



אוניברסיטת אריאל בשומרון

פקולטה: מדעי הטבע

מחלקה: מדעי המחשב

שם הקורס: אלגוריתמים כלכליים

קוד הקורס: 2-7062310 קבוצה 1

סמסטר _____ ב _____ מועד _____ ב _____ תאריך בחינה: כ"א תמוז ה'תשע"ט, 24/7/2019

משך הבחינה: 3 שעות

שם המרצה: אראל סגל-הלוי

בבחינה חמש שאלות. סכום הנקודות בכל השאלות הוא 120. מותר לענות על כל השאלות.

- מי שיצבור 100 נקודות ומעלה, ציונו במבחן יהיה 100.

יש לענות במחברת הבחינה.

- אין צורך להעתיק את השאלון למחברת - השאלון יתפרסם במודל לאחר הבחינה.

חומר עזר מותר בשימוש: דף-נוסחאות אישי בגודל פוליו (A4). מותר לכתוב משני צדי הדף.

- יש להגיש את דף-הנוסחאות יחד עם מחברת הבחינה.

הנחיות לפתרון שאלות תיכנות:

- מותר לכתוב קוד בכל שפה שאתם מכירים או בפסאודו-קוד.
- יש לתעד את הקוד ולהסביר היטב בעברית מה עושה כל שורה ואיך זה מתאים לאלגוריתם.
- אתם לא נבחנים על התחביר של השפה אלא על האלגוריתם. העיקר שאוכל להבין מה עשיתם.

הנחיות כלליות:

- יש להסביר כל תשובה בפירוט. ניקוד מלא יינתן רק על תשובה נכונה עם הסבר נכון.
- אם נראה לכם ששאלה כלשהי אינה מוגדרת עד הסוף (חסרות הנחות מסויימות), הניחו את ההנחות שנראות בעיניכם הגיוניות ביותר בהתאם לשאלה.

בהצלחה!!

שאלה 1 [30 נק']. שיבוץ תורניות

בפיצרייה "פיצות כלכליות" ישנן 15 משמרות בשבוע ו-15 עובדים. ההנהלה שיבצה באקראי כל עובד לאחת מ-15 המשמרות השבועיות. כמה עובדים באו להנהלה וביקשו להחליף משמרת. ההנהלה הזמינה אתכם לפתח תוכנה שתאפשר לעובדים להחליף משמרות באופן עצמאי, בלי להטריח את המנהלים. המערכת תפנה לכל עובד, תבקש ממנו לרשום את סדר-העדיפות שלו על המשמרות (מה המשמרת שהוא הכי רוצה, השניה, השלישית וכו'), ותחזיר שיבוץ חדש העומד בדרישות הבאות:

1. אף עובד לא יקבל משמרת גרועה יותר מהמשמרת הנוכחית שלו.
2. אף עובד לא יכול להרויח מכך ש"ירגל" אחרי העובדים האחרים וינסה לגלות את סדר העדיפות שלהם.
3. אף קבוצת-עובדים לא יכולה להרויח מכך שהעובדים בקבוצה יסרבו להירשם למערכת ויבצעו החלפות בינם לבין עצמם.

א [10 נק']. ציינו שם של אלגוריתם המתאים לדרישות ההנהלה. ציינו שלוש תכונות של האלגוריתם, והתאימו אותן לשלוש הדרישות של ההנהלה.

- פתרון: אלגוריתם מעגלי המסחר. תכונות האלגוריתם בהתאמה: 1. רציונלי ליחידים, 2. אמיתי, 3. יציב.

ב [10 נק']. הניחו ש $n=5$;

- המשמרות הן: שחרית, בוקר, צהריים, ערב, לילה;
- העובדים הם: רחל, אברהם, יצחק, יעקב, דינה.
- השיבוץ הראשון הוא: שחרית-רחל, בוקר-אברהם, צהריים-יצחק, ערב-יעקב, לילה-דינה.

העדפות העובדים הם (מהמשמרת המועדפת למשמרת הפחות מועדפת):

- רחל: בוקר, לילה, צהריים, ערב, שחרית
- אברהם: צהריים, ערב, לילה, שחרית, בוקר
- יצחק: ערב, לילה, שחרית, בוקר, צהריים
- יעקב: בוקר, צהריים, ערב, לילה, שחרית
- דינה: ערב, שחרית, בוקר, צהריים, לילה

הדגימו את פעולת האלגוריתם מסעיף א.

- הסבירו בפירוט כל שלב.
- כיתבו את השיבוץ הסופי המתקבל לאחר סיום האלגוריתם.

פתרון: מציירים גרף שבו כל עובד מצביע למשמרת שהוא הכי רוצה, וכל משמרת מצביעה לעובד שמשוּבץ לה עכשיו. בגרף הראשון יש רק מעגל אחד: אברהם – צהריים – יצחק – ערב – יעקב – בוקר – אברהם. לכן מתבצעת ההחלפה לפי מעגל זה, והשיבוץ שנקבע הוא: אברהם-צהריים, יצחק-ערב, יעקב-בוקר. אחרי שמוחקים את שלושת העובדים והמשמרות הנ"ל מהגרף, נשארות ההעדפות הבאות:

- רחל: לילה, שחרית
- דינה: שחרית, לילה

בגרף זה יש מעגל אחד: רחל – לילה – דינה – שחרית – רחל.

לכן מתבצעת ההחלפה לפי מעגל זה, והשיבוץ שנקבע הוא: **רחל-לילה, דינה-שחרית**.

ג [10 נק']. הדגימו את תכונה מספר 2 של האלגוריתם על הקלט הנ"ל. כלומר: הוכיחו, שאף אחד מחמשת העובדים בדוגמה לא יכול להרויח מכך ש"ירגל" אחרי עובדים אחרים וינסה לגלות את סדר העדיפויות שלהם (בהנחה שהעובדים האחרים לא מרגלים).

פתרון:

- אברהם, יצחק ויעקב משובצים כל אחד למשמרת שהוא הכי רוצה, ולכן ודאי שהם לא יכולים להרויח מריגול.
- רחל משובצת למשמרת שחרית, אבל אף עובד אחר לא שם את משמרת זו בעדיפות ראשונה. לכן, לא משנה מה רחל תעשה, היא לא תהיה חלק מהמעגל הראשון. המעגל הראשון יהיה כמו בסעיף ב, והמשמרות ערב בוקר וצהריים ייצאו מהגרף. לכן, לא משנה מה רחל תעשה, היא לא תקבל את המשמרת שהיא הכי רוצה (בוקר). המשמרת הטובה ביותר שהיא יכולה לקבל היא השניה בסדר העדיפות שלה – משמרת לילה – וזה בדיוק מה שהיא מקבלת כשהיא לא מרגלת.
- ההוכחה לגבי דינה דומה להוכחה לגבי רחל.

שאלה 2 [30 נק']. חלוקת זמן נסיעה במונית

שלושה חברים ממטולה נכנסים למונית בשעה 6:00 בבוקר ונוסעים לכיוון אילת. משך הנסיעה – 9 שעות. במונית יש שני מושבים אחוריים ומושב אחד קדמי. מהמושב הקדמי רואים טוב יותר את הנוף, ולכן כולם רוצים לשבת בו. כדי שלא יריבו, הנהג מציע שיחלקו ביניהם את זמן-הנסיעה לשלושה חלקים שווים – בין 6:00 ל 9:00, בין 9:00 ל 12:00 ובין 12:00 ל-15:00, כך שכל אחד יישב מקדימה 3 שעות בדיוק. החברים לא מסכימים, כי זה חשוב להם באיזה חלק של הנסיעה הם יושבים מקדימה – יש הבדל בין לשבת מקדימה בתחילת הנסיעה, באמצעה או בסופה (הנוף שונה, רמת העייפות שונה, וכו'...).

הציעו לחברים אלגוריתם שבעזרתו יוכלו לחלק את זמן הנסיעה ביניהם. האלגוריתם צריך לקיים את התכונות הבאות:

1. כל אחד מהחברים חושב, שהזמן שבו הוא יושב מקדימה, טוב לפחות כמו הזמנים של שני החברים האחרים.
2. האלגוריתם חייב לתת פתרון מדויק ולא מקורב, והוא חייב להסתיים תוך מספר צעדים קטן.
3. כל חבר יושב מקדימה במשך פרק-זמן רצוף, או שני פרקי-זמן רצופים, אבל לא יותר (כך שלא יצטרכו יותר מדי פעמים לעצור את הרכב ולהתחלף ביניהם).
4. האלגוריתם לא משתמש בכסף (דמי-הנסיעה שולמו מראש על-ידי ההורים; החברים נוסעים בלי ארנק).

א [10 נק']. ציינו שם של אלגוריתם המקיים את הדרישות. ציינו את תכונות של האלגוריתם והתאימו לתכונות למעלה.

- פתרון: אלגוריתם סלפרידג'-קונוויי. האלגוריתם משמש לחלוקה הוגנת של "עוגה" בין שלושה אנשים. במקרה שלנו, העוגה היא זמן הנסיעה. תכונות האלגוריתם:
 1. חלוקה ללא קנאה,
 2. באלגוריתם יש שני שלבים, כל אחד מהם כולל חיתוך אחד או שניים.
 3. בכל שלב, כל אחד מהשחקנים מקבל "פרוסת-עוגה" רציפה, ולכן במקרה שלנו כל חבר מקבל לכל היותר שני פרקי-זמן רציפים.
 4. האלגוריתם לא משתמש בכסף – הוא מחלק רק את ה"עוגה".

ב [10 נק']. נתונה טבלה המתארת את רמת האושר שכל חבר מרגיש כאשר הוא יושב מקדימה בכל שעה של הנסיעה.

שעה:	7 - 6	8 - 7	9 - 8	10 - 9	11 - 10	12 - 11	13 - 12	14 - 13	15 - 14
חבר א:	4	1	1	1	1	1	1	1	1
חבר ב:	1	3	3	3	3	4	4	4	4
חבר ג:	2	2	2	2	2	2	2	2	2

הדגימו את פעולת האלגוריתם מסעיף א. תארו בפירוט כל שלב באלגוריתם, ופרטו את התוצאה הסופית – מי יושב מקדימה בכל שעה במהלך הנסיעה. הראו שתוצאת החישוב אכן מקיימת את הדרישות של סעיף א.

פתרון (אחת מתוך כמה אפשרויות):

בשלב ראשון,

חבר א מחלק את העוגה לשלושה חלקים שווים בעיניו, והם (כולם בשווי 4):

• 7 - 6

• 11 - 7

• 15 - 11

חבר ב חושב שהפרוסה השלישית שווה הכי הרבה, ולכן הוא מקצץ אותה כך שתהיה שווה בעיניו לפרוסה השנייה. אז הוא מקצץ את 14-15 והפרוסות הנשארות הן:

• 7 - 6

• 11 - 7

• 14 - 11

חבר ג בוחר את הפרוסה העדיפה בעיניו, שהיא 11 - 7. חבר ב בוחר את הפרוסה העדיפה בעיניו מבין הנשארות, שהיא 11 - 14. חבר א מקבל את הפרוסה הנשארת, שהיא 6 - 7.

בשלב שני, צריך לחלק את הפרוסה הנשארת, שהיא 14 - 15. כיוון שבפרוסה זו לכל החברים יש העדפות אחידות, יש רק דרך אחת לחלק אותה – כל אחד יקבל שליש שעה (20 דקות).

לסיכום: חלוקת הזמן הסופית היא:

שעה:	7 - 6	8 - 7	9 - 8	10 - 9	11 - 10	12 - 11	13 - 12	14 - 13	15 - 14
מקדימה:	א	ג	ג	ג	ג	ב	ב	ב	כל אחד 20 דקות

ניתן לראות שהתכונות מתקיימות: אף אחד לא מקנא, וכל אחד יושב מקדימה שני פרקי-זמן רציפים לכל היותר.

שאלה 3 [30 נק']. בתיים בירושה

הקבלן שלומי חסון ז"ל נפטר והותיר לילדיו מספר דירות. הילדים פנו לשמאי שחילק ביניהם את הדירות. הם רוצים לבדוק האם החלוקה של השמאי הוגנת. עליכם לכתוב פונקציה שתעזור להם לברר.

א [10 נק']. כיתבו בפסאודו-קוד פונקציה המקבלת חלוקה של דירות לילדים, ובודקת האם החלוקה ללא קנאה. העזרו בהגדרות:

- בית מיוצג ע"י מספר שלם (מספר סידורי של הבית).

- ילד מיוצג ע"י המחלקה הבאה:

```
class Child {  
    set<int> houses;  
  
    // Keeps the set of houses that this child received.  
    map<int,float> map_house_to_value;  
  
    // Keeps, for each house, its value for this child.  
};
```

- הפונקציה מקבלת וקטור של ילדים, ומחזירה true אם-ורק-אם החלוקה של בתיים לילדים היא ללא קנאה.

פתרון: נשתמש בפונקציית-עזר המחשבת את הערך של ילד x לכל הבתים של ילד y :

```
float value_of_houses(Child x, Child y) {  
    float sum=0;  
    for (int h: y.houses)  
        sum += x.map_house_to_value[h];  
    return sum;  
}
```

```
bool is_envy_free(vector<Child> children) {  
    for (Child x: children) {  
        for (Child y: children) {  
            if (value_of_houses (x,y) > value_of_houses (x,x)) {
```

```

        return false;
    }
}
return true;
}

```

ב [10 נק']. כיתבו פונקציה המקבלת חלוקה של דירות לילדים, ובודקת האם החלוקה היא ללא-קנאה פרט לחפץ אחד (EF1).

פתרון: נשתמש בפונקציית-עזר המחשבת את הערך של ילד x לבית הטוב ביותר שיש לילד y :

```

float value_of_best_house(Child x, Child y) {
    float max=0;
    for (int h: y.houses)
        if (x.map_house_to_value[h] > max)
            max = x.map_house_to_value[h];
    return max;
}

```

```

bool is_EF1(vector<Child> children) {
    for (Child x: children) {
        for (Child y: children) {
            float other_value =
                value_of_houses (x,y) - value_of_best_house(x,y);
            if (other_value > value_of_houses (x,x)) {
                return false;
            }
        }
    }
}

```

```

}
return true;
}

```

ג [10 נק']. תנו שלוש דוגמאות עם 3 ילדים לפעולות הפונקציה שכתבתם:

- דוגמה שבה שתי הפונקציות מחזירות true;
- דוגמה שבה שתי הפונקציות מחזירות false;
- דוגמה שבה פונקציה אחת מחזירה true והשנייה מחזירה false.

עבור כל דוגמה, תארו את הקלט (וקטור הילדים, ערכי השדות בכל Child) ופרטו את שלבי החישוב.

פתרון: נניח שיש 3 בתים. הערכים שכל ילד מייחס לכל בית הם:

- ילד א: בית 1=1, בית 2=1, בית 3=0.
- ילד ב: בית 0=1, בית 1=2, בית 3=1.
- ילד ג: בית 1=1, בית 2=0, בית 3=1.

בחלוקה א-1 ב-2 ג-3 אין קנאה, ולכן שתי הפונקציות מחזירות true.

בחלוקה שבה א מקבל את כל הבתים ושאר הילדים לא מקבלים כלום, יש קנאה, כי הערכים של שאר הילדים הם 0, וגם אם נוריד בית אחד של ילד א, הערך שלו (בעיני האחרים) הוא עדיין 1. ולכן שתי הפונקציות מחזירות false.

בחלוקה א-3 ב-1 ג-2 יש קנאה, כי הערך של כל ילד הוא 0 והערך של הילדים האחרים בעיניו הוא 1. לכן is_envy_free מחזירה false. אבל אם נוריד בית אחד מהאחרים, הערך שלהם בעיניו יהיה 0. לכן is_EF1 מחזירה true.

שאלה 4 [20 נק']. שידוך מקסימלי בגרף דו-צדדי

נתון גרף דו-צדדי שבו לכל קשת יש משקל. המשקל של קשת בין צומת i לצומת j הוא: $w[i, j]$.

כדי למצוא שידוך עם משקל מקסימלי בגרף, השתמשו בתוכנית האופטימיזציה הבאה:

$$\text{Maximize } \sum_{i,j} w[i,j] * x[i,j]$$

$$\text{Such that For all } i: \sum_j x[i,j] = 1$$

$$\text{For all } j: \sum_i x[i,j] = 1$$

$$\text{For all } i,j: 1 \geq x[i,j] \geq 0$$

תרגמתם אותה לשפת מתמטיקה וקיבלתם את התוצאה הבאה (המטריצה x):

	$i=0$	$i=1$	$i=2$
$j=0$	0.1	0.45	0.45
$j=1$	0.2	0.3	0.5
$j=2$	0.7	0.25	0.05

א [10 נק']. תארו אלגוריתם כללי, המקבל מטריצה x עם ערכים לא-שלמים בגודל כלשהו (כמו המטריצה למעלה), ומחזיר מטריצה עם ערכים שלמים, המייצגת שידוך עם משקל מקסימלי בגרף. האלגוריתם מקבל כקלט את המטריצה x בלבד.

פתרון: אם כל הערכים במטריצה שלמים (0 או 1) – סיימנו.

אחרת, בוחרים תא במטריצה שיש בו ערך לא שלם. כיוון שסכום הערכים בכל שורה הוא 1, חייב להיות ערך נוסף באותה שורה שגם הוא לא שלם. כיוון שסכום הערכים בכל עמודה הוא 1, חייב להיות ערך נוסף באותה עמודה שגם הוא לא שלם. ממשיכים ללכת בטבלה מתא לא-שלם אחד לאחר, עד שחוזרים לתא שכבר היינו בו – מצאנו מעגל שכולו מספרים שבורים.

מעבירים משקל מתא לתא במעגל, כך שהסכום בכל שורה ועמודה נשאר 1, ולפחות אחד התאים הופך להיות שלם (0 או 1), ואף תא לא חורג מהתחום.

חוזרים להתחלה וממשיכים כך עד שכל הערכים במטריצה שלמים. במצב זה, בכל שורה יש "1" יחיד ובכל עמודה יש "1" יחיד כך שיש לנו שידוך.

ב [10 נק']. הדגימו את פעולת האלגוריתם על המטריצה x למעלה. פרטו את כל המטריצות המתקבלות בשלבי-הביניים של החישוב. תארו את השידוך המקסימלי המתקבל – איזה i משודך לאיזה j ?

פתרון:

נתחיל (שרירותית) מתא $0,0$, שבו יש מספר לא שלם. נעבור משם לתא $1,0$ ולתא $1,1$ ולתא $0,0$. בחזרה – סגרנו מעגל. נעביר במעגל את הערך הגדול ביותר שאפשר להעביר בלי לחרוג מהתחום, שהוא 0.1 . הטבלה החדשה:

	$i=0$	$i=1$	$i=2$
$j=0$	0	0.55	0.45
$j=1$	0.3	0.2	0.5
$j=2$	0.7	0.25	0.05

נעשה אותו הדבר על המעגל 1,0 2,0 2,1 1,1 - הפעם המשקל יהיה 0.2:

	$i=0$	$i=1$	$i=2$
$j=0$	0	0.75	0.25
$j=1$	0.3	0	0.7
$j=2$	0.7	0.25	0.05

נעשה אותו הדבר על המעגל 1,0 2,0 2,2 2,1 - הפעם המשקל יהיה 0.25:

	$i=0$	$i=1$	$i=2$
$j=0$	0	1	0
$j=1$	0.3	0	0.7
$j=2$	0.7	0	0.3

נשאר עוד מעגל אחד והמשקל הוא 0.3:

	$i=0$	$i=1$	$i=2$
$j=0$	0	1	0
$j=1$	0	0	1
$j=2$	1	0	0

השידוך המתקבל:

$j=0 \ i=1$
 $j=1 \ i=2$
 $j=2 \ i=0$

שאלה 5 [20 נק'] כמה עולה העוגה?

עמי קנה לתמי עוגה, ושילם בביטקוין. תמי רוצה לדעת כמה עלתה העוגה. היא מסתכלת בשרשרת-הבלוקים, באתר live.clockcypher.com, מזהה את הבלוק והעיסקה של קניית העוגה, ורואה שם את הנתונים הבאים:

Block height: 1569632

Tx ID: b6f5f7c0e7f9e2ed17eb601e9bbe3f5e1d5975b7ccfd929ca06

- Input 0: 0.3 BTC from 2NFzMXUaf7vCd77Sj8Ke8ZfVL1zWpKC
- Output 0: 0.11 BTC to 2NFzMXUaf7vCd77Sj8Ke8ZfVL1zWpKC
- Output 1: 0.18 BTC to 2MvCdYFmtpPF79SFXwHDMXuHURZ9Bfm

א [5 נק'] מהו מחיר העוגה?

ב [5 נק'] מה גובה עמלת-העיסקה?

ג [5 נק'] מי קיבל את עמלת-העיסקה?

ד [5 נק'] הבלוק הגבוה ביותר בשרשרת-הבלוקים הוא 1569638. לפני כמה זמן (בקירוב) נקנתה העוגה?

פתרון.

א. בעיסקה המדוברת, יש קלט אחד בגובה 0.3 מכתובת מסויימת, ויש פלט אחד בגובה 0.11 לאותה כתובת – זה כנראה העודף. לכן הסכום ששולם על העוגה הוא הפלט השני - **0.18 ביטקוין**.

ב. ההפרש בין הקלט לבין הפלטים הוא **0.01 ביטקוין**, ולכן זו עמלת העיסקה.

ג. עמלת-העיסקה ניתנת לכורה או מועדון-הכריה שהצליח "לכרות" את הבלוק. המפתח הציבורי שלו לא כתוב כאן.

ד. זמן-כרייה של כל בלוק הוא 10 דקות בקירוב. אם גובה הבלוק הנוכחי גבוה ב-6 מגובה הבלוק של העיסקה, אז העוגה נקנתה בקירוב לפני 60 דקות.

– בהצלחה!!