פתרון מועד ב': תכנון אלגוריתמים 2015

שאלה 1

סעיף א

הטענה נכונה.

נראה תחילה שהבעיה $Max - Spanning - Tree$ הינה ב-P.
להלן אלגוריתם הרץ בזמן פולינומי המכריע את השפה:
בהינתן גרף G ופונקציית משקל w, נריץ אלגוריתם למציאת עץ פורש מינימום (למשל קרוסקל) על G עם פונקציית משקל w', המוגדרת להיות $w'(e) = -w(e)$ לכל קשת $e \in E$. נחשב את משקל העץ המוחזר T.
אם $w'(T) \leq -k$ נחזיר כן, אחרת נחזיר לא.

האלגוריתם רץ בזמן פולינומי, שכן בניית פונקציית המשקל w' וחישוב משקל העץ שהתקבל ייקח זמן $O(|V| + |E|)$, וראינו גם שקרוסקל רץ בזמן פולינומי (וגודל הקלט לא השתנה). יתר על כן, קל לראות שמתקיים $w'(T') = -w(T')$ לכל עץ פורש T', ולכן קיים עץ פורש ב-G במשקל לפחות k לפי w אם"ם קיים עץ פורש ב-G במשקל לכל היותר $-k$ לפי משקל w', וזה קורה אם"ם משקל עץ פורש מינימום ב-G לפי משקל w' הינו לכל היותר $-k$. לכן האלגוריתם נכון.

כעת נסתמך על כך ש-$Max - Spanning - Tree$ ב-P על מנת להוכיח את הטענה. אם $Max - Spanning - Tree$ היא NP-קשה, זה אומר שלכל שפה A ב-NP יש רדוקציה ל-$Max - Spanning - Tree$. אזי אפשר להפעיל את האלגוריתם מבוסס הרדוקציה ל-$Max - Spanning - Tree$, ולהכריע שפה A כזו בזמן פולינומי (שכן, מימוש הקופסה השחורה באמצעות האלגוריתם הנ"ל ייקח זמן פולינומי). כיוון שזה נכון לכל שפה $A \in NP$, נובע ש-$NP \subseteq P$. יתר על כן, הוכחנו בכיתה כי $P \subseteq NP$, ולכן קיבלנו ש-$P = NP$.

סעיף ב

הטענה אינה נכונה.

יהי $G = (V, E)$ הגרף הבא:

$$V = \{s, u, v\}, \qquad E = \{(s, u), (s, v), (u, v)\}$$

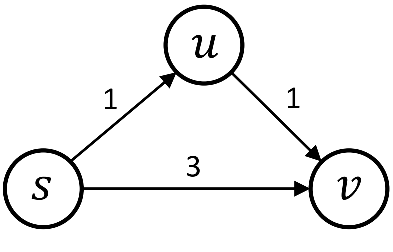
ומשקלי הקשתות:

$$w(s, u) = 1, \qquad w(s, v) = 3, \qquad w(u, v) = 1$$

בגרף זה קיים מסלול מכוון מ-s לכל קודקוד בגרף באורך (מספר קשתות) לכל היותר $k = 1$. נתבונן בריצה של איטרציה אחת של בלמן-פורד לפי סדר קשתות (משמאל לימין):

$$(u, v) \;\; (s, u) \;\; (s, v)$$

לפי האתחול $d[u] = d[v] = \infty$ בתחילת האיטרציה, ולכן $Relax(u, v)$ לא ישנה את הערך של $d[v]$.
לאחר מכן $Relax(s, u)$ ו-$Relax(s, v)$ יעדכנו את ערכי $d[v], d[u]$ כך שבסוף האיטרציה נקבל ש-$d[v] = 3, d[u] = 1$.
עם זאת, משקל מסלול קל ביותר מ-s ל-v ב-G הינו 2 (משקל המסלול $s - u - v$), בסתירה לטענה.



סעיף ג

הטענה אינה נכונה.

הערה: תשובות אשר הסתמכו על ההשערה $P \neq NP$ לא זוכו בניקוד מלא. זוהי השערה בלבד, ואיננו יודעים אם היא נכונה או לא.
לשאלה זו התקבלו תשובות שונות ומגוונות. להלן כמה פתרונות מספקים:

תשובה 1:
הסבר על סמך מבנה הנתונים.
האלגוריתם הרקורסיבי משתמש בטבלה בה יש כניסה לכל תשובה אפשרית בטווח המתאים. אנחנו יודעים כי הפרמטר k יקבל כל ערך בין 0 ל-n, ואילו הפרמטר b יכול לקבל ערכים בין $\frac{1}{2}\sum_{i=1}^{n} a_i-$ ל-$\frac{1}{2}\sum_{i=1}^{n} a_i$.
מכיוון שהמספרים a_i אינם בהכרח פולינומים ב-n (או בכלל חסומים ע"י פונקציה כלשהי של n), הטבלה יכולה להיות בגודל לא פולינומי ב-n, ואתחול הטבלה ייקח יותר מזמן פולינומי.

תשובה 2:
הסבר אינטואיטיבי לזמן ריצה עבור מספרים גדולים בקלט.
האלגוריתם הרקורסיבי מבוסס על מבנה רקורסיה של עץ בינארי בגודל 2^n, שכן עבור כל איבר a_k, הוא מתפצל לשתי קריאות שונות. עם זאת, באמצעות $memoization$, זמן הריצה יכול לרדת אם ברקורסיה יתקבלו רק מספר פולינומי של ערכים שונים של b. אך הערכים השונים מתקבלים מסכומים של תת-קבוצות של המספרים $a_1, \dots, a_n$. אם מספרים אלה גדולים דיים (למשל בעלי n ביטים או יותר) ומפוזרים באופן אקראי, אזי יהיו מעט התנגשויות (או אפילו אפס התנגשויות) בין 2^n התת-סכומים השונים, אשר יתפלגו כמו מספרים אקראיים בין 0 ל-2^n. לכן, ה-$memoization$ לא יעזור, והרקורסיה עדיין תחשב מספר אקספוננציאלי של כניסות שונות.

תשובה 3:
דוגמה נגדית מפורשת.
עבור סדרת המספרים $a_i = 2^i$, הסכום של כל תת-קבוצה $A \subseteq \{a_1, \dots, a_n\}$ הוא המספר השלם אשר בייצוג בינארי מופיעה בו הספרה 1 במקום ה-i לכל $a_i \in A$. כלומר, 2^n התת-קבוצות של $\{a_1, \dots, a_n\}$ מייצרות בדיוק את הסכומים $0, \dots, 2^n - 1$. נשים לב שבמבנה הרקורסיה, הפרמטר b מתאים לסכום של תת-קבוצה של איברים שנבחרו (קונקרטית, אם נתחיל את הרקרוסיה מהערך $b = b_0 = \frac{1}{2}\sum_{i=1}^{n} a_i$, אזי ב"עלים" של הרקורסיה (כאשר $k = 0$), נקבל כל ביטוי אפשרי עבור b מהסוג $b_0 - \sum_{a_i \in A} a_i$. באופן דומה, לא יהיו חזרות של ערכים שונים של b עבור אף k. לכן, האלגוריתם הרקורסיבי יחשב מספר אקספוננציאלי של ערכים שונים שלא יחזרו על עצמם (ולכן לא יהיה שום חיסכון מהשימוש ב-$memoization$).

שאלה 2

סעיף א

השפה הינה NP-קשה.
נראה רדוקציה $SAT \leq_p Multi - SAT$:

הרדוקציה: בהינתן קלט פסוק CNF φ מעל n משתנים הפונקציה תחשב ותחזיר את פסוק ה-CNF:

$$\psi = \varphi \wedge (t_1 \vee \overline{t_1}) \wedge \dots \wedge (t_n \vee \overline{t_n})$$

כאשר $t_1, \dots, t_n$ משתנים חדשים.
פולינומיאליות הרדוקציה: הוספת n פסוקיות, כל אחת בגודל קבוע. פולינומי בגודל הקלט.
נכונות הרדוקציה: צ"ל $\varphi \in SAT \Leftrightarrow \psi \in Multi - SAT$
$\Leftarrow$ נניח $\psi \in \mathrm{Multi-SAT}$, אזי קיימת הצבה מספקת τ ל-ψ (קיימות הרבה. ניקח אחת). הצבה זו מספקת כל פסוקית ב-ψ, ובפרט את הפסוקיות ב-φ. לכן צמצום הצבה זו רק למשתנים של φ מספקת פסוק זה. מצאנו הצבה מספקת ל-φ ולכן $\varphi \in SAT$.

$\Leftarrow$ נניח $\varphi \in SAT$, אזי קיימת הצבה מספקת π ל-φ. ניתן להרחיב הצבה זו להצבה ל-ψ ע"י נתינת ערך למשתנים $t_1, \dots, t_n$ (נזכור כי אלו אינם מופיעים ב-φ). נתבונן ב-τ, הרחבה כלשהי כנ"ל. אזי הפסוקיות ב-φ עדיין מסתפקות תחת השמה זו. בנוסף, כל פסוקית $(t_i \vee \overline{t_i})$ מסתפקת לכל הצבה ל-t_i. קיבלנו כי τ הצבה מספקת ל-ψ. מספר הדרכים להרחיב את π הוא כמס' הדרכים לתת ערכים למשתנים $t_1, \dots, t_n$, כלומר 2^n.
נשים לב כי ב-ψ יש $2n$ משתנים (n ישנים ו-n חדשים), ולכן קיבלנו כי יש ל-ψ לפחות $2^n = 2^{2n/2}$ השמות מספקות שונות, ולכן $\psi \in Multi - SAT$.

<u>סעיף ב</u>

השפה הינה ב-P.
נראה אלג' פולינומי המכריע את השפה:
<u>האלגוריתם</u>: בהינתן קלט $(G = (V, E), U, k)$:

1. בדוק כי $U \subseteq V$, וכן $|U| = \log |V|$, ובנוסף $k \leq |U|$. אם אחד מאלה לא מתקיים דחה.
2. לכל קבוצה $C \subseteq U$ בגודל k בדוק האם C כיסוי בצמתים של G.
3. אם נמצאה C כנ"ל קבל, אחרת דחה.

<u>נכונות האלגוריתם</u>: אם $(G = (V, E), U, k) \in Log - VC$ אזי $|U| = \log |V|$, וכן קיימת $C \subseteq U$ מגודל k שהיא כיסוי בצמתים לגרף, ובפרט $k \leq |U|$. נקבל כי האלג' לא ידחה בשלב 1, ובשלב 2 ימצא קבוצה כנדרש ולכן יקבל. בכיוון השני, אם האלג' קיבל אזי $|U| = \log |V|$ (שלב 1) וכן קיימת קבוצה $C \subseteq U$ מגודל k המהווה כיסוי בצמתים לגרף (שלב 2), ולכן לפי הגדרה $(G = (V, E), U, k) \in Log - VC$.
<u>זמן ריצה</u>: שלב 1 בזמן לינארי.
עבור קבוצה $C \subseteq V$ כלשהי, בדיקה כי היא כיסוי בצמתים בזמן לינארי (מעבר על כל הקשתות, ובדיקה כי כל קשת מכילה איבר מ-C). מס' הקבוצות שנבדקות בשלב 2 הוא כמס' הקבוצות בגודל k מתוך U, וזה חסום ע"י מס' כל תתי הקבוצות של U, והוא $2^{|U|} = 2^{\log|V|} = |V|$. לכן סה"כ זמן הריצה בשלב 2 פולינומי בגודל הקלט, וקיבלנו כי זמן ריצת האלג' כולו פולינומי בקלט.

<u>שאלה 3</u>

<u>סעיף א</u>

דוגמה למופע עבורו האלגוריתם יחזיר פתרון שאינו אופטימלי:

$$I_1 = [0,0.5], I_2 = [0.5,1], I_3 = [0.05,0.95]$$

האלגוריתם יבחר ראשית ב-I_3, ולאחר מכן יבחר ביתר המקטעים, בעוד הפתרון האופטימלי מכיל שני מקטעים בלבד.

<u>סעיף ב</u>

סימון: עבור כל קטע I_j נגדיר את זמן תחילתו וסיומו כ-s_j ו-f_j בהתאמה.

האלגוריתם:

1. איתחול: $B = \emptyset$, $d = 0$.
2. כל עוד $d < 1$:
   2.1 יהי j אינדקס הקטע בעל זמן הסיום f_j המאוחר ביותר מבין הקטעים המקיימים $d \in I$.
   2.2 עדכן $B = B \cup \{j\}$ ו-$d = f_j$.
3. החזר את B.

הוכחת נכונות האלגוריתם:

סימון: נסמן ב-B_i ו-d_i את ערכם של B ו-d בהתאמה בתום האיטרציה ה-i.

<u>אבחנה 1:</u> לכל $x \in [0,1)$ קיים קטע $\mathrm{I}_j = [s_j, f_j]$ בקלט כך ש-$s_j \leq x < f_j$.

איחוד הקטעים מכסה את $[0,1]$, ואם כל קטע I_j המכיל את x מקיים $x = f_j$ אזי קיים $\epsilon > 0$ קטן דיו כך ש-$x + \epsilon$ אינו מכוסה ע"י הקטעים בקלט, וזאת משום שיש מס' סופי של קטעים, וכולם סגורים.

<u>אבחנה 2:</u> האלגוריתם עוצר לאחר לכל היותר n איטרציות, שכן מאבחנה 1 כל עוד $d < 1$ הוא גדל ממש בכל איטרציה. מאחר ומס' הקטעים סופי, תוך מס' סופי של צעדים נקבל $d = 1$ והאלגוריתם עוצר.

<u>אבחנה 3:</u> בכל שלב, הקבוצה B_i מכסה את המקטע $[0, d_i]$.

<u>טענת עזר:</u> בכל שלב קיים פתרון אופטימלי O המקיים $B_i \subseteq O$.

<u>טענה ראשית:</u> האלגוריתם מחזיר פתרון אופטימלי.

<u>הוכחת הטענה הראשית:</u> מאבחנה 2, האלגוריתם עוצר כאשר $d = 1$ ומחזיר את קבוצת המקטעים B. מאבחנה 3 B מכסה את $[0,1]$, ולכן הפתרון המוחזר חוקי. מטענת העזר $B \subseteq O$ עבור פתרון אופטימלי כלשהו, ולכן בהכרח $|B| = |O|$ (אחרת נקבל סתירה לאופטימליות O). כלומר B אופטימלי.

<u>הוכחת טענת העזר:</u> נוכיח באינדוקציה על מספר האיטרציות i שהושלמו ע"י האלגוריתם.

בסיס: $i = 0$, כלומר $B_0 = \emptyset$ ולכן מתקיים $B_0 \subseteq O$ לכל קבוצה O ובפרט עבור O אופטימלי.

הנחה: נניח כי קיים פתרון אופטימלי כלשהו O כך ש-$B_{\mathrm{i}} \subseteq O$.

צעד: נוכיח כי קיים פתרון אופטימלי O' כך ש-$B_{i+1} \subseteq O'$. יהא j המקטע אשר נוסף לקבוצה B באיטרציה ה-$i+1$, כלומר $B_{i+1} = B_i \cup \{j\}$.

אם $j \in O$, נקבל מהנחת האינדוקציה כי $B_{i+1} \subseteq O$ ולכן נוכל לסמן $O' = O$.

אחרת נגדיר $A = \{k \in O \mid d_i \in I_k\}$, כלומר קבוצת כל המקטעים ב-O אשר מכילים את d_i. יהי j^* המקטע בעל זמן הסיום המקסימלי ב-A. נשים לב כי $A \neq \emptyset$ מאחר והמקטעים ב-A מכסים את $[0,1]$. ממקסימליות זמן סיום j מתקיים $f_{j^*} \leq f_j$. נשים לב כי $j^* \notin B_i$, זאת כי בהכרח קיים מקטע ב-O המכיל את d_i ומסתיים אחריו (אותו טיעון כבאבחנה 1, זאת כי $d_i < 1$), ולפי אבחנה 3 הקטעים ב-B_i מכסים רק את $[0, d_i]$.

נגדיר $O' = O \setminus \{j^*\} \cup \{j\}$. נשים לב כי $B_{i+1} \subseteq O'$, ולכן לפי אבחנה 3 O' מכסה את $[0, f_j = d_{i+1}]$.

בנוסף, לכל $x \in (f_j, 1] \subseteq (f_{j^*}, 1]$ היה קיים מקטע ב-O ששונה מ-I_{j^*} המכסה את x (שכן $x > f_{j^*}$), ולכן מקטע זה קיים גם ב-O'. קיבלנו כי O' פתרון חוקי. בנוסף גודלו זהה לגודלו של הפתרון האופטימלי O, ולכן אף הוא אופטימלי בגודלו כנדרש.

ניתוח זמן הריצה:

ננתח את זמן ריצת האלגוריתם תחת ההנחה שהוא ממומש בצורה נאיבית באמצעות מערך.

מאבחנה 1, יתכנו לכל היותר n איטרציות. בכל איטרציה מתבצע מעבר על כל מקטעי הקלט ובדיקה לכל מקטע האם הוא מכיל את d. מציאת המקטע בעל זמן הסיום המקסימלי הינה לינארית בגודל הקבוצה (לכל היותר n).

לכן כל איטרציה ב-$O(n)$. סה"כ נקבל כי זמן ריצת האלגוריתם הינו $O(n^2)$.

<u>שאלה 4</u>

<u>סעיף א</u>

⇐ יהי $(S,V\setminus S)$ חתך-(a,b) מכוון ב-G. נניח בשלילה כי קיים מסלול $(b=v_0,\dots,v_k=a)$ מ-b ל-a ב-G. יהי v_i הקודקוד האחרון במסלול כך ש-$v_i\in V\setminus S$. קיים כזה כי $b\in V\setminus S$. בנוסף v_i אינו הקודקוד האחרון במסלול כי $a\in S$. לכן $v_{i+1}\in S$ וקיבלנו כי (v_i,v_{i+1}) קשת חוצה נגד כיוון החתך, בסתירה.

⇒ נניח כי לא קיים מסלול מכוון מ-b ל-a. נגדיר $V\setminus S$ להיות כל הקודקודים הנגישים מ-b (ו-S שאר הקודקודים). לפי הגדרה $b\in V\setminus S$. בנוסף נתון כי a לא נגיש מ-b, ולכן $a\in S$. לא קיימת קשת (u,v) כך ש-$u\in V\setminus S$ וכן $v\in S$, אחרת לפי הגדרת $V\setminus S$ קיים מסלול P מ-b ל-u, ולכן $P\circ(u,v)$ מסלול מ-b ל-v, כלומר v נגיש מ-b, ולכן $v\in V\setminus S$ בסתירה לכך ש-$v\in S$. קיבלנו כי $(S,V\setminus S)$ חתך-(a,b) מכוון כנדרש.

<u>סעיף ב</u>

<u>אלגוריתם</u>: נגדיר את B להיות כל הקודקודים הנגישים מ-b.
אם $a\in B$ נחזיר "לא קיים חתך-(a,b) מכוון". אחרת נחזיר $(V\setminus B,B)$.
<u>זמן ריצה</u>: הגדרת B ע"י ריצת BFS מ-b בזמן $O(|V|+|E|)$. בדיקת $a\in B$ ויצירת $V\setminus B$ בזמן $O(|V|)$.
תחת ההנחה $|V|\le|E|$ זמן ריצה כולל $O(|E|)$.
<u>נכונות</u>: לפי סעיף א, אם $a\in B$ אזי קיים מסלול מ-b ל-a ולכן לא קיים חתך-(a,b) מכוון. אחרת קיים חתך-(a,b), ובהוכחה בסעיף א הראינו כי החתך המוגדר ע"י האלג' (כל הקודקודים הנגישים מ-b) עונה להגדרת חתך-(a,b) מכוון.

<u>סעיף ג</u>

סימון: משקל חתך $w(S,T)$ יתייחס לגרף G, וקיבולת חתך $c(S,T)$ תתייחס לגרף G'.
<u>ט.עזר</u>: יהי (S,T) חתך ב-G (כלומר $a\in S$ וכן $b\in T$). אזי:

1. אם (S,T) חתך לא מכוון ב-G אזי $c(S,T)=\infty$.
2. אם (S,T) חתך-(a,b) מכוון ב-G אזי $c(S,T)=w(S,T)$ (ובפרט קיבולו קטן מ-∞).

<u>הוכחת טענה ראשית</u>

⇐ יהי (S,T) חתך-(a,b) מכוון מינימום עבור (G,w). נניח בשלילה כי קיים חתך אחר (S',T') כך ש-$c(S',T')<c(S,T)$ (*). בפרט $c(S',T')<\infty$, ולכן לפי ט.עזר זהו חתך-(a,b) מכוון ב-G (אחרת קיבולו היה ∞). לפי ט.עזר על שני החתכים: $w(S',T')=c(S',T')<_{(*)}c(S,T)=w(S,T)$. קיבלנו כי (S',T') חתך-(a,b) מכוון בעל משקל קטן יותר מ-(S,T), בסתירה למינימליות (S,T).

⇒ יהי (S,T) חתך מינימום ברשת הזרימה. נתון כי קיים חתך-(a,b) מכוון ב-G, ולפי ט.עזר קיבולו סופי. ממינימליות (S,T) נקבל כי קיבולו סופי גם כן, ולכן לפי ט.עזר הוא בהכרח חתך-(a,b) מכוון ב-G (אחרת קיבולו היה ∞). נניח בשלילה כי קיים חתך-(a,b) מכוון (S',T') כך ש-$w(S',T')<w(S,T)$. לפי ט.עזר על שני החתכים: $c(S',T')=w(S',T')<w(S,T)=c(S,T)$, בסתירה למינימליות הקיבולת של (S,T).

<u>הוכחת ט.עזר</u>

1. (S,T) חתך לא מכוון, לכן קיימת קשת $(u,v)\in E$ כך ש-$u\in T$ וכן $v\in S$. לפי בניית G' קיימת קשת $(v,u)\in E''$ וקיבולה ∞. לכן $c(S,T)=\sum_{x\in S,y\in T}c(x,y)=\infty$.
2. (S,T) חתך-(a,b) מכוון. נראה כי לא קיימת קשת $(u,v)\in E''$ חוצה חתך (כלומר $u\in S$ ו-$v\in T$). נניח בשלילה כי קיימת כזו, אזי לפי הבניה $(v,u)\in E$. מצאנו קשת נגד כיוון החתך ב-G, בסתירה לכך שזהו חתך-(a,b) מכוון. נסיק מהבניה כי הצלעות היחידות שחוצות את (S,T) ב-G' וב-G זהות (הצלעות מ-E), וכן משקלן שווה לקיבולן לפי הגדרת c, ולכן $c(S,T)=w(S,T)$.