



אוניברסיטת אריאל בשומרון

פקולטה: מדעי הטבע

מחלקה: מדעי המחשב

שם הקורס: אלגוריתמים כלכליים

קוד הקורס: 2-7062310 קבוצה 4

סמסטר _____ ב _____ מועד _____ א _____ תאריך בחינה: 26/6/2019

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 שגויה או אינה מוגדרת עד הסוף (חסרות הנחות מסויימות), הניחו את ההנחות שנראות בעיניכם הגיוניות ביותר בהתאם לשאלה.

בהצלחה!!

שאלה 1 [30 נק']. מכירת אגרטלים

חברת "חפירות כלכליות" מצאה בחפירות ארכיאולוגיות בסין שלושה אגרטלים זהים מתקופת שושלת מינג. החברה העמידה את האגרטלים למכרז. שבעה משתתפים הגיעו למכרז: אבי, בתיה, גדי, דני, הניה, ורד, זילפה. כל אחד מהם מעוניין לקנות אגרטל אחד. לכולם יש העדפות קוואזי-ליניאריות. הערכים שהם מייחסים לאגרטל הם: אבי-10, בתיה-20, גדי-30, דני-40, הניה-50, ורד-60, זילפה-70. ערך האגרטלים עבור החברה הוא 5.

א [10 נק']. המכרז התנהל בשיטת ויקרי-קלארק-גרובס (VCG). מי היו הזוכים, כמה כל אחד שילם, וכמה הרויחה החברה? הסבירו בפירוט את אופן החישוב.

פתרון: בשיטת וק"ג, נבחרת התוצאה הממקסמת את סכום הערכים, ולכן הזוכות יהיו הניה, ורד וזילפה. התשלום של כל אחת הוא ה"נזק" שהיא גורמת לשאר המשתתפים. נחשב לגבי זילפה: אם היא לא משתתפת, הזוכים יהיו דני, הניה וורד, וסכום הערכים שלהם יהיה $40+50+60$. אם היא כן משתתפת, סכום הערכים של האחרים הוא רק $50+60$. התשלום שלה הוא ההפרש בין הסכומים, שהוא 40. אותו חישוב בדיוק נכון לגבי ורד והניה. בסה"כ החברה הרויחה 120.

ב [10 נק']. המכרז התנהל בשיטת מיירסון (Myerson) עם כלל-בחירה יעיל פארטו. מי היו הזוכים, כמה כל אחד שילם, וכמה הרויחה החברה? הסבירו בפירוט את אופן החישוב.

פתרון: כשההעדפות הן קוואזי-ליניאריות, הכלל היחיד שהוא יעיל-פארטו הוא מיקסום סכום הערכים. לכן, גם במקרה זה, הזוכות יהיו הניה, ורד וזילפה. התשלום של כל אחת הוא ערך-הסף – הערך שבו היא עוברת מהפסד לזכייה. במקרה זה, ערך הסף של כולן הוא ההכרזה של דני – 40. לכן כל אחת משלמת 40 והרווח של החברה הוא 120.

ג [10 נק']. המכרז התנהל בשיטת מיירסון (Myerson) עם כלל-בחירה "בחר את שלושת הערכים הנמוכים ביותר". מי היו הזוכים, כמה כל אחד שילם, וכמה הרויחה החברה? הסבירו בפירוט את אופן החישוב.

פתרון: מכרז מיירסון מוגדר רק כשכלל-הבחירה הוא מונוטוני-עולה. במקרה זה, כלל הבחירה לא מונוטוני: אבי בתיה וגדי זוכים, אבל אם גדי (למשל) יעלה את ההכרזה שלו ל-50, הוא כבר יפסיק לזכות. לכן אי-אפשר לבצע מכרז מיירסון עם כלל-בחירה זה.

לחלופין, אם בכל-זאת ננסה לבצע את מכרז מיירסון עם כלל זה, המכרז לא יהיה אמיתי – המשתתפים לא יגידו את הערכים האמיתיים שלהם, אלא יגידו את הערך הנמוך ביותר האפשרי (0) כדי לזכות במכרז. לא ברור מי בדיוק יזכה במצב זה, אבל הרווח של החברה יהיה 0.

שאלה 2 [30 נק']. חלוקת דואר

ברחוב "מכתבים כלכליים" יש בית-דואר אחד, הנמצא בקצה הרחוב.

ברחוב גרים n תושבים. התושב ה- k גר במרחק של $k/2$ דקות הליכה מבית-הדואר.

הדורך יוצא בכל בוקר מבית-הדואר, הולך עד הקצה השני של הרחוב ומחלק מכתבים, ואז חוזר בחזרה לבית-הדואר:



המשכורת של הדורך היא 12 ש"ח לדקה. ועד הרחוב החליט, שמשכורתו של הדורך תשולם ע"י התושבים. חלוקת התשלום בין התושבים צריכה לקיים את העקרונות הבאים:

1. התשלום של כל תושב תלוי רק בעלויות השוליות שהוא מוסיף כאשר הוא מצטרף לרחוב.
2. תושב שאינו מוסיף שום עלות שולית – לא משלם כלום.
3. אם מעסיקים כמה דוורים (עם משכורות שונות), מחשבים את התשלום של כל תושב לכל דורך בנפרד, ומסכמים את התשלומים – התוצאה זהה לתוצאה המתקבלת כשמחשבים את התשלום של כל תושב לכל הדוורים יחד.

א [10 נק']. ציינו שם של אלגוריתם המתאים להחלטת ועד הרחוב. ציינו שלוש תכונות של האלגוריתם, והתאימו אותן לשלושת העקרונות של ועד הרחוב.

- פתרון: ערך שאפלי. תכונות: 1. סימטריה, 2. "עציץ", 3. ליניאריות.

ב [10 נק']. הניחו שיש $n=3$ תושבים ברחוב. חשבו כמה ישלם כל תושב לפי האלגוריתם של סעיף א. פרטו את אופן החישוב. **שימו לב:** משכורת הדורך במקרה זה היא 36 (דקה וחצי הלך, דקה וחצי חזר, סה"כ 3 דקות, כפול 12 ש"ח לדקה).

- פתרון: נחשב את העלויות השוליות של כל אחד מהתושבים. תושב 1 מוסיף 12 ש"ח כשהוא לבד (2 סדרים), ולא מוסיף כלום כשהוא מצטרף לתושב 2 ו/או 3 (4 סדרים), ולכן הממוצע שלו $4 = 24/6$.
- תושב 2 מוסיף 24 ש"ח כשהוא לבד (2 סדרים), 12 ש"ח כשהוא מצטרף לתושב 2 בלבד (סדר 1), ולא מוסיף כלום כשהוא מצטרף לתושב 3 או 2+3 (שלושה סדרים), ולכן הממוצע שלו $10 = 60/6$.
- תושב 3 מוסיף 36 ש"ח כשהוא לבד (2 סדרים), 24 ש"ח כשהוא מצטרף לתושב 1 בלבד (סדר 1), ו-12 ש"ח כשהוא מצטרף לתושב 2 או 1+2 (שלושה סדרים), ולכן הממוצע שלו $22 = 132/6$.
- **בדיקה:** סכום התשלומים הכולל הוא $36 =$ משכורת הדורך.

ג [10 נק']. הניחו שיש $n=4$ תושבים ברחוב. חשבו כמה ישלם כל תושב לפי האלגוריתם של סעיף א.

דמ: יש שני פתרונות:

- פתרון אחד פשוט, אבל ארוך מאד;
 - פתרון שני קצר, כללי (מתאים לכל n), ומשתמש בשלושת התכונות שציינתם בסעיף א.
- פתרון ארוך: כמו בסעיף ב, רק שהפעם יש 24 פרמוטציות.
- פתרון קצר: נפריד את הבעיה לסכום של 4 בעיות שונות:
- בעיה 1 – רק הקטע הראשון (בין בית הדואר לתושב 1). תושבים 1,2,3,4 צריכים לשלם ביחד 12. לפי תכונת הסימטריה, כל אחד מהם משלם 3.
 - בעיה 2 – רק הקטע השני (בין תושב 1 לתושב 2). תושבים 2,3,4 צריכים לשלם ביחד 12. לפי תכונת העציץ, תושב 1 לא משלם כלום. לפי תכונת הסימטריה, כל אחד משאר התושבים משלם 4.
 - בעיה 3 – רק הקטע השלישי. תושבים 3,4 משלמים ביחד 12, לפי עציץ+סימטריה, כל אחד מהשניים משלם 6.
 - בעיה 4 – רק הקטע הרביעי, תושב 4 משלם לבד 12 (לפי תכונת העציץ, השאר לא משלמים).
- לפי תכונת הליניאריות, התשלומים בבעיה הכללית הם סכום התשלומים בכל אחת מהבעיות, ולכן הסכום הוא:
- תושב 1 משלם 3
 - תושב 2 משלם 7
 - תושב 3 משלם 13
 - תושב 4 משלם 25

שאלה 3 [20 נק']. חלוקת שאלות

שלושה סטודנטים פותרים ביחד מטלה עם שלוש שאלות. השאלות תלויות זו בזו – צריך קודם לפתור את שאלה א, אחר-כך את שאלה ב, ובסוף את שאלה ג.

כל שאלה עוסקת בנושא אחר ולכל סטודנט יש מומחיות אחרת, ולכן זמני הפתרון של השאלות משתנים: הזמן שלוקח לסטודנט i לפתור את שאלה j הוא $T[i, j]$ דקות.

הסטודנטים רוצים לחלק ביניהם את השאלות לפי העקרונות הבאים:

1. כל סטודנט פותר שאלה אחת.
2. כל שאלה נפתרת ע"י סטודנט אחד.
3. הסטודנטים מסיימים את כל המטלה מהר ככל האפשר.

א [10 נק']. כיתבו פקודה בשפת Mathematica (או שפה דומה) שתעזור לסטודנטים להחליט מי יפתור איזו שאלה. הסבירו בפירוט את אופן פעולת הפקודה שכתבתם, ומדוע היא עונה לדרישות.

פתרון: נגדיר, לכל זוג i, j , משתנה בשם $x[i, j]$ שערכו יהיה 1 אם סטודנט i פותר את מטלה j ו-0 אחרת.

```
FindMinimum[ x[1,1]*T[1,1] + x[1,2]*T[1,2] + x[1,3]*T[1,3]
+ x[2,1]*T[2,1] + x[2,2]*T[2,2] + x[2,3]*T[2,3]
+ x[3,1]*T[3,1] + x[3,2]*T[3,2] + x[3,3]*T[3,3] ,
x[1,1] + x[1,2] + x[1,3] == 1,
x[2,1] + x[2,2] + x[2,3] == 1,
x[3,1] + x[3,2] + x[3,3] == 1,
x[1,1] + x[2,1] + x[3,1] == 1,
x[1,2] + x[2,2] + x[3,2] == 1,
x[1,3] + x[2,3] + x[3,3] == 1,
0 <== x[1,1] <== 1, ..., 0 <== x[3,3] <== 1 ]
```

הסבר: שלוש השורות הראשונות הן הזמן הכולל שייקח לשלושת הסטודנטים לפתור את 3 השאלות. אנחנו רוצים שהזמן הזה יהיה קצר ביותר ולכן `FindMinimum`.

שלוש השורות הבאות מבטיחות שכל סטודנט i פותר בדיוק שאלה אחת.

שלוש השורות הבאות מבטיחות שכל שאלה j נפתרת ע"י סטודנט אחד בדיוק.

השורה האחרונה מבטיחה שכל משתנה הוא רק 0 (לא פותר) או 1 (פותר).

ב [10 נק']. הדגימו את פעולת הפקודה שכתבתם: כיתבו מטריצה T , כיתבו את הפלט של הפקודה, והסבירו איך מבינים מתוך הפלט של Mathematica, איזה סטודנט יפתור איזו שאלה.
פתרון:

$$T = \begin{bmatrix} 1, 2, 2; \\ 2, 1, 2; \\ 2, 2, 1 \end{bmatrix}$$

הפלט המתקבל הוא:

$$x = \begin{bmatrix} 1, 0, 0; \\ 0, 1, 0; \\ 0, 0, 1 \end{bmatrix}$$

כי במצב זה, הסכום הוא 3, סכום כל שורה הוא 1, וסכום כל עמודה הוא 1. בכל השמה אחרת, הסכום קטן יותר.
משמעות הפלט: סטודנט 1 פותר שאלה 1, סטודנט 2 פותר שאלה 2, סטודנט 3 פותר שאלה 3.

שאלה 4 [20 נק']. יעילות פארטו

הגדרות:

- מצב א נקרא **שיפור פארטו** חזק של מצב ב, אם מצב א טוב יותר לכל המשתתפים.
- מצב נקרא **יעיל פארטו** חלש אם לא קיים מצב אחר שהוא שיפור-פארטו-חזק שלו.

א [10 נק']. תנו דוגמה עם שלושה משתתפים ושלושה מצבים, שבה מצב מסויים הוא יעיל-פארטו-חלש אבל לא יעיל-פארטו. הוכיחו את תשובתכם (הוכיחו שהמצב יעיל-פארטו-חלש, והוכיחו שהמצב לא יעיל-פארטו).

פתרון: נתונה טבלה המפרטת את הערך של כל אדם לכל מצב:

	מצב א	מצב ב	מצב ג
אדם א	3	3	3
אדם ב	3	2	1
אדם ג	3	2	1

מצב ב הוא יעיל-פארטו-חלש. הוכחה: לא קיים מצב אחר שהוא טוב יותר לכל המשתתפים (מצב א טוב יותר לאנשים ב, ג אבל לא לאדם א).

מצב ב הוא לא יעיל-פארטו. הוכחה: קיים מצב אחר (מצב א) שהוא טוב יותר לחלק מהמשתתפים (ב,ג) וטוב באותה מידה לשאר המשתתפים (א).

ב [10 נק']. כיתבו פונקציה, בשפת-תיכנות לבחירתכם או בפסאודו-קוד, המקבלת מערך של שחקנים, מספר המציין מצב, ומערך המכיל את כל המצבים האפשריים, ובודקת האם המצב הנתון הוא יעיל-פארטו-חלש. היעזרו במחלקה הבאה:

```
class Person {
    double value(int state);
    // INPUT:    the index of a state.
    // OUTPUT:   the value of the state to the person.
}
```

פתרון:

```
bool isWeakParetoOptimal(Person[] people, int state, int[] allStates) {
    for (int other_state: allStates) {
        bool is_other_strong_improvement_of_state = true;
        for (Person p: people) {
            if (p.value(other_state) <= p.value(state)) {
                is_other_strong_improvement_of_state = false;
            }
        }
        if (is_other_strong_improvement_of_state) {
            return false; // state is not weak-Pareto-optimal
            // since other_state is a strong-Pareto improvement of it.
        }
    }
    return true; // state is not weak-Pareto-optimal
}
```


שאלה 5 [20 נק'] . זוג-או-פרד

במשחק "זוג או פרד", כל שחקן בוחר מספר שלם. אם סכום המספרים הוא זוגי, השחקן הראשון מנצח; אם הסכום איזוגי, השחקן השני מנצח.

א [10 נק'] . כיתבו חוזה בשפת solidity שיאפשר לשחקנים לשחק על כסף. הדרישות:

- כדי להשתתף במשחק, צריך לשלם לחוזה 100 פני לוחות (אפשר לשלם יותר).
- כל זוג שחקנים עוקבים משחקים אחד (השחקן הראשון משחק מול השני, השלישי מול הרביעי וכו').
- הזוכה במשחק (בכל זוג בנפרד) מקבל 180 פני.

תזכורת לשפת solidity:

- הטיפוס address מציין כתובת אתריום.
- לטיפוס address יש שיטה בשם transfer המשמשת לשליחת כסף לכתובת זו.
- כל שיטה מקבלת כקלט, מלבד הקלט הרגיל שלה, אובייקט מסוג msg עם שני שדות עיקריים:
 - msg.sender – הכתובת של שולח ההודעה הנוכחית (שדה מסוג address);
 - msg.value – סכום הכסף ששולח-ההודעה צירף להודעה.
- הפונקציה require משמשת לבדיקת תנאים בזמן ריצה (כמו assert בשפת סי). עליכם להשלים את השלד הבא – לממש את הבנאי ואת play. הוסיפו שדות לפי הצורך.

```
contract ZogOPered {
    constructor() public {
        ...
    }

    function play(int choice) public payable {
        ...
    }
}
```

אם אינכם זוכרים את התחביר של שפת solidity, הניחו שהתחביר זהה לשפת Java או C++ לפי בחירתכם. כמו כן, אתם יכולים להשתמש בכל הפונקציות הנמצאות בספריות התקניות של השפות הללו. הסבירו היטב בעברית מה אתם עושים.

פתרון. לצורך הבחינה הסתפקתי בפתרון פשוט, שאינו מתייחס לכל דרישות התחביר של solidity אלא רק לאלגוריתם.

```
contract ZogOPered {
    address player1, player2;
    int choicel, choice2;
    constructor() public {
        player1 = player2 = null;
    }

    function play(int choice) public payable {
        require (msg.value >= 100 finney);
        if (player1=null) { // first player comes - register choice
            player1 = msg.sender;
            choicel = choice;
        } else { // second player comes - calculate winner
            player2 = msg.sender;
            choice2 = choice;
            if (choicel+choice2 % 2 == 0) { // even sum - player1 wins
                player1.transfer(180 finney);
            } else { // odd sum - player2 winsd
                player2.transfer(180 finney);
            }
            // zero the fields for the next player:
            player1 = player2 = null;
        }
    }
}
```

ב [10 נק']. הדגימו בפירוט את הפונקציה שכתבתם בסעיף א על מהלך-האירועים הבא:

1. שחקן עם כתובת 1212 קורא ל-play(11) עם 100 פיני.
2. שחקן עם כתובת 3434 קורא ל-play(33) עם 100 פיני.
3. שחקן עם כתובת 5656 קורא ל-play(66) עם 100 פיני.
4. שחקן עם כתובת 7878 קורא ל-play(88) עם 50 פיני.
5. שחקן עם כתובת 9090 קורא ל-play(99) עם 150 פיני.

לגבי כל אחד מ-5 השלבים, ציינו מה בדיוק קורה בכל שלב:

- כמה כסף עובר במערכת, מאיזו כתובת ולאילו כתובות;
- איך משתנים ערכי השדות הפנימיים של החוזה.
- מהי היתרה של החוזה (כמה כסף יש בחוזה) אחרי השלב.

פתרון.

1. 100 פיני עוברים מכתובת 1212 לכתובת החוזה. Player1=1212, choicel=11. יתרת החוזה 100.

2. 100 פיני עוברים מכתובת 3434 לכתובת החוזה. `Player2=3434, choice2=33`. כיוון שהסכום זוגי, שחקן 1 מנצח ולכן החוזה שולח 180 פיני לכתובת 1212. יתרת החוזה 20.
3. 100 פיני עוברים מכתובת 5656 לכתובת החוזה. `Player1=5656, choice1=66`. יתרת החוזה 120.
4. הסכום נמוך מ-100 פיני ולכן ה-`require` עוצר את הביצוע – אין שינוי במשתנים הפנימיים. יתרת החוזה 120.
5. 150 פיני עוברים מכתובת 9090 לכתובת החוזה. `Player2=9090, choice2=99`. כיוון שהסכום איזוגי, שחקן 2 מנצח, ולכן החוזה שולח 180 פיני לכתובת 9090. יתרת החוזה 90.