עבור **סדרת מספרים** ממשיים x1,x2,…,xn לחשב עבור כל i את מס' תתי-הסדרות העולות ממש ומסתיימות ב- xj (כולל). **פתרון**: ניצור מערך בגודל n מאותחל ל- 1, כי כל סדרה יכולה לכלול רק את האיבר האחרון. אח"כ נסכום עבור כל xi (כש- i<j) שקטן מה- xj את תתי הסדרות שלו, כי פשוט ניתן להוסיף בסוף את xj. **סיבוכיות:** n^2.

```
List a[1…n] ← [1…1]
 For j= 1 to n
  For i= 1 to j-1
 If  xi<xj  then
 a[i]= a[j]+a[i];
```

(*) <u>שאלה:</u> נתונות **2 סדרות** S ו-T. תארו אלגוריתם יעיל ככל האפשר שקובע אם S מופיע **כתת סדרה** בT כך שכל שני איברים עוקבים של S מופיעים במרחק של לכל היותר 50 זה מזה. **פתרון:** נכין טבלה M כך שM(i,j) יהיה 1 אם j האיברים הראשונים של S עונים על התנאים עבור i האיברים הראשונים של T (ו-j נופל על האיבר ה-i של T). כדי לחשב M(i,j) נבדוק האם T[i]=S[j] וגם [האם מתקיים M(i-1,j-1) או M(i-2,j-2) או … M(i-50,j-1)]. **סבוכיות:** מילוי הטבלה יתקיים ב-O(mn).

(*) שאלה: נתונה קבוצה של **n מספרים שלמים $1 \le a_1 < a_2 \dots < a_n < 2n$** ומספר שלם נוסף m. תארו אלגוריתם יעיל המחשב את מס' הפתרונות **למשוואה $a_1x_1 + a_2x_2 + \dots + a_nx_n = m$ כאשר $x_i \in \{0,1,2\}$**. **פתרון:** נגדיר מטריצה עם עמודות $0 \dots m (\le 4n^2)$ ושורות עם a_i. בכל תא בטבלה נמלא את מספר הפתרונות הקיימים כאשר עבור m=לערך העמודה וכאשר הסדרה מכילה את $a_1..a_i$ בלבד. העמודה הראשונה תכיל 1 (כי יש פיתרון יחיד כשm=0). בשורה הראשונה נסמן 1 בתאים שווים ל-$0*a_1$, $1*a_1$ ו-$2*a_1$ ובשאר 0. נמלא את המטריצה לפי שורות ואז עמודות (ע"פ הקוד). **סבוכיות:** O(mn)

```
If 2ai≤j then
a[i,j]=a[i-1,j-2ai]+
a[i-1,j-ai]+ a[i-1,j]
else if ai≤j then
a[i,j]=a[i-1,j-ai]+
a[i-1,j]
else
a[i,j]=a[i-1,j]
```

(*) שאלה: A מערך של n מספרים ממשיים. צ"ל אלגו' למציאת **תת קבוצה** של A **שסכומה מקסימאלי** אבל **אסור לבחור 2 איברים סמוכים מA. פתרון:** מטריצה M תכיל את הסכום המקסימאלי עם תתי סדרות המכילות רק איברים הקטנים שווים לו. מטריצה P תכיל זוג סדור עם קבוצה להוסיף לתת ופוינטר לאינדקס הבא של P. מאתחלים 2 תאים ראשונים. עוברים על המערך לפי התנאים שבקוד. לסיום, עוברים על המערך שוב מהסוף להתחלה ע"פ הפוינטר בזוג הסדור כשבכל שלב מוסיפים את האיבר שבחלק הראשון של הזוג הסדור.

```
if(A [i-2]+A[i]>A[i-1])
  M[i]←A[i-2]+A[i];
P[i]←<{i},i-2>
  else M[i]←A[i-1]; P[i]←<∅,i-1>
```

(*) שאלה: אותו דבר אבל **אורך תת הסדרה הוא לכל היותר k**. **פתרון:** מריצים אלגוריתם דומה, אבל לכל סכום סדרה שומרים גם כמה איברים יש עד כה ועותק ממש של כל איברי תת הסדרה. אם הגענו לתת סדרה הגדולה מk, נסיר מהעותק שלנו את האיבר הראשון (הוא הכי קטן) ונשים במקומו את האיבר הנוכחי (צריך גם לעדכן את הסכום להיות הסכום הקודם פחות האיבר הראשון + האיבר הנוכחי). **סבוכיות:** kn^2

(*) שאלה: נתונה **סדרת מספרים ממשיים**, רוצים **תת סדרה רציפה** כך ש**הסכום מקס'**. **פתרון:** מגדירים +Si ו-Si- פתרון אופטימלי עם/בלי x_i. חישוב הערכים: $S_i^+ = \max\{S_{i-1}^+ + x_i, x_i\}$, $S_i^- = \max\{S_{i-1}^+, S_{i-1}^-\}$. בסוף $S = \max\{S_n^+, S_n^-\}$. סוף הסדרה= התא הראשון מהסוף כך ש-$S_i^+ = S$.

11 תרגילים ופתרונות

(*) מציאת מעגל אויילר בגרף <u>לא מכוון</u> - נתחיל מסלול מצומת כלשהו עד שנחזור אליו ונסיר את כל הקשתות שעברנו עליהם. אם במסלול יש את כל הקשתות שב-E אז סיימנו, אחרת נתחיל מסלול מאחד הצמתים שעברנו עליהם שדרגתו חיובית עד שנחזור אליו בחזרה, ואז נאחד בין 2 המסלולים. נחזור על התהליך.
(*) מציאת מסלול אויילר בגרף <u>לא מכוון</u> – נוסיף קשת בין 2 הצמתים שדרגתם אי-זוגית ואז נריץ את האלג' חמציאת מעגל של אויילר.

<u>*BFS*</u> (מק"ב - לפי מס' הקשתות)

(*) צביעת צמתים בגרף דו"צ – נריץ BFS מצומת כלשהו וכל הצמתים במרחק אי-זוגי יהיו אדומים וכל הצמתים במרחק זוגי יהיו כחולים.
(*) האם הגרף הוא דו"צ – נריץ BFS ואם אין 2 צמתים באותה שכבה שיש בינהם קשת אז גם אין מעגל באורך אי-זוגי ולכן הגרף הוא דו"צ.
(*) גרף המק"בים מ-s בגרף <u>לא מכוון</u> - זהו גרף $G' = (V, E')$ כאשר $E' = \{e \in E \mid e\ is\ part\ of\ a\ shortest\ path\ from\ s\}$. אלג' ה-$BFS$ זורק 2 סוגי קשתות בגרף <u>לא מכוון</u>: (1) בין צמתים מאותה רמה. (2) בין צמתים מרמות עוקבות. נשנה את האלג' ה-BFS כך שנזרוק רק קשתות בין צמתים מאותה רמה ונקבל ש-π מייצג את גרף המק"בים מ-s.
(*) גרף המק"בים מ-s ל-t בגרף <u>מכוון</u> – זהו גרף $G' = (V, E')$ כאשר $E' = \{e \in E \mid e\ is\ part\ of\ a\ shortest\ path\ from\ s\ to\ t\}$. אלג' ה-$BFS$ זורק 3 סוגי קשתות בגרף <u>מכוון</u>: (1) בין צמתים מאותה רמה. (2) בין צומת מרמה i לצומת ברמה $i-1$. (3) בין צומת מרמה i לצומת ברמה $j < i$. נשנה את אלג' ה-BFS כך שלא נזרוק קשתות מסוג (2), ואז נריץ את ה-BFS החדש מ-s, נהפוך את הקשתות של הגרף שקיבלנו ונריץ את ה-BFS החדש מ-t, נהפוך שוב את הקשתות וסיימנו.
(*) מס' המק"בים מ-s לכל צומת – נשמור לכל צומת u את $w[u]$ שייצג כמה מק"בים מצאנו עד כה מ-s ל-u. נעדכן את ה-BFS: נניח שהגענו ל-u מהצומת v. אם $d[u] = \infty$ או ש-$d[u] = d[v] + 1$ אז נבצע $w[u] = w[u] + w[v]$.
(*) האם הקשת $\{u,v\}$ נמצאת על מעגל בגרף <u>לא מכוון</u> - נסיר את הקשת מהגרף ונבדוק אם יש מסלול מ-v ל-u ע"י BFS.

<u>*DFS*</u>

(*) מסלול שעובר בכל צמתי הקב' S בגרף <u>מכוון</u> – בעזרת DFS נבנה גרף על ונבצע מיון טופולוגי עליו. נחפש בגרף זה מסלול שעובר בכל הרק"חים שיש בהם לפחות צומת אחד מהקב' S. נעשה זאת ע"י BFS בין כל 2 רק"חים כאלה לפי הסדר הטופולוגי.

<u>עפ"מים</u>

(*) מציאת עפ"מ עם מס' מקסימלי של קשתות מהקב' S (מבין העפ"מים) – למשקל הקשתות מהקב' S נוסיף ε ואז נמצא עפ"מ כרגיל.
(*) מציאת עפ"מ שכל צמתי הקב' L הם עלים בעפ"מ – ניצור גרף ללא הצמתים שב-L וללא הקשתות שכוללות צומת מ-L. אם הגרף לא קשיר אז לא קיים עפ"מ כזה. אחרת, נמצא עפ"מ T על גרף זה, ועבור כל צומת מ-L נמצא את הקשת הקלה ביותר שמחברת בינה לבין T.
(*) האם קיים עפ"מ המכיל את הקשת $\{u,v\}$ – נסמן $V_1 = \{u\}$ ו-$V_2 = V\backslash\{u\}$. אם לכל קשת חוצה $\{u, v_2\} \in E$, $v_2 \in V_2$, מתקיים $w(u,v) \le w(u,v_2)$ אז קיים עפ"מ שמכיל את הקשת $\{u,v\}$.
(*) האם כל עפ"מ המכיל את הקשת $\{u,v\}$ – נסמן $V_1 = \{u\}$ ו-$V_2 = V\backslash\{u\}$. אם לכל קשת חוצה $\{u, v_2\} \in E$, $v_2 \in V_2$, מתקיים $w(u,v) < w(u,v_2)$ אז כל עפ"מ מכיל את הקשת $\{u,v\}$.
(*) האם יש לגרף 2 עפ"מים בדיוק – נמצא עפ"מ T_1, נוסיף למשקל קשתותיו בגרף ε. נמצא עפ"מ T_2. אם יש 0 או 2 קשתות ומעלה ב-T_2 שלא ב-T_1 אז אין. אחרת, נוסיף למשקל קשתות T_2 בגרף ε, נמצא עפ"מ T_3. אם T_3 שווה לאחד העפ"מים הקודמים אז יש בדיוק 2, ואחרת אין.

<u>*Single Source Shortest Paths*</u>

(*) מק"ב שמסתיים בצומת בגרף <u>מכוון</u> – אין מעגלים שליליים בגרף. עבור כל צומת $v \in V$ נרצה לחשב: $\delta^*(v) = \min_{u \in V} \delta(u,v)$. נוסיף צומת s לגרף עם קשתות במשקל 0 לכל הצמתים, ונריץ את אלג' BF מהצומת s. לאחר מכן יתקיים $\delta^*(v) = \delta(s,v)$, וזה כי מתקיים: $\delta(s,v) = \min_u\{w(s,u) + \delta(u,v)\} = \min_u\{\delta(u,v)\} = \delta^*(v)$.
(*) גרף כל המק"בים מ-s ל-t עם פונ' משקלים בגרף <u>מכוון</u> – נריץ את אלג' BF על G מהצומת s, ועל הגרף G^R (הקשתות הפוכות) מהצומת t. עבור כל $u \in V$ שמקיים $d(u) + d^R(u) = d(t)$ נכניס לגרף H את כל הקשתות במק"ב מ-s ל-u ומ-u ל-t. H הגרף הרצוי.
(*) מציאת מק"בים מ-s עם מס' קשתות זוגי בגרף <u>מכוון</u> – נשתמש ב-2 העתקים של הגרף, וכל הקשתות יהיו מעותק מס' 1 לעותק מס' 2 ולהפך. המרחק של המק"ב הרצוי מ-s ל-v יהיה $\delta(s_1, v_1)$.
(*) מציאת מק"בים מ-s עם x קשתות בגרף <u>מכוון</u> – נשתמש ב-$x+1$ העתקים של הגרף, וכל הקשתות יהיו מעותק מס' i לעותק מס' $i+1$. המרחק של המק"ב הרצוי מ-s ל-v יהיה $\delta(s_1, v_{x+1})$.
(*) מציאת המעגל הקל ביותר שמכיל את s וכל צומת $v \in V$ בגרף <u>מכוון</u> – נריץ אלג' למציאת מק"בים מהצומת s על G ועל G^R.
(*) מציאת מעגל שלילי בגרף <u>מכוון</u> – נמצא את כל הרק"חים ונריץ על כל אחד מהם בנפרד את אלג' BF. סיבוכיות: $O(E \cdot V)$.
(*) מציאת מסלול מ-s לכל צומת $v \in V$ עם מס' קשתות אדומות פחות כחולות מקסימלי בגרף <u>מכוון</u> **–** לקשת אדומה ניתן משקל -1 ולקשת כחולה ניתן משקל 1. נריץ BF מ-s.

<u>*All – Pairs Shortest Paths*</u>

(*) עבור כל 2 צמתים, האם קיים מסלול באורך m ביניהם בגרף <u>מכוון</u> – נגדיר מט' שכנויות: אם $(i,j) \in E$ אז $a_{ij} = 1$ ואחרת $a_{ij} = 0$. נחשב את A^m לפי כפל בוליאני- $A \times B = C$ מוגדר: $c_{ij} = \bigvee_{k=1}^{n}(a_{ik} \wedge b_{kj})$.
(*) עבור כל 2 צמתים, כמה מסלולים באורך m קיימים ביניהם בגרף <u>מכוון</u> **–** כמו מקודם, נחשב את A^m אך לפי כפל מט' רגיל- $A \times B = C$ מוגדר: $c_{ij} = \sum_{k=1}^{n}(a_{ik} \cdot b_{kj})$.

<u>*Network Flow*</u>

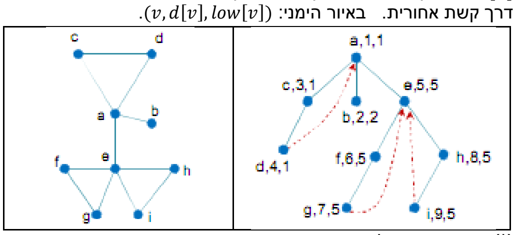
(*) כמה מסלולים זרים בקשתות מ-s ל-t יש בגרף <u>מכוון</u> – ניתן קיבול 1 לכל הקשתות בגרף ונריץ אלג' דיניץ. הזרימה המקס' זה הערך הרצוי.
(*) כמה מסלולים זרים בצמתים מ-s ל-t יש בגרף <u>מכוון</u> – נפצל כל $v \in V$ ל-v_{in}, v_{out} ונריץ את האלג' הקודם.
(*) מציאת החתך המינימלי (S,T) – נבנה רשת שיורית ביחס לזרימה המקס'. נריץ BFS מהמקור s ונכניס את כל הצמתים שבעץ ה-BFS ל-S ואת כל שאר הצמתים ל-T. סיבוכיות: $O(E+V)$.

<u>*String Matching (KMP)*</u>

(*) הרישא המקס' של T שהיא פולינדרום $(x = x^R)$ – לכל $T = xy$ מתקיים $T^R = y^R x^R$, ואם x פולינדרום אז רישא של T היא סיפא של T^R. נריץ KMP עם הטקסט T^R והתבנית T. נחזיר את $\sigma[n]$ $(|T| = n)$.
(*) מציאת חלוקה $T = xyx$ כך ש-$|y| \ge 10$ ו-$|x|$ מקס' – בהכרח $|x| \le \lfloor 0.5 \cdot (n-10) \rfloor = k$. נריץ KMP עם k התווים האחרונים של T כטקסט ו-k התווים הראשונים של T כתבנית. נחזיר את $\sigma[k]$.
(*) מציאת חלוקה $T = xyx$ כך ש-y תת-מחרוזת של x – אם יש חלוקה כזאת אז $|x| \le \lfloor 0.5 \cdot n \rfloor = k$. נריץ KMP עם k התווים האחרונים של T כטקסט ו-k התווים הראשונים של T כתבנית. מתקיים $x = [1 \dots \sigma[k]]$ ו-y יהיה כל שאר התווים שנשארו באמצע.
(*) האם T' סיבוב מעגלי (דחיפת האותיות ימינה) של T - נריץ KMP עם TT כטקסט ו-T' כתבנית. אם T' יופיע ב-TT אז נחזיר $True$.

(*) בעיית שער חליפין – c_i מטבע ו-a_{ij} מציין כמה מטבעות מסוג c_j נקבל תמורת מטבע אחד מסוג c_i. נרצה לבדוק האם קיימת סידרת המרות שתביא לרווח, כלומר האם קיימים $i_1, \dots, i_k$ כך ש: $a_{i1,i2} \cdot \dots \cdot a_{ik,i1} > 1$. <u>פתרון</u>: נוציא log משני האגפים ונכפיל ב-(-1) ונקבל: $-log(a_{i1,i2}) - \dots - log(a_{ik,i1}) < 0$. נבנה גרף מכוון שלם עם המטבעות כצמתים ולכל קשת (c_i, c_j) ניתן משקל $-log(a_{ij})$. כעת נחפש מעגל שלילי בגרף החדש.
(*) האם קיים מעגל בגודל 0 בגרף <u>מכוון</u>. נוסיף צומת s וקשתות ממנו לכל שאר הצמתים במשקל 0. נריץ BF וניצור את הגרף G' כאשר: $E' = \{(u,v) \in E \mid \delta(s,v) = \delta(s,u) + w(u,v)\}$. יש מעגל ב-$G'$ אם קיימת קשת אחורית (בדיקה עם DFS).
<u>טענה</u>: קיים **מעגל שנגיש מ-s** ב-G' ⟺ משקל מעגל ב-G הוא 0.

(*) קשת מנתקת (גשר), קודקוד מנתק, גרף דו קשיר, גרף דו"צ - נחשב את low תוך כדי הרצת DFS על הגרף. החישוב יתבצע לכל קודקוד ברגע שמסיימים לטפל בו (לאחר חישוב ערך f שלו). זאת בעזרת:
$$low[v] = min\begin{cases} d[v] & \text{הערות:} \\ d[u] & (v,u)\ \text{קשת אחורית בעץ} \\ low[u] & u\ \text{בן של}\ v\ \text{בעץ} \end{cases}$$
$low[u]$ מציין כמה גבוה ניתן להגיע בעץ ה-DFS מ-u או מצאצא שלו דרך קשת אחורית. באיור הימני: $(v, d[v], low[v])$.



(*) <u>הגדרה:</u> בגרף <u>לא מכוון</u>, **קודקוד מנתק** הוא קודקוד שהסרתו מנתקת את רכיבי הקשירות שלו.
<u>טענה:</u> בגרף <u>לא מכוון</u>, צומת v (<u>שאינו שורש</u>) **מנתק** ⟺ קיים w בן של v כך ש: $low(w) \ge d[v]$.
<u>טענה:</u> בגרף <u>לא מכוון</u>, אם-v <u>שורש</u> אזי **v מנתק** ⟺ יש לו לפחות 2 בנים באותה רמה (כלומר, מקיימים $d[w] < f[w] < d[z] < f[z]$).
(*) <u>הגדרה:</u> **גשר** בגרף <u>לא מכוון</u> היא קשת שהסרתה מנתקת את רכיבי הקשירות שלה. **<u>טענה:</u>** קשת **אינה גשר** ⟺ היא נמצאת על מעגל פשוט.
<u>טענה:</u> (u,v) **גשר** ⟺ v בן של u ב-DFS ומתקיים $low(v) = d[v]$.
<u>טענה:</u> גרף <u>לא מכוון</u> הוא **דו"צ** ⟺ אין **מעגלים באורך אי-זוגי**.
(*) במעגל מס' הצמתים=מס' הקשתות.

(*) (2) גרף <u>לא מכוון וקשיר</u> הוא **דו-קשיר** בקשתות (בצמתים) אם אין לו קשתות (צמתים) מנתקות. (3) **רכיב דו-קשירות** הוא **תת-גרף דו-קשיר מקסימלי**.
<u>טענה</u>: גרף <u>לא מכוון וקשיר</u> ניתן **לכיוון לקשיר בחוזקה** ⟺ אין בו גשרים.
<u>טענה</u>: לכל 2 **רכיבי דו-קשירות** יש לכל היותר קודקוד משותף אחד.
<u>מסקנה</u>: רכיבי הדו-קשירות מהווים חלוקה של קשתות הגרף.
(*) מוצאים **רכיבי דו-קשירות** בדומה לקודקודים מנתקים. אלא שלכל u שאינו שורש בודקים את בניו. לכל בן v אשר $low(v) \ge d(u)$ נפצל את u – עותק אחד עבור הקשת (u,v) ועותק אחד עבור השאר. עבור השורש נתאים עותק לכל בן.

12 תרגילים נוספים

(*) שאלה: n בנים וn בנות. כל אחד מכיר k מהמין השני. להוכיח שיש זיווג מושלם. **פתרון:** נבנה גרף דו"ץ בנים בנות. לכל קבוצה A⊆X מתקיים |Γ(A)|>|A|. נניח בשלילה כי |Γ(A)|<|A|. כל בן מכיר בדיוק k בנות ולהפך, ולכן, מכל קודקוד ב-A יוצאים k קודקודים אל Γ(A). יש k|A| קשתות מ-A אל Γ(A). דרגת הקודקודים ב-Γ(A) היא k. לכן, יש k|Γ(A)| קודקודים ממנה אל A. הקשר בין הקודקודים הוא הדדי, ולכן, הקשתות המחוברות אל קודקודים שב-A מוכלות בקבוצת הקשתות המחוברות לΓ(A). נקבל כי k|Γ(A)|<k|A|. כלומר, |Γ(A)| <|A| בסתירה. מכאן Hall.