שאלה 1

1) אתחל מטריצה G* בגודל $n \times n$ מלאה באפסים.
2) לכל $v \in V$ בצע:
3) BFS(G,v)
4) לכל $u \in V$ בצע:
5) אם $d[u] < \infty$ אז $G^*[v,u] \leftarrow 1$.

<u>סבוכיות</u>:

שורה 1 - $O(|V|^2)$

שורה 3 - $O(|V|(|V|+|E|))$

שורה 5 - $O(|V|^2)$

סה"כ: $O(|V|(|V|+|E|))$.

<u>נכונות</u>: G*[v,u]=1 ⇔ הושם הערך בשורה 5 ⇔ $d[u] < \infty$ בביצוע DFS(G,v) ⇔ קיים מסלול מ-u ל-v.

- **שאלה 2**

<u>האלגוריתם:</u>

1. אם מספר הצמתים בגרף אי-זוגי, החזר שאין חלוקה המתאימה לנתוני השאלה.
2. אחרת, הפרד את הגרף לרכיבי הקשירות שלו, באמצעות אלגוריתם הפירוק לרכיבי קשירות.
3. מצא את מספר רכיבי הקשירות של הגרף, ואת מספר הצמתים בכל רכיב. הכנס את הרכיבים שנמספרם $\{1,2,...,p\}$ לקבוצה C, והצמד לכל רכיב את מספר הצמתים שבו t_i.
4. השתמש באלגוריתם לפתרון סכומי תת-הקבוצות, כאשר קבוצת הפריטים היא C, קבוצת המשקלים היא $\{v_1,v_2,...,v_p\}$, והחסם הוא $|V|/2$.
5. בדוק את המשקל המקסימלי שפלט האלגוריתם. אם משקל זה שווה ל-$|V|/2$, הפרד את $|V|$ לקבוצה $A=S$ (קבוצת המשקלים שבחר האלגוריתם), ולקבוצה $B=V-S$. לקבוצה A הכנס את כל רכיבי הקשירות שהאלגוריתם החזיר כתת-קבוצה, ול-B את שאר הרכיבים (על כל צמתיהם). החזר חלוקה זאת.
6. אחרת, החזר שאין חלוקה המתאימה לנתוני השאלה.

<u>הוכחת נכונות:</u>

כמובן שאם מספר הצמתים בגרף אי-זוגי, אין שני תת קבוצות של V בגודל $|V|/2$, ולכן במקרה זה אין חלוקה.

טענה: אם שני צמתים בגרף נמצאים באותו רכיב קשירות, אז הם חייבים להמצא באותה קבוצה בחלוקת V.

הוכחה: יהיו u,v צמתים בגרף הנמצאים ברכיב קשירות של G, ונניח ש-$u \in A$ ו-$v \in B$. בגלל שהצמתים נמצאים באותו רכיב קשירות, קיים מסלול בין u ל-v. מכיוון שהצמתים נמצאים כל אחד בקבוצה שונה, חייבת להיות במסלול קשת בין צומת מ-A לצומת מ-B. קיבלנו סתירה לתנאי (ii) של החלוקה.

לכן כל רכיב קשירות חייב להמצא באופן מלא באחת הקבוצות. כדי שהקבוצות יהיו שוות, יש לחלק את קבוצת הרכיבים לשתי קבוצות, כך שהסכום של מספר הצמתים בכל רכיב יהיה שווה בשני

הקבוצות, כלומר $\sum_{i\in A} v_i = \sum_{j\in B} v_j$. יש כזאת חלוקה אם ורק אם יש קבוצה של רכיבים, שהסכום של מספר הצמתים בהם שווה בדיוק ל-$|V|/2$. האלגוריתם מוצא אם קיימת כזאת קבוצה באופן נכון לפי הנכונות של האלגוריתם הפותר את בעיית סכומי תת הקבוצות. אם אין תת-קבוצה שסכומה שווה בדיוק ל-$|V|/2$, אז אין חלוקה שווה של רכיבי הקשירות בין הקבוצות A ו-B (בכל אחת צריכים להיות בדיוק $|V|/2$ צמתים . אם אין תת קבוצה שסכומה $|V|/2$ אז אין כלל חלוקה מתאימה). לכן האלגוריתם קובע נכונה האם יש חלוקה מתאימה. בשלב (5) מחלק האלגוריתם את V ל-A ו-B, באופן שיש בכל אחת מהן בדיוק $|V|/2$ איברים. נוכיח שהחלוקה מקיימת את תנאי (ii).

נניח בשלילה שקיימת קשת $(u,v)\in E$ כך ש-$u\in A$ ו-$v\in B$. מכאן, קיים מסלול בין u ו-v ולכן הם נמצאים באותו רכיב קשירות. אבל האלגוריתם מכניס ל-A או ל-B את רכיבי הקשירות בשלמותם. סתירה.

<u>ניתוח סיבוכיות:</u>

שלב (1) מתבצע ב-$O(1)$.

את שלב (2) ניתן לבצע בזמן $O(|E|+|V|)$. לאחר מכן בשלב (3) צריך לעבור על כל קבוצה של רכיבים ולמנות את מספר הצמתים בה. בסך הכל עוברים על כל צומת פעם אחת ולכן זה נעשה בזמן $O(|V|)$.

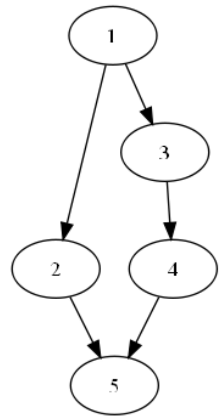
(4) רץ בזמן $O(p|V|)$, כאשר p הוא מספר רכיבי הקשירות. במקרה הגרוע, יש $|V|$ רכיבי קשירות (גרף ללא קשתות), ולכן שלב זה ירוץ בלכל היותר $O(|V|^2)$.

שלב (5) דורש את הבנייה של הקבוצות A ו-B, שתקח לכל היותר $O(|V|)$, ופעולות אחרות המתבצעות בזמן קבוע. בסך הכל, האלגוריתם רץ בסיבוכיות:

$$O\left(|V|^2+|E|+|V|\right)=O\left(|V|^2+|E|\right)=O\left(|V|^2\right)$$

שאלה 3

א. דוגמא נגדית:



קל לראות שהגרף מקיים את התנאים (1) ו-(2) ולכן הוא סדור.

האלגוריתם הנתון יתחיל מ- $w=1$, ויעבור ל- 2, כי זו הצומת עם האינדקס המינימלי שיש קשת מ-1 אליה. לאחר מכן יעבור האלגוריתם באותו אופן ל-5 – הצומת האחרונה. בסך הכל, יחזיר האלגוריתם את הערך 2, בעוד שאורך המסלול הארוך ביותר הוא 3: $1 \to 3 \to 4 \to 5$.

ב.

1. החזק מערך $S[1..n]$. $S[i]$ יכיל את האורך המסלול הארוך ביותר מ- v_i ל- v_n.
2. קבע $S[n]=0$.
3. עבור i מ- $n-1$ עד 1:
4. $T[i]=\max\{S[j]+1\}$ כאשר המקסימום נלקח על כל הצמתים v_j ($i<j\le n$) כך ש- $(v_i,v_j)\in E$.
5. החזר את $S[1]$.

הוכחת נכונות:

נוכיח באינדוקציה את הטענה הבאה: "לכל $0 \le i \le n-1$, אחרי האיטרציה ה- i של האלגוריתם, $S[n-i]$ יכיל את אורך המסלול הארוך ביותר מ- v_{n-i} ל- v_n".

בסיס: עבור $i=0$, אנו מאתחלים $S\left[n\right]=0$. המסלול היחיד האפשרי הוא המסלול הריק שאורכו 0.

נניח שהטענה נכונה עבור $0 \le i < k \le n-1$, ונוכיח שהיא נכונה עבור $i=k$. לפי הנחת האינדוקציה, $T\left[n-i\right]$ מכיל את אורך המסלול הארוך ביותר מ- v_{n-i} ל- v_n, לכל $i<k$. מכיוון שהגרף הוא סדור, מהצומת v_{n-k} יוצאות רק קשתות $\left(v_{n-k}, v_{n-i}\right)$, עבור $i<k$. אם נבחר ללכת לצומת v_{n-j}, אז אורך המסלול יהיה "האורך של המסלול הארוך ביותר מ- v_{n-j} ל- v_n, ועוד 1 עבור הקשת $\left(v_{n-k}, v_{n-j}\right)$".

אם כך, אורך המסלול יהיה מקסימלי כאשר הביטוי בגרשיים מקסימלי, ולפי הנחת האינדוקציה זה קורה בדיוק כאשר $\max\left\{S\left[j\right]+1\right\}$ מקסימלי; האלגוריתם מציב ערך זה ב- $T\left[n-k\right]$ ולכן הטענה מתקיימת עבור k.

מכאן נובע שעבור $i=n-1$, אחרי האיטרציה ה- $n-1$ והאחרונה של הלולאה, $S\left[n-\left(n-1\right)\right]=S\left[1\right]$ יכיל את אורך המסלול הארוך ביותר בין v_1 ל- v_n. את ערך זה יחזיר האלגוריתם, ומכאן נובעת נכונותו.

ניתוח סיבוכיות:

בשורות (2) (3) ו-(4) האלגוריתם עובר על כל הצמתים של הגרף, מ- v_n ועד v_1. עבור כל צומת, בוחן האלגוריתם את כל הקשתות היוצאות מהצומת, כדי למצוא את המקסימום של $\left\{S\left[j\right]+1\right\}$.

בסך הכל, האלגוריתם עובר על כל צומת בגרף בדיוק פעם אחת, וכן על כל קשת בגרף בדיוק פעם אחת (כאשר בוחנים את הצומת שהיא יוצאת ממנה). לכן בסך הכל הסיבוכיות של האלגוריתם היא:

$$O\left(|V|+|E|\right)$$

שאלה 5

נסמן ב- $S(i)$ את תת המחרוזת המכילה את i האיברים הראשונים ב- S. כמו כן נסמן $|S|=n$.

```
1. Let T be an array of size 0..|S|, initialized to zeros.
2. T[0] := 1
3. For 1 ≤ i ≤ |S|:
4.       T[i] = max_{S(i) ends with k∈L} T[i-|k|]
5. Return T[|S|].
```

<u>הוכחת נכונות</u>:

נוכיח את הטענה הבאה באינדוקציה:

לכל i, $T[i]=1$ אם ורק אם תת המחרוזת $S(i)$ היא שרשור של מחרוזות מתוך L.

<u>בסיס</u>: עבור $i=0$, $T[0]=1$ לפי שורה (2), ואכן השרשור הריק הוא שרשור של מחרוזות מתוך L.

נוכיח שאם הטענה נכונה לכל $0 \le i < l \le n$ אז הטענה נכונה עבור l.

כדי ש- $S(l)$ תהיה שרשור של מחרוזות מ- L, צריך שאחת מהמחרוזות $k \in L$ תהיינה סיפא של $S(l)$, ושהרישא הנותרת - $S(l-|k|)$ - תהיה שרשור של מחרוזות מתוך L.

מכאן, ולפי הנחת האינדוקציה, $T[l]=1$, אם ורק אם קיים $k \in L$ כך ש- $T[l-|k|]=1$. אם קיים כזה, ברור ש- $T[i]=\max_{S(l)\text{ ends with }k\in L} T[i-|k|]$ יהיה 1, ואם לא קיים מקסימום אז $T[l]$ יישאר 0.

לכן $T[l]=1$ אם ורק אם תת המחרוזת $S(l)$ היא שרשור של מחרוזות מתוך L.

מכאן, עבור $i=n$ מתקבל שהאלגוריתם מחזיר 1 אם ורק אם תת המחרוזת $S(n)=S$ היא שרשור של מחרוזות מתוך L.

ניתוח סיבוכיות:

מכיוון שגודל הא"ב קבוע, נתח את הסיבוכיות במונחים של $n=|S|$ ו- m - מספר האותיות הכולל ב- L.

שורות (1), (2) ו-(5) מתבצעות בזמן קבוע.

הלולאה שבשורה (3) רצה n פעמים.

בכל איטרציה של הלולאה, עוברים על כל $|L|$ המחרוזות ב- L, עבור כל מחרוזת $k \in L$, בודקים האם היא סיפא של $S(i)$, על ידי השוואת כל אות ב- k לאות במקום המתאים ב- S. זה לוקח $|k|$ השוואות. בסך הכל מקבלים שחישוב המקסימום דורש $O\left(|L||k|\right)=O\left(m\right)$.

לכן שורות (3) ו-(4) רצות בסיבוכיות

$$nO\left(m\right)=O\left(nm\right)$$

וזוהי הסיבוכיות הכוללת של האלגוריתם.