

מדעי המחשב

על פי תכנית הרפורמה ללמידה משמעותית

הוראות לנבחן

א. משך הבחינה: שלוש שעות.

ב. מבנה השאלון ומפתח ההערכה: בשאלון זה שלושה פרקים.

פרק ראשון – בפרק זה שלוש שאלות, ענה על פי

ההוראות בפרק. — $(10 \times 1) + (15 \times 1)$ — 25 נקודות

פרק שני – בפרק זה שלוש שאלות,

ומהן עליך לענות על שתיים. — (25×2) — 50 נקודות

פרק שלישי – בפרק זה שאלות בארבעה מסלולים שונים.

ענה על שאלה אחת במסלול שלמדת. — (25×1) — 25 נקודות

סה"כ — 100 נקודות

ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ ממחשב הניתן לתכנות.

ד. הוראות מיוחדות:

(1) את כל התכניות שאתה נדרש לכתוב בשפת מחשב בפרקים הראשון והשני

כתוב בשפה אחת בלבד — Java או C#.

(2) רשום על הכריכה החיצונית של המחברת באיזו שפה אתה כותב — Java או C#.

(3) רשום על הכריכה החיצונית של המחברת את שם המסלול שלמדת.

המסלול הוא אחד מארבעת המסלולים האלה:

מערכות מחשב ואסמבלי, מבוא לחקר ביצועים, מודלים חישוביים, תכנות מונחה עצמים.

הערה: בתכניות שאתה כותב לא יורדו לך נקודות, אם תכתוב אות גדולה במקום

אות קטנה או להפך.

כתוב במחברת הבחינה בלבד, בעמודים נפרדים, כל מה שברצונך לכתוב ב**פניוטה** (ראשי פרקים, חישובים וכדומה).

רשום "טייטה" בראש כל עמוד טייטה. רישום טייטות כלשהן על דפים שמחוץ למחברת הבחינה עלול לגרום לפסילת הבחינה!

ההנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות ולנבחנים כאחד.

בהצלחה!

/המשך מעבר לדף/

השאלות

בשאלון זה שלושה פרקים.

עליך לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

פרק ראשון (25 נקודות)

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפותרים ב־ Java : בכל שאלה שנדרשת בה קליטה, הנח שבתכנית כתובה ההוראה:

```
Scanner input = new Scanner (System.in);
```

ענה על שאלה 1 — חובה (10 נקודות)

1. נתונה מחלקה תלמיד — **Student** שיש לה שתי תכונות: שם התלמיד — name מטיפוס מחרוזת, ומערך חד־ממדי arrTest בגודל 3 מטיפוס שלם. בכל תא במערך מאוחסן ציון של התלמיד במבחן.
כתוב ב־ Java או ב־ C# פעולה במחלקה **Student** שתחזיר את הציון הממוצע של התלמיד בשלושת המבחנים.
הערה: אין צורך לבדוק את תקינות הנתונים במערך.

ענה על אחת מהשאלות 2 - 3 (15 נקודות)

2. נתונה מחלקה שחקן — **Actor** שיש לה שלוש תכונות: מספר תעודת זהות — `id` מטיפוס מחרוזת, מין השחקן — `gender` מטיפוס מחרוזת ("F" מייצג נקבה, "M" מייצג זכר), מספר הסרטים שהשחקן השתתף בהם — `numFilms` מטיפוס שלם. לפניך ממשק המחלקה **Actor** הכתוב ב־Java וב־C#.

כותרת הפעולה ב־Java	תיאור הפעולה
<code>public Actor(String id, String gender, int numFilms)</code>	פעולה הבונה שחקן שמספר תעודת הזהות שלו <code>id</code> , מינו <code>gender</code> , מספר הסרטים שהשתתף בהם <code>numFilms</code> .
<code>public void addFilm()</code>	פעולה המוסיפה 1 למספר הסרטים שהשחקן השתתף בהם.
<code>public int compare(Actor other)</code>	פעולה שמחזירה: 1 — אם מספר הסרטים שהשחקן הנוכחי השתתף בהם גדול ממספר הסרטים שהשחקן <code>other</code> השתתף בהם. 2 — אם מספר הסרטים שהשחקן הנוכחי השתתף בהם קטן ממספר הסרטים שהשחקן <code>other</code> השתתף בהם. 3 — אם מספר הסרטים שהשחקן הנוכחי השתתף בהם זהה למספר הסרטים שהשחקן <code>other</code> השתתף בהם.

(שים לב: המשך השאלה בעמוד הבא.)

תיאור הפעולה	כותרת הפעולה ב-C#
פעולה הבונה שחקן שמספר תעודת הזהות שלו id, מינו gender, מספר הסרטים שהשתתף בהם numFilms.	public Actor(string id, string gender, int numFilms)
פעולה המוסיפה 1 למספר הסרטים שהשחקן השתתף בהם.	public void AddFilm()
פעולה שמחזירה: 1 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם גדול ממספר הסרטים שהשחקן other השתתף בהם. 2 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם קטן ממספר הסרטים שהשחקן other השתתף בהם. 3 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם זהה למספר הסרטים שהשחקן other השתתף בהם.	public int Compare(Actor other)

הנח שלכל תכונה הוגדרו ב־ Java פעולות get ו־ set וב־ C# פעולות Get ו־ Set.

א. ממש את הפעולה הבונה של המחלקה **Actor**.

ב־ Java את הפעולה הבונה:

```
public Actor(String id, String gender, int numFilms)
```

ב־ C# את הפעולה הבונה:

```
public Actor(string id, string gender, int numFilms)
```

ב. ממש ב־ Java את הפעולה compare או ב־ C# את הפעולה Compare.

ג. בפעולה הראשית במחלקה Program הוגדר מערך חד־ממדי actArr בגודל 37

מטיפוס **Actor**.

כתוב ב־ Java או ב־ C# פעולה במחלקה Program שתקבל מערך חד־ממדי מטיפוס

Actor ומספר num שלם וגדול מ־ 0.

הפעולה תדפיס את מספר השחקנים במערך שהשתתפו ב**יותר** מ־ num סרטים.

3. נתונה מחלקה תכנית טלוויזיה – **TvProgram** שיש לה שלוש תכונות:
קוד התכנית – code מטיפוס שלם, היום בשבוע שבו משודרת התכנית – day מטיפוס שלם, המקבל ערכים מ-1 עד 7 (כולל), האם זו תכנית ספורט – isSport מטיפוס בוליאני, שערכה true אם זו תכנית ספורט, אחרת – false.
הנח שלכל תכונה הוגדרו ב- Java פעולות get ו- set וב- C# פעולות Get ו- Set.
- א. כתוב ב- Java או ב- C# פעולה בונה במחלקה **TvProgram** שתקבל ערכים לכל תכונה.
- ב. נתונה מחלקה תכניות השבוע – **TvWeek** שיש לה שתי תכונות:
מערך חד-ממדי – arrProg בגודל 100 מטיפוס **TvProgram**, מספר התכניות הנוכחי בשבוע – current מטיפוס שלם, שערכו קטן מגודל המערך.
הנח שלכל תכונה הוגדרו ב- Java פעולות get ו- set וב- C# פעולות Get ו- Set.
- (1) כתוב ב- Java או ב- C# את כותרת המחלקה **TvWeek** ואת התכונות שלה.
- (2) כתוב ב- Java או ב- C# במחלקה **TvWeek** פעולה שתקבל תכנית **TvProgram**, תוסיף אותה לתכניות השבוע ותעדכן את מספר התכניות הנוכחי בשבוע.
הנח שיש מקום להוסיף את התכנית.
- (3) כתוב ב- Java או ב- C#, במחלקה **TvWeek** פעולה שתחזיר את מספר תכניות הספורט שיש בשבוע בטלוויזיה.

פרק שני (50 נקודות)

שים לב: בכל שאלה שנדרש בה מימוש אתה יכול להשתמש בפעולות של המחלקות תור, מחסנית, עץ בינרי וחוליה, בלי לממש אותן. אם אתה משתמש בפעולות נוספות, עליך לממש אותן.

ענה על שתיים מהשאלות 4-6 (לכל שאלה – 25 נקודות).

4. לפניך המחלקה טבעת – **Ring** שלה שתי תכונות:

גודל הטבעת מטיפוס מחרוזת ("S" – טבעת קטנה, "L" – טבעת גדולה);

מספר שלם המייצג את צבע הטבעת.

Java

```
public class Ring
{
    private String size; //גודל הטבעת
    private int color; //צבע הטבעת
    public Ring()
    {
        this.size = "L";
        this.color = 0;
    }
    public Ring(String str, int c)
    {
        this.size = str;
        this.color = c;
    }
    public String getSize()
    {
        return this.size;
    }
    public int getColor()
    {
        return this.color;
    }
}
```

C#

```
public class Ring
{
    private string size; //גודל הטבעת
    private int color; //צבע הטבעת
    public Ring()
    {
        this.size = "L";
        this.color = 0;
    }
    public Ring(string str, int c)
    {
        this.size = str;
        this.color = c;
    }
    public string GetSize()
    {
        return this.size;
    }
    public int GetColor()
    {
        return this.color;
    }
}
```

לפניך ממשק המחלקה מוט – **Pole**.

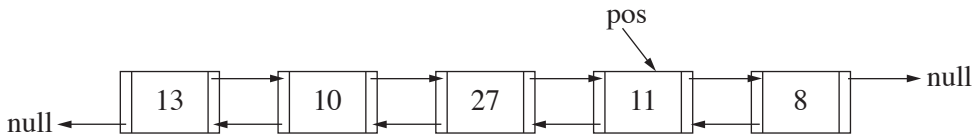
Java	C#	הפעולה
public Pole()	public Pole()	פעולה הבונה מוט ריק. סיבוכיות זמן הריצה של הפעולה היא $O(1)$.
public void add(Ring r)	public void Add(Ring r)	פעולה המכניסה טבעת r לראש המוט. סיבוכיות זמן הריצה של הפעולה היא $O(1)$.
public Ring remove()	public Ring Remove()	פעולה המחזירה את הטבעת שבראש המוט. בנוסף, הפעולה מוציאה את הטבעת מראש המוט. סיבוכיות זמן הריצה של הפעולה היא $O(1)$.
public boolean isEmpty()	public bool IsEmpty()	אם המוט ריק הפעולה מחזירה $true$, אחרת – היא מחזירה $false$. סיבוכיות זמן הריצה של הפעולה היא $O(1)$.
public void sort()	public void Sort()	פעולה המסדרת את הטבעות שעל המוט כך: הטבעות הגדולות "מונחות" בתחתית המוט והטבעות הקטנות מעליהן.

- א. ממש את הפעולה `sort()` ב-Java או `Sort()` ב-C#, שבמחלקה **Pole**.
אתה יכול להשתמש בפעולות הנוספות של המחלקה **Pole** בלי לממש אותן.
בתשובתך השתמש רק בפעולות המחלקות **Pole** ו-**Ring**.
ב. מהי סיבוכיות זמן הריצה של הפעולה שמימשת בסעיף א? נמק את תשובתך.

שים לב: לשאלה זו שני נוסחים:
 נוסח אחד ב-Java בעמוד 8-9, ונוסח אחר ב-C# בעמ' 10-11.
 עבוד על פי השפה שלמדת.

לפתורים ב-Java

נגדיר **רשימה דו־כיוונית** כאוסף סדור של חוליות מטיפוס `BinNode<Integer>` המקושרות כך:
 לכל זוג חוליות `p1`, `p2` ברשימה, אם מתקיים `p1.getRight() == p2`, אז מתקיים גם
`p2.getLeft() == p1`. **ברשימה דו־כיוונית** יש לפחות שתי חוליות.
 כלומר: כל חוליה ברשימה – חוץ מהחוליה שבקצה הימני של הרשימה והחוליה שבקצה השמאלי
 של הרשימה – מצביעה על החוליה שלפניה ועל החוליה שאחריה.
 לפניך דוגמה ל**רשימה דו־כיוונית** ומשתנה `pos` מטיפוס `BinNode<Integer>` המצביע על
 חוליה כלשהי ברשימה דו־כיוונית.



הפעולה `firstLeft` מקבלת מצביע `pos` שונה מ-`null` מטיפוס `BinNode<Integer>` המצביע
 על חוליה כלשהי ברשימה דו־כיוונית ומחזירה את החוליה השמאלית ביותר ברשימה.
 הפעולה `firstRight` מקבלת מצביע `pos` שונה מ-`null` מטיפוס `BinNode<Integer>` המצביע
 על חוליה כלשהי ברשימה דו־כיוונית ומחזירה את החוליה הימנית ביותר ברשימה.
א. לפניך שלד של הפעולה `firstLeft`.

העתק אותו למחברתך והשלם אותו, כך שהפעולה תבצע את הנדרש.

```
public static BinNode<Integer> firstLeft(BinNode<Integer> pos)
{
    while ( _____ )
        pos = _____ ;
    return _____ ;
}
```

/המשך בעמוד 9/

- ב. לפניך הפעולה `what(BinNode<Integer> pos)` המקבלת מצביע לחוליה כלשהי ברשימה דו־כיוונית ומחזירה ערך בוליאני. הרשימה הדו־כיוונית מכילה לפחות 3 חוליות. (1) עקוב אחר ביצוע הפעולה בעבור המשתנה `pos` והרשימה שבדוגמה המוצגת בתחילת השאלה.
- במעקב הראה את הרשימה הדו־כיוונית ואת ערכי המשתנים `pos, left, right, sum`.

```
public static boolean what(BinNode<Integer> pos)
{
    BinNode<Integer> left = firstLeft(pos);
    BinNode<Integer> right = firstRight(pos);

    int sum = left.getValue() + right.getValue();
    left = left.getRight();
    right = right.getLeft();

    while ((left != right) && (left.getRight() != right) &&
           (left.getValue() + right.getValue() == sum))
    {
        left = left.getRight();
        right = right.getLeft();
    }
    if (left == right)
        return right.getValue() == sum;

    if (left.getRight() == right)
        return left.getValue() + right.getValue() == sum;
    return false;
}
```

- (2) קבע אם אפשר או אי אפשר להחליף את 3 השורות האחרונות שבפעולה – השורות המוקפות במסגרת – בהוראה:

```
return left.getValue() + right.getValue() == sum;
```

נמק את קביעתך.

/המשך בעמוד 10/

לפתורים ב- C#

נגדיר **רשימה דו־כיוונית** כאוסף סדור של חוליות מטיפוס `BinNode<int>` המקושרות כך:

לכל זוג חוליות `p1`, `p2` ברשימה, אם מתקיים `p1.GetRight() == p2`, אז מתקיים גם

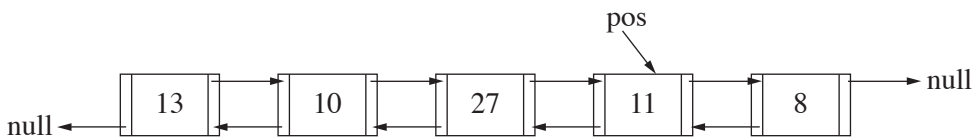
`p2.GetLeft() == p1`. ברשימה דו־כיוונית יש לפחות שתי חוליות.

כלומר: כל חוליה ברשימה – חוץ מהחוליה שבקצה הימני של הרשימה והחוליה שבקצה השמאלי

של הרשימה – מצביעה על החוליה שלפניה ועל החוליה שאחריה.

לפניך דוגמה ל**רשימה דו־כיוונית** ומשתנה `pos` מטיפוס `BinNode<int>` המצביע על חוליה

כלשהי ברשימה דו־כיוונית.



הפעולה `FirstLeft` מקבלת מצביע `pos` שונה מ־`null` מטיפוס `BinNode<int>` המצביע על

חוליה כלשהי ברשימה דו־כיוונית ומחזירה את החוליה השמאלית ביותר ברשימה.

הפעולה `FirstRight` מקבלת מצביע `pos` שונה מ־`null` מטיפוס `BinNode<int>` המצביע על

חוליה כלשהי ברשימה דו־כיוונית ומחזירה את החוליה הימנית ביותר ברשימה.

א. לפניך שלד של הפעולה `FirstLeft`.

העתק אותו למחברתך והשלם אותו, כך שהפעולה תבצע את הנדרש.

```
public static BinNode<int> FirstLeft(BinNode<int> pos)
```

```
{
    while ( _____ )
        pos = _____ ;
    return _____ ;
}
```

/המשך בעמוד 11/

ב.

לפניך הפעולה `What(BinNode<int> pos)` המקבלת מצביע לחוליה כלשהי ברשימה דו־כיוונית ומחזירה ערך בוליאני. הרשימה הדו־כיוונית מכילה לפחות 3 חוליות.

(1) עקוב אחר ביצוע הפעולה בעבור המשתנה `pos` והרשימה שבדוגמה המוצגת בתחילת השאלה.

במעקב הראה את הרשימה הדו־כיוונית ואת ערכי המשתנים `pos`, `left`, `right`, `sum`.

```
public static bool What(BinNode<int> pos)
{
    BinNode<int> left = FirstLeft(pos);
    BinNode<int> right = FirstRight(pos);

    int sum = left.GetValue() + right.GetValue();
    left = left.GetRight();
    right = right.GetLeft();

    while ((left != right) && (left.GetRight() != right) &&
           (left.GetValue() + right.GetValue() == sum))
    {
        left = left.GetRight();
        right = right.GetLeft();
    }
    if (left == right)
        return right.GetValue() == sum;

    if (left.GetRight() == right)
        return left.GetValue() + right.GetValue() == sum;
    return false;
}
```

(2) קבע אם אפשר או אי אפשר להחליף את 3 השורות האחרונות שבפעולה — השורות המוקפות במסגרת — בהוראה:

```
return left.GetValue() + right.GetValue() == sum;
```

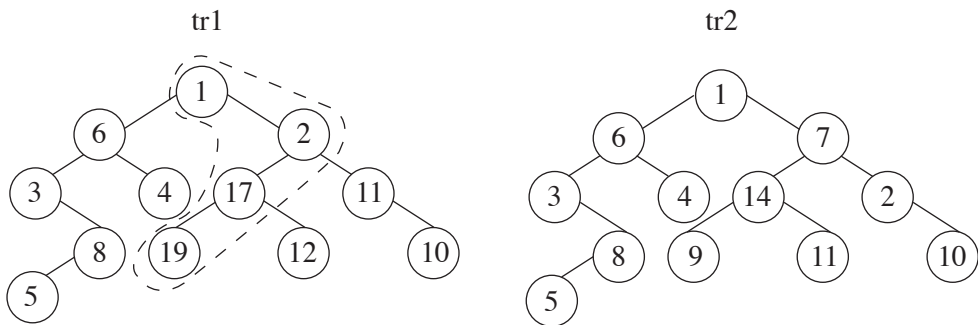
נמק את קביעתך.

/המשך בעמוד 12/

6. **עץ מספרים** הוא עץ בינארי לא ריק מטיפוס שלם, שהערכים בצמתים שלו הם מספרים שלמים וגדולים מ-0 השונים זה מזה.

על **עץ מספרים** מוגדרת פעולה "מסלול-עולה", המחזירה true אם יש בעץ מסלול המתחיל בשורש העץ ומסתיים באחד העלים שלו, וערכי הצמתים ממוינים בסדר עולה מהשורש לעלה. אם אין מסלול כזה – הפעולה מחזירה false. לדוגמה:

בעבור **עץ מספרים** tr1 הפעולה "מסלול-עולה" מחזירה true. המסלול מוקף בקו שבור.
 בעבור **עץ מספרים** tr2 הפעולה "מסלול-עולה" מחזירה false.



ממש ב-Java או ב-C# את הפעולה "מסלול-עולה" בעבור **עץ מספרים** tr.
 כותרת הפעולה ב-Java: `public static boolean upPath(BinNode<Integer> tr)`
 כותרת הפעולה ב-C#: `public static bool UpPath(BinNode<int> tr)`

פרק שלישי (25 נקודות)

בפרק זה שאלות בארבעה מסלולים:
 מערכות מחשב ואסמבלי, עמודים 13-16.
 מבוא לחקר ביצועים, עמודים 17-20.
 מודלים חישוביים, עמוד 21.
 תכנות מונחה עצמים ב־ Java, עמודים 22-26; תכנות מונחה עצמים ב־ C#, עמודים 27-31.
ענה על שאלה אחת במסלול שלמדת.

מערכות מחשב ואסמבלי

אם למדת מסלול זה, ענה על אחת מהשאלות 7-8 (25 נקודות).

7. בשאלה זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניהם.
 א. כתוב באסמבלי קטע תכנית להצפנת תו.
 התו מוגדר כך במקטע הנתונים:

TAV DB ?

ההצפנה נעשית כך:

סופרים את מספר הסיביות הדלוקות בתו.

אם המספר זוגי — מתבצעת הזזה מעגלית ימינה של הסיביות בתו, מספר פעמים השווה למספר הסיביות הדלוקות.

אם המספר אי־זוגי — מתבצעת הזזה מעגלית שמאלה של הסיביות בתו, מספר פעמים השווה למספר הסיביות הדלוקות.

ב. (אין קשר לסעיף א.)

לכל אחד מההיגדים (1)-(6) שלפניך, קבע אם הוא נכון או אינו נכון. אם ההיגד אינו נכון, הסבר מדוע.

(1) לפניך קטע תכנית באסמבלי.

MOV AX , 8

MOV BX , 2

DIV BX

בסיום קטע התכנית האוגר AX יכיל בהכרח 4.

(שים לב: המשך הסעיף בעמוד הבא.)

(2) לפניך קטע תכנית באסמבלי.

```
MOV AL , 56
ADD AL , 200
JZ STOP
INC AL
```

STOP:

בסיום קטע התכנית האוגר AX יכיל 1.

(3) לפניך קטע תכנית באסמבלי.

```
ARRAY DW 1, 2, 3, 4
MOV BX , ARRAY
ADD BX , 2
MOV AX , [BX]
```

בסיום קטע התכנית האוגר AX יכיל 3.

(4) לפניך קטע תכנית באסמבלי.

```
MOV CX , 3
MOV AX , 1
```

DO:

```
SHL AX , 1
LOOP DO
```

בסיום קטע התכנית האוגר AX יכיל 8.

(5) לפניך קטע תכנית באסמבלי.

```
MOV AX , 11000001B
MOV BX , 01000001B
```

המספר הנמצא ב־AX בהכרח גדול מהמספר הנמצא ב־BX.

(6) לפניך ההוראה:

```
OR AL , 3
```

לאחר ביצוע ההוראה ערכו של AL תמיד אי־זוגי.

/המשך בעמוד 15/

8. בשאלה זו שני סעיפים, א-ב, שאין קשר ביניהם, ענה על שניהם.

א. במקטע נתונים הוגדרו הנתונים כך:

```
VEC1 DB 45H , 26H , 32H , 82H
```

```
VEC2 DB 4 DUP(0)
```

לפניך קטע תכנית באסמבלי.

```
START: MOV CL , 4
        XOR CH , CH
        XOR SI , SI
NEXT:  PUSH CX
        MOV AL , VEC1[SI]
        MOV AH , AL
        AND AL , 0FH
        MOV CL , 4
        SHR AH , CL
        MUL AH      ;(*)
        MOV VEC2[SI] , AL
        INC SI
        POP CX
        LOOP NEXT
SOF:   NOP
```

(1) עקוב בעזרת טבלת מעקב אחר הביצוע של קטע התכנית וסרטט את VEC2 לאחר הביצוע.

בטבלת המעקב יש לכלול עמודה לכל אחד מהאוגרים AL , AH , CL , CH , SI .
צייר את המחסנית בכל שלב.

(2) מה מבצע קטע התכנית?

(3) בקטע התכנית, במקום ההוראה המסומנת ב- (*) נכתבה ההוראה: IMUL AH .

האם ביצוע קטע התכנית ישתנה? נמק את תשובתך.

(שים לב: סעיף ב של השאלה בעמוד הבא.)

ב. באוגר AL מאוחסן המספר 2 ובאוגר BL מאוחסן המספר 5.
 יש לאחסן באוגר DX את סכום המספרים מ- AL עד BL (כולל).
 לפניך שלושה קטעים i-iii באסמבלי. עקוב בעזרת טבלת מעקב אחר הביצוע של
כל אחד מהקטעים, וקבע אם הוא מבצע את הנדרש או אינו מבצע את הנדרש.

```

i          MOV    DX , 0
           MOV    AH , 0
           MOV    CL , AL
           SUB    CL , BL
AGAIN:    ADD    DX , AX
           INC    AL
           LOOPE AGAIN
           NOP

ii         XOR    DX , DX
           MOV    BH , 0
AGAIN:    ADD    DX , BX
           DEC    BL
           CMP    BL , AL
           JGE    AGAIN
           NOP

iii        XOR    DX , DX
           XOR    AX , AX
           MOV    BX , 0
AGAIN:    ADD    DX , AX
           ADD    AX , 1
           CMP    AX , BX
           JL     AGAIN
           NOP
    
```


מבוא לחקר ביצועים

אם למדת מסלול זה, ענה על אחת מהשאלות 9-10 (25 נקודות).

9. נתונה בעיית תכנון לינארי:

$$\max \{z = (2 + 2k)x_1 + 2x_2\}$$

בכפוף לאילוצים האלה:

$$(1) \quad 2x_1 + x_2 \leq 10$$

$$(2) \quad x_1 + x_2 \leq 6$$

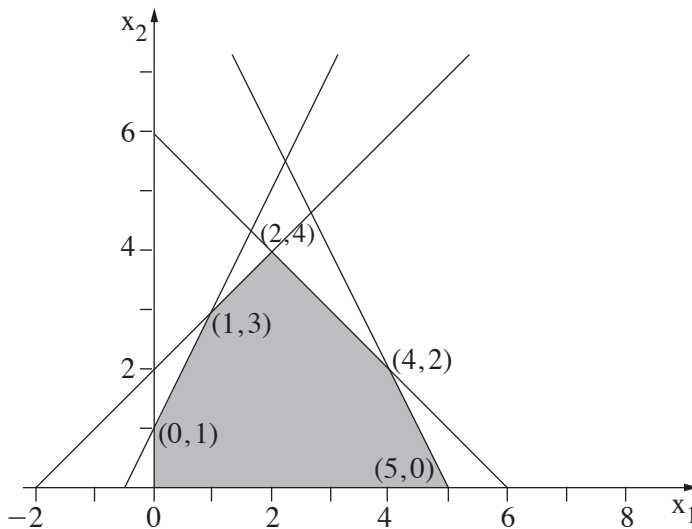
$$(3) \quad -x_1 + x_2 \leq 2$$

$$(4) \quad -2x_1 + x_2 \leq 1$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

לפניך סרטוט של תחום הפתרונות האפשריים של הבעיה הנתונה.



(שים לב: סעיפי השאלה בעמוד הבא.)

כל אחד מהסעיפים א-ד מתייחס לבעיית התכנון הלינארי הנתונה.
הסעיפים אינם תלויים זה בזה. ענה על כל הסעיפים.

- א. לפניך שני תת-סעיפים (1)-(2) שאינם תלויים זה בזה. בכל אחד מהם נתון ערך מסוים של הפרמטר k .
- (1) $k = -1$
- (2) $k = -3$

בעבור כל אחד מהתת-סעיפים (1)-(2) נתונים ארבעה היגדים iv-i שלפניך:

i הפתרון האופטימלי הוא יחיד.

ii יש אין-סוף פתרונות אופטימליים.

iii הפתרון האופטימלי לא חסום.

iv אין פתרון אופטימלי.

ובכל תת-סעיף רק אחד מההיגדים iv-i הוא נכון.

לכל אחד מתת-הסעיפים (1)-(2) קבע איזה היגד הוא הנכון, העתק אותו למחברתך, ונמק את קביעתך.

- אם קבעת שהיגד i הוא הנכון — עליך למצוא את הפתרון האופטימלי היחיד בתת-הסעיף הזה, ואת ערכה של פונקציית המטרה בפתרון שמצאת.
- אם קבעת שהיגד ii הוא הנכון — עליך למצוא את הפתרון האופטימלי הכללי לבעיה, ואת ערכה של פונקציית המטרה בתחום הפתרונות האופטימליים.

ב. בעבור אילו ערכים של k יהיה (2, 4) הפתרון האופטימלי לבעיית התכנון הלינארי הנתונה בתחילת השאלה? נמק את תשובתך.

ג. מבטלים את האילוץ $x_1 \geq 0$ שבבעיה הנתונה בתחילת השאלה, כלומר $-\infty \leq x_1 \leq \infty$.

האם קיים ערך של k שבעבורו הפתרון האופטימלי יהיה לא חסום? נמק את תשובתך.

ד. במקום האילוץ $x_2 \geq 0$ שבבעיה הנתונה בתחילת השאלה נכתוב: $x_2 \geq 4$.

האם לבעיה יש פתרון אופטימלי לאחר שינוי האילוץ? אם כן — כתוב מהו הפתרון, ואם לא — הסבר מדוע אין פתרון אופטימלי.

10. בשאלה זו שלושה סעיפים, א-ג, שאין קשר ביניהם, ענה על שלושתם.

א. $G = (V, E)$ הוא גרף מכוון המיוצג על ידי מטריצת הסמיכויות שלפניך:

	a	b	c	d	e
a	0	1	1	0	0
b	0	0	0	1	0
c	0	1	0	0	0
d	0	0	1	0	1
e	0	1	0	0	0

(1) סרטט את הגרף G המיוצג על ידי המטריצה.

(2) מצא את רכיבי הקשירות החזקה (Strong Connected Components) —

רק"חיים) שבגרף הנתון. בעבור כל רק"ח שמצאת רשום את קבוצת הקדקודים שלו.

(3) מצא בגרף הנתון מעגל באורך מינימלי ושמשפר הקשתות בו זוגי, וסרטט אותו במחברתך.

(4) מהו המספר המינימלי של קשתות שיש להוסיף לגרף הנתון כדי שהגרף יכיל

רק"ח אחד בלבד? מהי הקשת או מה הן הקשתות שיש להוסיף?

ב. בטבלה שלפניך נתון חלק מפתרון בסיסי אפשרי לבעיית התובלה: $x_{11} = 10$.

מקורות	יעדים				היצע
	1	2	3	4	
1	20	20	30	21	20
	10				
2	22	17	29	30	30
3	10	24	26	38	10
ביקוש	10	20	20	10	

(1) העתק את הטבלה למחברתך והשלם בה את הפתרון הבסיסי האפשרי לפי שיטת הפינה הצפונית מערבית.

(2) משנים את המחיר בתא (1, 3) מ-10 ל-20. האם ישתנה הפתרון הבסיסי האפשרי שמצאת בתת-סעיף (1)? נמק.

(שים לב: סעיף ג של השאלה בעמוד הבא.)

ג. בטבלה שלפניך נתון פתרון בסיסי אפשרי לבעיית תובלה, ונתון ערכו של u_3 .

מקורות	יעדים			היצע	u_i
	1	2	3		
1	10	8	3	10	
2	12	15	9	3	
3	2	7	1	90	0
ביקוש	20	40	43		
v_j					

(1) העתק את הטבלה למחברתך והשלם בה את הערכים של u_1, u_2, v_1, v_2, v_3 .

(2) הסבר מדוע הפתרון הנתון אינו פתרון אופטימלי.

/המשך בעמוד 21/

מודלים חישוביים

אם למדת מסלול זה, ענה על אחת מהשאלות 11-12 (25 נקודות).

11. לפניך השפות הרגולריות L_1 ו- L_2 .

$$L_1 = \{a^{2n} \mid n \geq 0\}$$

$$L_2 = \{b^{2n+1} \mid n \geq 0\}$$

נתונה השפה L מעל הא"ב $\{a, b\}$.

$$L = \{a^n b^k \mid n \geq 0, k \geq 0, n \text{ אי-זוגי ו-} k \text{ זוגי או } n \text{ זוגי ו-} k \text{ אי-זוגי}\}$$

א. הוכח, באמצעות השפות L_1 ו- L_2 בלבד ובאמצעות תכונות סגירות בלבד, שהשפה L היא רגולרית.

ב. בנה אוטומט סופי לא דטרמיניסטי שיקבל את השפה L .

12. נתונה פעולה הכתובה ב-Java וב-C#.

הפעולה מקבלת שלושה מספרים שלמים גדולים מ-0.

```

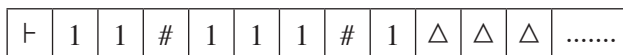
Java
public static int foo(int x , int y , int z)
{
    if ((x % 3) == 0) return x;
    if ((x % 3) == 1) return y;
    return z;
}
    
```

```

C#
public static int Foo(int x , int y , int z)
{
    if ((x % 3) == 0) return x;
    if ((x % 3) == 1) return y;
    return z;
}
    
```

כתוב מכונת טיורינג שתממש את הפעולה הנתונה.

הקלט של המכונה הוא שלושת המספרים x, y, z שהפעולה מקבלת והוא כתוב על הסרט מתחילתו. כל מספר כתוב בצורה אונרית. בין מספר למספר מפריד סימן #. למשל אם הפעולה מקבלת 2 ל- x , 3 ל- y , ו-1 ל- z , הסרט יראה כך:



↑
ראש קורא כותב

הפלט הוא הערך שהפעולה מחזירה והוא ייכתב על הסרט במקום כלשהו כערך אונרי

בין שני סימני \$.

תכנות מונחה עצמים

אם למדת מסלול זה ואתה כותב ב־ Java, ענה על אחת מהשאלות 13-14. (25 נקודות)

13. בשאלה זו שני סעיפים א-ב שאין קשר ביניהם. ענה על שניהם.

א. נתונות חמש מחלקות B, X, Y, Z, Run . התת־סעיפים (1)-(4) שלפניך מתייחסים למחלקות אלה. התת־סעיפים אינם קשורים זה לזה. ענה על כולם.

(1) בכל אחת מהמחלקות B, X, Y, Z הוגדרה הפעולה `public void foo() {}` ובמחלקה `Run` הוגדרה הפעולה `public void bar (Object g) {g.foo(); }`

האם תתקבל שגיאת הידור (קומפילציה)? אם כן – הסבר מדוע.

(2) הנח שהמחלקות B, X, Y, Z יורשות את המחלקה `B`.

בכל אחת מהמחלקות B, X, Y, Z הוגדרה הפעולה `public void foo() {}` ובמחלקה `Run` הוגדרה הפעולה `public void bar(B g) {g.foo(); }`

האם תתקבל שגיאת הידור (קומפילציה)? אם כן – הסבר מדוע.

(3) הנח שהמחלקות B, X, Y, Z יורשות את המחלקה `B`.

בכל אחת מהמחלקות B, X, Y, Z הוגדרה הפעולה `public void foo() {}` ובמחלקה `Run` הוגדרה הפעולה `public void bar(B g) {g.foo(); }`

האם תתקבל שגיאת הידור (קומפילציה)? אם כן – הסבר מדוע.

(4) הנח שהמחלקות B, X, Y, Z יורשות את המחלקה `B`.

בכל אחת מהמחלקות B, X, Y, Z הוגדרה הפעולה `public void foo() {}` ובמחלקה `Run` הוגדרה הפעולה `public void bar(Object g) {g.foo(); }`

האם תתקבל שגיאת הידור (קומפילציה)? אם כן – הסבר מדוע.

ג. (אין קשר לסעיף א).

לפניך המחלקה **Singer** שיורשת את המחלקה **Artist** והממשק **IPerform** שהמחלקה **Singer** מממשת.

```
interface IPerform
{
    void act();
    int train();
}
public class Singer extends Artist implements IPerform
{
    private int hits;
    public Singer(String name, double sal)
    {
        super(sal, name);
        this.hits = 5;
    }
    public Singer(double sal, int hits)
    {
        super(sal, "Singer Name");
        this.hits = hits;
    }
    public Singer(int hits)
    {
        super(6532.6, "Some", "One");
        this.hits = hits;
    }
    public double value() {return this.hits * this.price();}
    public int getNum()   {return Artist.num;           }
    public double calc(double d) {return d * super.calc(10.2); }
    public void print()   {super.print(); System.out.println("Singer");}
    public void act()     {System.out.println("I am singing"); }
}
```

כתוב ב־ Java את כותרת המחלקה **Artist**, את התכונות ואת כותרות הפעולות המתחייבות מהמחלקה **Singer** והממשק **IPerform** הנתונים. אין צורך לממש את הפעולות של המחלקה **Artist**.

14. לפניך המחלקות A, B.

שלושת הסעיפים א-ג מתייחסים למחלקות האלה, אך אינם קשורים זה לזה. ענה על שלושתם.

```
public class A {
    private int x;
    public A() { this.x = 0; }
    public A(int x) { this.x = x; }
    public int getX() { return x; }
    public void doubleX() { this.x = 2 * getX(); }
    public void tripleX() { this.x = 3 * getX(); }
    public void sub() { this.x = x - 1; }
    public void calc() { sub(); }
    public String toString() { return "xA="+this.x; }
}

public class B extends A {
    private int x;
    public B() { super(); this.x = 1; }
    public B(int x) { super(x); this.x = -x; }
    public B(int xA, int xB) { super(xA); this.x = xB; }
    public int getX() { return x; }
    public int baseX() { return super.getX(); }
    public void tenTimesX() { this.x = 10 * getX(); }
    public void tripleX() { this.x = 3 * getX(); }
    public void sub() { this.x = x - 2; }
    public String toString() { return super.toString()+" xB="+this.x+"; }
}
```

/המשך בעמוד 25/

א. לפניך סדרת הוראות שהפלט בעבורה צריך להיות:

$x_A=1 \ x_B=20$; $x_A=1 \ x_B=20$; $x_A=1 \ x_B=20$; $x_A=1 \ x_B=20$;

בסדרת ההוראות נפלה שגיאת הידור (קומפילציה). תקן את השגיאה כדי שיתקבל הפלט הנכון.

```
A a1 = new B(1, 20);
Object obj = a1;
B b1 = a1;
A a2 = a1;
System.out.println(a1+" "+obj+" "+a2+" "+b1);
```

ב. לפניך סדרת הוראות. הצג את העצמים הנוצרים, ולכל עצם הצג את ערכי התכונות שלו.

```
A aa = new B(3, 10);
aa.sub();
Object[] ar = new Object[6];
ar[0] = new A();
ar[1] = new A(5);
ar[2] = new B();
ar[3] = new B(5);
ar[4] = new B(2, 4);
ar[5] = aa;
((A)ar[3]).tripleX();
((B)ar[4]).tenTimesX();
```

(שים לב: סעיף ג של השאלה בעמוד הבא.)

ג. לפניך פעולה ראשית.

```
public static void main(String[] args)
{
    A a1 = new A(1);
    A a2 = new B(2, 99);
    /***
}
```

לפניך הקטעים (i)-(vi).

- (i) a2.doubleX();
System.out.println(a2);
- (ii) a2.tenTimesX();
System.out.println(a2.tenTimesX());
- (iii) if (a2 instanceof B)
 - {
 - a2.tenTimesX();
 - System.out.println(a2);
 - }
- (iv) ((B)a1).tenTimesX();
System.out.println(a1);
- (v) a2.calc();
System.out.println(a2);
- (vi) B bb = (B)a2;
System.out.println(bb.baseX());

בעבור כל אחד מן הקטעים (i)-(vi) בצע:

— כתוב את הקטע במקום /*** בפעולה הראשית.

— קבע אם הקטע תקין או אינו תקין.

— אם הקטע תקין — כתוב את הפלט המתקבל בעקבות ביצועו.

אם הקטע אינו תקין — כתוב אם השגיאה היא שגיאת הידור (קומפילציה) או

שגיאת זמן ריצה.

תכנות מונחה עצמים

אם למדת מסלול זה ואתה כותב ב- C#, ענה על אחת מהשאלות 15-16. (25 נקודות)

15. בשאלה זו שני סעיפים א-ב שאין קשר ביניהם. ענה על שניהם.

א. נתונות חמש מחלקות `B`, `X`, `Y`, `Z`, `Run`, התת סעיפים (1)-(4) שלפניך מתייחסים למחלקות אלה. התת-סעיפים אינם קשורים זה לזה. ענה על כולם.

(1) בכל אחת מהמחלקות `B`, `X`, `Y`, `Z` הוגדרה הפעולה `public virtual void Foo()` ובמחלקה `Run` הוגדרה הפעולה `public void Bar(Object g) {g.Foo(); }`

האם תתקבל שגיאת הידור (קומפילציה)? אם כן – הסבר מדוע.

(2) הנח שהמחלקות `X`, `Y`, `Z` יורשות את המחלקה `B`.

בכל אחת מהמחלקות `X`, `Y`, `Z` הוגדרה הפעולה `public virtual void Foo() { }` ובמחלקה `Run` הוגדרה הפעולה `public void Bar(B g) {g.Foo(); }`

האם תתקבל שגיאת הידור (קומפילציה)? אם כן – הסבר מדוע.

(3) הנח שהמחלקות `X`, `Y`, `Z` יורשות את המחלקה `B`.

בכל אחת מהמחלקות `B`, `X`, `Y`, `Z` הוגדרה הפעולה `public virtual void Foo()` ובמחלקה `Run` הוגדרה הפעולה `public void Bar(B g) {g.Foo(); }`

האם תתקבל שגיאת הידור (קומפילציה)? אם כן – הסבר מדוע.

(4) הנח שהמחלקות `X`, `Y`, `Z` יורשות את המחלקה `B`.

בכל אחת מהמחלקות `B`, `X`, `Y`, `Z` הוגדרה הפעולה `public virtual void Foo() { }` ובמחלקה `Run` הוגדרה הפעולה `public void Bar(Object g) {g.Foo(); }`

האם תתקבל שגיאת הידור (קומפילציה)? אם כן – הסבר מדוע.

(שים לב: סעיף ב של השאלה בעמוד הבא.)

ג. (אין קשר לסעיף א).

לפניך המחלקה **Singer** שיורשת את המחלקה **Artist** והממשק **IPerform** שהמחלקה **Singer** מממשת.

```
interface IPerform
{
    void Act();
    int Train();
}
public class Singer : Artist , IPerform
{
    private int hits;
    public Singer(string name, double sal) : base(sal, name)
    {
        this.hits = 5;
    }
    public Singer(double sal, int hits) : base(sal, "Singer Name")
    {
        this.hits = hits;
    }
    public Singer(int hits) : base(6532.6, "Some", "One")
    {
        this.hits = hits;
    }
    public double Value()          {return this.hits * this.Price(); }
    public int GetNum()            {return Artist.num;          }
    public override double Calc(double d) {return d * base.Calc(10.2); }
    public override void Print()    {base.Print(); Console.WriteLine("Singer"); }
    public void Act()              {Console.WriteLine("I am singing"); }
}
```

כתוב ב- C# את כותרת המחלקה **Artist**, את התכונות ואת הפעולות המתחייבות מהמחלקה **Singer** והממשק **IPerform** הנתונים. אין צורך לממש את הפעולות של המחלקה **Artist**.
אין לשנות את המחלקה **Singer** ואת הממשק **IPerform**.

16. לפניך המחלקות A, B.

שלושת הסעיפים א-ג מתייחסים למחלקות האלה אך אינם קשורים זה לזה. ענה על שלושתם.

```
public class A {
    private int x;
    public A() { this.x = 0; }
    public A(int x) { this.x = x; }
    public virtual int GetX() { return x; }
    public void DoubleX() { this.x = 2 * GetX(); }
    public virtual void TripleX() { this.x = 3 * GetX(); }
    public virtual void Sub() { this.x = x - 1; }
    public void Calc() { Sub(); }
    public override string ToString(){ return "xA="+this.x; }
}

public class B:A {
    private int x;
    public B() : base() { this.x = 1; }
    public B(int x) : base(x) { this.x = -x; }
    public B(int xA, int xB) : base(xA) { this.x = xB; }
    public override int GetX() { return x; }
    public int BaseX() { return base.GetX(); }
    public void TenTimesX() { this.x = 10 * GetX(); }
    public override void TripleX() { this.x = 3 * GetX(); }
    public override void Sub() { this.x = x - 2; }
    public override string ToString(){ return base.ToString()+" xB="+this.x+";"}
}
```

(שים לב: סעיפי השאלה בעמודים הבאים.)

א. לפניך סדרת הוראות שהפלט עבורה צריך להיות:

$x_A=1 \ x_B=20$; $x_A=1 \ x_B=20$; $x_A=1 \ x_B=20$; $x_A=1 \ x_B=20$;

בסדרת ההוראות נפלה שגיאת הידור (קומפילציה). תקן את השגיאה כדי שיתקבל

הפלט הנכון.

```
A a1 = new B(1, 20);
```

```
Object obj = a1;
```

```
B b1 = a1;
```

```
A a2 = a1;
```

```
Console.WriteLine(a1+" "+obj+" "+a2+" "+b1);
```

ב. לפניך סדרת הוראות. הצג את העצמים הנוצרים, ולכל עצם הצג את ערכי התכונות שלו.

```
A aa = new B(3, 10);
```

```
aa.Sub();
```

```
Object[] ar = new Object[6];
```

```
ar[0] = new A();
```

```
ar[1] = new A(5);
```

```
ar[2] = new B();
```

```
ar[3] = new B(5);
```

```
ar[4] = new B(2, 4);
```

```
ar[5] = aa;
```

```
((A)ar[3]).TripleX();
```

```
((B)ar[4]).TenTimesX();
```

/המשך בעמוד 31/

ג. לפניך פעולה ראשית.

```
public static void Main()
{
    A a1 = new A(1);
    A a2 = new B(2, 99);
    /***
}
```

לפניך הקטעים (i)-(vi).

- (i) a2.DoubleX();
Console.WriteLine(a2);
- (ii) a2.TenTimesX();
Console.WriteLine(a2.TenTimesX());
- (iii) if (a2 is B)
{
 a2.TenTimesX();
 Console.WriteLine(a2);
}
- (iv) ((B)a1).TenTimesX();
Console.WriteLine(a1);
- (v) a2.Calc();
Console.WriteLine(a2);
- (vi) B bb = (B)a2;
Console.WriteLine(bb.BaseX());

בעבור כל אחד מן הקטעים (i)-(vi) בצע:

- כתוב את הקטע במקום /*** בפעולה הראשית.
- קבע אם הקטע תקין או אינו תקין.
- אם הקטע תקין — כתוב את הפלט המתקבל בעקבות ביצועו.
- אם הקטע אינו תקין — כתוב אם השגיאה היא שגיאת הידור (קומפילציה) או שגיאת זמן ריצה.

בהצלחה!