תאריך הבחינה: 18.9.2015
שמות המרצים: פרופ' עמוסביימל
פרופ' יפים דיניץ
מר רן טייג
דר' עדן כלמטץ'
מר אורי שטמר
גב' מיכל שמש

שם הקורס: תכנון אלגוריתמים
מספר הקורס: 202-1-2041
שנה: 2015 סמסטר: ב' מועד: ג'
משך הבחינה: **3.5 שעות**
חומר עזר: **אסור**

תכנון אלגוריתמים
מבחן מועד ג'

אנא קיראו היטב את ההראות שלהלן:

- בטופס הבחינה **4** עמודים כולל עמוד זה. ודאו כי כולם נמצאים בידכם.
- המבחן הינו ללא חומר עזר.
- משך המבחן 3½ שעות.
- סה"כ נקודות 100.
- פתרו את המבחן תחילה במחברת הטיוטה. לאחר מכן העתיקו את התשובות למקום המיועד בטופס התשובות. **בדיקת המבחן לא תביא בחשבון את מחברת הטיוטה או תוספות בגב העמוד.** מחברת הטיוטא מיועדת לגריסה!
- רשמו את מספר הנבחן בראש כל דף.
- המבחן מורכב מ- 4 שאלות, יש לענות על כל השאלות.
- לסדר הופעת השאלות בטופס או לניקוד אין בהכרח קשר לקושי השאלה.
- מותר להשתמש במבני נתונים ידועים מבלי לפרט את מימושם.
- מותר להשתמש באלגוריתמים ידועים (כולל מתרגולים) מבלי לפרט את מימושם.
- כל שימוש בתוצאה **מעבודות הבית** דורשת הוכחה מלאה.
- **ניתן להשתמש בטענות של סעיפים קודמים אפילו אם לא פתרתם אותם**.
- טענות ללא נימוק לא תתקבלנה.
- ניתן להסתמך על טענות ומשפטים מהכיתה ומהתרגולים, אך יש לנסח אותם במדויק.
- **אם לא מצוין במפורש אחרת, על תיאור אלגוריתם לכלול ניתוח זמן ריצה והוכחת נכונות.**
- **במידה ואינכם יודעים את התשובה לסעיף כלשהו, רשמו "לא יודעים" ותזכו ב- 20% מניקוד הסעיף.**
- מותר להשתמש בעיפרון, אך במידה והינכם עושים זאת וודאו כי מה שכתבתם הינו קריא וברור.
- מומלץ מאוד לבדוק את עבודתכם לפני הגשתה.

בהצלחה!

שאלה 1 [25 נקודות]

נגדיר את בעיית המסלול הקל ביותר באורך k:

קלט: גרף מכוון $G=(V,E)$, פונקציית משקל $w: E \to \mathbb{R}$, קודקוד $s \in V$ ומספר טבעי $k \geq |V|$.

פלט: לכל קודקוד $v \in V$ יש למצוא <u>משקל</u> מסלול קל ביותר מבין כל המסלולים מ- s ל- v המכילים <u>לכל היותר</u> k קשתות (המסלול לא בהכרח פשוט).

סעיף א [7 נקודות]

לפתרון הבעיה הנ"ל הוצע להשתמש בגרסה של אלגוריתם של בלמן-פורד שתעצור אחרי k שלבים, כלומר באלגוריתם הבא:

```
INITIALIZE-SINGLE-SOURCE(G, s)
  for each vertex v ∈ V do
      d[v] ← ∞
  d[s] ← 0

RELAX(u, v, w)
  if d[v] > d[u] + w(u, v) then
     d[v] ← d[u] + w(u, v)

BELLMAN-FORD-k (G,s,w,k)
  Initialize-Single-Source(G,s)
  for i = 1 to k do
     for each edge (u,v) ∈ E do
           Relax(u,v,w)
```

הראו דוגמא נגדית בה האלגוריתם לא יחזיר פתרון נכון לבעיה. הסבירו את תשובתכם.

סעיף ב [18 נקודות]

תארו אלגוריתם בסיבוכיות $O(|E||V|)$ הפותר את בעיית המסלול הקל ביותר באורך k. נתחו את סיבוכיות האלגוריתם. אין צורך להוכיח את נכונות האלגוריתם.

שאלה 2 [24 נקודות]

סימונים:
תהי $N=(G=(V,E),c,s,t)$ רשת זרימה, ותהי f זרימה חוקית ברשת N. נאמר ש- f היא זרימת מסלול אם קיים מסלול מכוון פשוט P ב- G כך ש- P הוא מסלול מ- s ל- t או מ- t ל- s, וכך שעבור קבוע $a>0$ כלשהו מתקיים

$$f(u,v)=\begin{cases} a & , \quad (u,v)\in P \\ -a & , \quad (v,u)\in P \\ 0 & , \quad \text{אחרת} \end{cases} \qquad (1)$$

באופן דומה, נאמר ש- f היא זרימת מעגל אם משוואה (1) מתקיימת עבור מעגל פשוט P ב- G.
בשאלה זו ניתן להסתמך על המשפט הבא ללא הוכחה:
משפט:
תהי $N=(G=(V,E),c,s,t)$ רשת זרימה, ותהי f זרימה חוקית ברשת N. אזי ישנה הצגה של f כסכום של לכל היותר $|E|$ זרימות מסלולים ו/או זרימות מעגלים. כלומר, קיים $t\leq |E|$ וקיימות זרימות $g_1,g_2,\dots,g_t$ כך שמתקיים:
א. $f=g_1+g_2+\cdots+g_t$
ב. לכל $1\leq i\leq t$ מתקיים ש- g_i היא או זרימת מסלול או זרימת מעגל.

היזכרו כי האלגוריתם של *Ford & Fulkerson* מתחזק זרימה חוקית, ובכל שלב מוצא מסלול שיפור כלשהו ומעדכן את הזרימה לאורכו. ראינו כי אם כל הקיבולים ברשת שלמים אז האלגוריתם עוצר, אך לא בזמן פולינומי. בשאלה זו נשנה מעט את האלגוריתם כך שבכל שלב ימצא מסלול שיפור עם צוואר בקבוק שיורי גדול ביותר.

נסמן את צוואר הבקבוק השיורי של מסלול P ברשת N יחסית לזרימה f ע"י
$C_f(P)=\min_{e\in P}\{c(e)-f(e)\}$.

נתון האלגוריתם הבא:

קלט: רשת זרימה $N=(G,c,s,t)$.
1. אתחל $f(u,v)=0$ לכל $u,v\in V$
2. כל עוד קיים מסלול שיפור ברשת N יחסית לזרימה f:
   א. יהי P מסלול שיפור כך ש- $C_f(P)=max\{C_f(P') : P' \text{ מסלול שיפור}\}$.
   ב. לכל $(u,v)\in P$ בצע:
   $$f(u,v)\leftarrow f(u,v)+C_f(P)$$
   $$f(v,u)\leftarrow f(v,u)-C_f(P)$$
3. החזר את f.

סעיף א [12 נקודות]

- נסמן ב- $|f_0^{max}|$ את גודל זרימת מקסימום ברשת N.
- נסמן ב- f_1 את הזרימה ברשת לאחר האיטרציה הראשונה של צעד 2 ונסמן ב- $|f_1^{max}|$ את גודל זרימת מקסימום ברשת N_{f_1}.

הוכיחו כי $|f_1^{max}|\leq |f_0^{max}|\left(1-\frac{1}{|E|}\right)$.

סעיף ב [12נקודות]

הסבירו כיצד לממש בצורה יעילה ככל האפשר את צעד 2א של האלגוריתם, כלומר, הראו איך למצוא מסלול שיפור עם צוואר בקבוק שיורי מקסימלי. נתחו את זמן ריצת המימוש. אין צורך להוכיח את נכונות האלגוריתם.

שאלה 3 [25 נקודות]

נגדיר את בעיית אריזת וקטורים (VP), בה נתונים n וקטורים $\overrightarrow{a_1},\overrightarrow{a_2},\dots,\overrightarrow{a_n}$, כאשר $\overrightarrow{a_i}\in[0,1]^d$ לכל $1\le i\le n$. כלומר, כל $\overrightarrow{a_i}$ הוא וקטור באורך d שכל הקואורדינטות שלו הן מספרים בתחום $[0,1]$, ונסמן את הקואורדינטה הַ-t-ית של $\overrightarrow{a_i}$ ע"י $a_i[t]$. נאמר כי תת-סדרה של וקטורים $\overrightarrow{a_{\iota_1}},\overrightarrow{a_{\iota_2}},\dots,\overrightarrow{a_{\iota_\ell}}$, כאשר $i_1<i_2<\cdots<i_\ell$, היא חסומה אם בכל קואורדינטה t סכום הערכים של הוקטורים בתת-סדרה בקואורדינטה ה- t-ית הינו לכל היותר 1. נאמר כי $\langle\overrightarrow{a_1},\overrightarrow{a_2},\dots,\overrightarrow{a_n},\ell\rangle\in\mathrm{VP}$ אם קיימת תת-סדרה חסומה של ℓ וקטורים $\langle\overrightarrow{a_{\iota_1}},\overrightarrow{a_{\iota_2}},\dots,\overrightarrow{a_{\iota_\ell}}\rangle$.

לדוגמא, נתבונן בוקטורים $\overrightarrow{a_1}=\begin{pmatrix}1/8\\5/12\end{pmatrix},\overrightarrow{a_2}=\begin{pmatrix}3/4\\1/2\end{pmatrix},\overrightarrow{a_3}=\begin{pmatrix}1/6\\7/12\end{pmatrix}$. תת-הסדרה $\overrightarrow{a_1},\overrightarrow{a_3}$ היא חסומה מכיוון שסכום הוקטורים בקואורדינטה הראשונה הוא $\frac{1}{6}+\frac{1}{8}=\frac{7}{24}\le 1$ ובשנייה הוא $\frac{7}{12}+\frac{5}{12}=1$.
לעומת זאת, תת-הסדרה $\overrightarrow{a_2},\overrightarrow{a_3}$ איננה חסומה כיוון שסכום הוקטורים בקואורדינטה השנייה הוא $\frac{7}{12}+\frac{1}{2}=1\frac{1}{12}>1$.

סעיף א [5 נקודות]

הראו שֶ- VP בְּ- NP.

סעיף ב [20 נקודות]

הראו רדוקציה $\mathrm{IndependentSet}\le_{\mathrm{P}}\mathrm{VP}$, כאשר IndependentSet היא השפה המוגדרת בדפי העזר.

שאלה 4 [26 נקודות] – שאלת הוכיחו/הפריכו. אין קשר בין סעיפים שונים בשאלה.

סעיף א [8 נקודות]

תהי $A\in NP$ כלשהי. נאמר שאלגוריתם $\mathcal{B}$ הוא אלגוריתם אקראי פולינומי עבור השפה A אם:
1. קיים קבוע c כך שלכל קלט x, זמן הריצה של $\mathcal{B}(x)$ בכל ריצה הוא לכל היותר $|x|^c$.
2. אם $x\in A$ אזי $\mathcal{B}(x)$ מחזיר "כן" בהתסברות לפחות $3/4$.
3. אם $x\notin A$ אזי $\mathcal{B}(x)$ מחזיר "לא" בהתסברות לפחות $3/4$.

נניח כי ל- SAT יש אלגוריתם אקראי פולינומי.
הוכיחו הפריכו: לכל שפה $A\in NP$ קיים אלגוריתם אקראי פולינומי.

סעיף ב

יהי $G=(V,E)$ גרף פשוט, לא מכוון, ממושקל, עם לפחות 4 צלעות. נסמן את צלעות הגרף כ- $e_1,e_2,\dots,e_{|E|}$ כאשר $w(e_1)\le w(e_2)\le\cdots\le w(e_{|E|})$. ידוע כי $w(e_2)<w(e_3)<w(e_4)$.

סעיף ב1 [5 נקודות]

הוכיחו או הפריכו: הצלעות e_1,e_2 נמצאות בכל עפ"מ של G.

סעיף ב2 [5 נקודות]

הוכיחו או הפריכו: הצלעות e_1,e_2,e_3 נמצאות בכל עפ"מ של G.

סעיף ג [8 נקודות]

יהי $G=(V,E)$ גרף מכוון ויהיו $u,v\in V$ זוג קודקודים. ידוע שקיימת ריצת *DFS* על G בה u צאצא של v וידוע כי קיימת ריצת *DFS* על G בה v צאצא של u.
הוכיחו או הפריכו: בכל ריצת *DFS* על G מתקיים: או ש- u צאצא של v, או ש- v צאצא של u.