

סוג הבדיקה: א. בגרות לבתי ספר על-יסודיים

ב. בגרות לנבחנים חיצוניים

מועד הבדיקה: קיץ תשע"ו, 2016

מספר השאלה: 603,899205

מְדֻעַי הַמִּחְשָׁב ב'

2 ייחדות לימוד (השלמה ל-5 י"ל)

הוראות לנבחן

א. משך הבדיקה: שלוש שעות.

- ב. מבנה השאלה ופתחה הערכה: בשאלון זה שני פרקים.
פרק ראשון – בפרק זה ארבע שאלות, ומהן יש לענות על שתיים.
פרק שני – בפרק זה שאלות באربعة מסלולים שונים. ענה על שאלות בק במסלול שלמדת, לפי ההוראות בקבוצת השאלות במסלול זה.
סה"כ – 100 נקודות

ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ מחשב הנitinן לתכניות.ד. הוראות מיוחדות:

- (1) את כל התכניות שאתה נדרש לכתוב בשפת מחשב בפרק הראשון 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 ואסambilי, מבוא לחקיר ביצועים, מודלים חישוביים, תכניות מונחה עצמים.
הערה: בתכניות שאתה כותב לא יורדו לך נקודות, אם כתובות אותן גדולה במקוםאות קטנה או להפר.

כתב במחברת הבדיקה בלבד, בעמודים נפרדים, כל מה שברצונך לכתוב בטיטה (ראשי פרקים, חישובים וכדומה). רשום "טיוטה" בראש כל עמוד טיוטה. רישום טיותות כלשון על דפים שהוחוץ למחברת הבדיקה עלול לגרום לפסילת הבדיקה!

הנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות ולນבחנים כאחד.

בהתלהה!

/המשר מעבר לדף/

השאלות

בשאלון זה שני פרקים: פרק ראשון ופרק שני.

עליך לענות על שאלות משני הפרקם, לפי ההוראות בכל פרק.

פרק ראשון (50 נקודות)

שים לב: בכל שאלה שנדרש בה שימוש אתה יכול להשתמש בפעולות של המחלקות `Tor`, `Machsנית`, עצ' בירני וחוליה, בלי למשוך אותן. אם אתה משתמש בפעולות נוספות, عليك למשוך אותן.

ענה על שתיים מהשאלות 1-4 (לכל שאלה — 25 נקודות).

- .1. בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניים.
א. לפניה פעולה מקבלת כפרמטר תור q של מספרים שלמים הממוינים בסדר עולה. כל המספרים שונים זה מזה.
הנח שבתור יש לפחות שני מספרים שלמים.

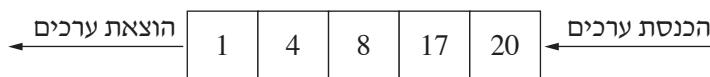
Java

```
public static int what(Queue<Integer> q)
{
    int x = q.remove();
    int y = q.head();
    q.insert(x);
    if (x>y)
        return 0;
    return(what(q) + (y-x));
}
```

C#

```
public static int What(Queue <int> q)
{
    int x = q.Remove();
    int y = q.Head();
    q.Insert(x);
    if (x>y)
        return 0;
    return (What(q) + (y-x));
}
```

(1) נתון התור q1.



עקבות אחר הפעולה `what` ב- C# או `Java` ב- What ב- C# בעבר התור הנוכחי q1.

וכתב את הערך שיותזה. הראה את המיעקב הרקורסיבי.

במיוקב הראה את הערכים של x ושל y ואת התור לאחר כל זימון רקורסיבי.

(2) (i) מה מכיון הפעולה `what` ב- Java או `What` ב- C# בעבר תור כלשהו q

של מספרים שלמים, הממוינים בסדר עולה ושיונים זה מזוה?

(ii) כיצד נראה התור לאחר סיום הפעולה?

(3) אם נחליף את התנאי `if(x>y)` בתנאי:

`Java if(q.isEmpty())`

או:

`C# if(q.IsEmpty())`

תיווצר שגיאת זמן ריצה.

מהי השגיאה שתיווצר? הסבר את תשובהך.

(שים לב: סעיף ב של השאלה בעמוד הבא.)

המשך בעמוד 4/

.ב. (אין קשר לסעיף א.)

לפניך פעולה המקבלת מערך חד-ממדי ar מטיפוס שלם, ושני מספרים שלמים x ו- y
גדולים מ- 0 או שווים לו, וקטנים מגודל המערך, ו- $x \leq y$.

Java	C#
<pre>public static int go(int[] ar , int x , int y) { if (x == y) return (ar[x] % 10) ; if ((ar[x] % 10) < (ar[y] % 10)) return go(ar, x + 1, y); return go(ar, x, y-1); }</pre>	<pre>public static int Go(int[] ar , int x , int y) { if (x == y) return (ar[x] % 10) ; if ((ar[x] % 10) < (ar[y] % 10)) return Go(ar, x + 1, y); return Go(ar, x, y-1); }</pre>

לפניך מערך ar1 מטיפוס שלם.

0	1	2	3	4	5	6
ar1	21	219	166	61	127	60

? Go(ar1 , 2 , 5) C# go(ar1 , 2 , 5) Java או ב- (1)

הראה את המאך הרקורסיבי. במעקב הראה את הערכיהם של:

. ar1[y] , ar1[x] , y , x

? go(ar1, 0, ar1.length-1) Java (2)

או ב- ? Go(ar1, 0, ar1.Length-1) C# (3)

כתב מה מבצעת הפעולה:

ב- Java public static int go(int[] ar , int x , int y) :

ב- C# public static int Go(int[] ar , int x , int y) :

בעבור מערך ar כלשהו מטיפוס שלם ושני מספרים שלמים x ו- y ,

גדולים מ- 0 או שווים לו, וקטנים מגודל המערך, ו- $x \leq y$.

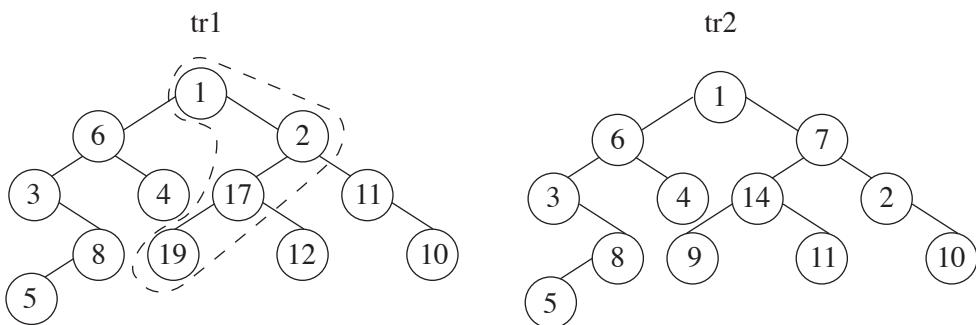
.2. **עַז מְסֻפָּרִים** הוא עַז בינהרֵי לא ריק מטיפוס שלם, שהערכים בצמתים שלו הם מספרים שלמים וגדולים מ- 0 השונים זה מזה.

על **עַז מְסֻפָּרִים** מוגדרת פעולה "מסלול-עליה", המחזירה true אם יש בעז מסלול המתחילה בשורש העז ומסתתרים באחד העלים שלו, וערכי הצמתים ממוקנים בסדר עולה מהשורש לעלה. אם אין מסלול כזה — הפעולה מחזירה false.

לדוגמה:

בעבור **עַז מְסֻפָּרִים** tr1 הפעולה "מסלול-עליה" מוחזירה true. המסלול מוקף בקו שברו.

בעבור **עַז מְסֻפָּרִים** tr2 הפעולה "מסלול-עליה" מוחזירה false.



משב Java או ב- C# את הפעולה "מסלול-עליה" בעבור **עַז מְסֻפָּרִים** tr.

គורת הפעולה ב- Java
`public static boolean upPath(BinNode<Integer> tr)`

គורת הפעולה ב- C#
`public static bool UpPath(BinNode<int> tr)`

לפניך המחלקה טבעת – **Ring** שליה 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 המכניתה 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 מוסיפה את הטבעת מרأس 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 של המוט כך: הטבעות הגדולות "מנוחות" בתחתית המוט וhteבעות הקטנות מעליהם.

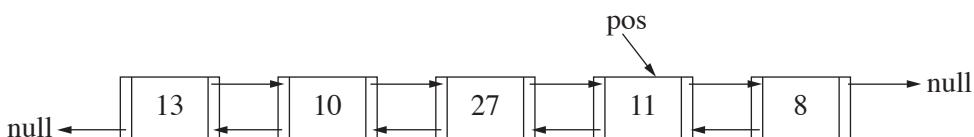
- א. ממש את הפעולה `sort()` ב- C# או `Sort()` ב- Java או `Sort()` ב- C#, שבמחלקה **Pole**, שבסדרת המחלקות **Pole** ו- **Ring**.
- ב. מהי סיבוכיות זמן הריצה של הפעולה שמימושה בסעיף א? נמק את תשובתך.

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

.4

פתרונות ב- Java

נגיד **רשיימה דו-כיוונית** כאוסף סדר של חוליות מטיפוס `BinNode<Integer>` הקשורות כך:
 לכל זוג חוליות `p1, p2` ברשימה, אם מתקיים `p1.getRight() == p2`, אז מתקיים גם
 `p1.getLeft() == p2`. **רשיימה 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 לרשימה 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.

A. לפניך שلد 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) עקב אחר ביצוע הפעולה בעבור המשתנה pos והרשימה שבדוגמה המוצגת בתחלת השאלה.

. pos, left, right, sum במקבב ראה את הרשימה הדו-כיוונית ואת ערכי המשתנים

```
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 שבסעה — השורות המוקפות במסגרת — בהוראה:

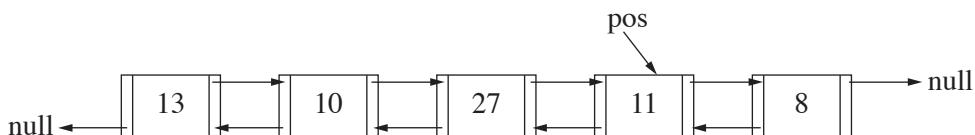
```
return left.getValue() + right.getValue() == sum;
```

נק את קביעתן.

לפותרים ב- C#

נגיד ר^שימנה דו-יכיונית כאוסף סדר של חוליות מטיפוס `BinNode<int>` המושו^רות כר':
לכל זוג חוליות `p1`, `p2` בראשימה, אם מתקיים `p1.GetRight() == p2`, או מתקיים גם
`p1.GetLeft() == p2`. ברשימנה דו-יכיונית יש לפחות שתי חוליות.

כלומר: כל חוליה בראשימה — חוץ מהחוליה שבקצתה הימני של הרשימה והחוליה שבקצתה השמאלי של הרשימה — מצביע על החוליה ש לפניה ועל החוליה שאחריה.
לפניך דוגמה לרשימה דו-יכיונית ומשתנה `pos` מטיפוס `BinNode<int>` המצביע על חוליה כלשהי בראשימה דו-יכיונית.



הפעולה `FirstLeft` מקבלת מצביע `pos` שונה מ-`null` מטיפוס `BinNode<int>` המצביע על חוליה כלשהי בראשימה דו-יכיונית ומהזירה את החוליה השמאלית ביותר בראשימה.

הפעולה `FirstRight` מקבלת מצביע `pos` שונה מ-`null` מטיפוס `BinNode<int>` המצביע על חוליה כלשהי בראשימה דו-יכיונית ומהזירה את החוליה הימנית ביותר בראשימה.

א. לפניך שلد של הפעולה `FirstLeft`.

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

```

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

לפניך הפעולה `What(BinNode<int> pos)` מקבלת מצביע לחוליה כלשהי. ברשימה דואכיאונית ומחזירה ערך בוליאני. הרשימה הדואכיאונית מכילה לפחות 3 חוליות.
 (1) עקב אחר ביצוע הפעולה בעבור המשתנה `pos` והרשימה שבדוגמה המוצגת בתחילת השאלה.
 . `pos, left, right, sum` במקבב הראה את הרשימה הדואכיאונית ואת ערכי המשתנים

```
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 להחליק את 3 השורות האחרונות שבסעה – השורות המוקפות בסגרת – בהוראה:
`return left.GetValue() + right.GetValue() == sum;`

נק את קביעתך.

פרק שני (50 נקודות)

בפרק זה שאלות בארבעה מסלולים:

מערכות מחשב ואסמבלי, עמודים 12-18.

מבוא לחקיר ביצועים, עמודים 19-27.

מודלים חישוביים, עמודים 28-29.

תכנות מונחה עצמים ב- Java , עמ' 30-39; תכנות מונחה עצמים ב- C# , עמודים 40-49.

ענה רק על שאלות במסלול שלמדת.

מערכות מחשב ואסמבלי

אם למדת מסלול זה, ענה על שתיים מהשאלות 5-8 (לכל שאלה — 25 נקודות).

.5. בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניים.

א. במקטע נתונים הוגדרו הנתונים כר:

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 התוכנית וסרטט את VEC2 לאחר הביצוע.

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

(2) מה מבצע קטע התוכנית?

(3) IMUL AH בקטע התוכנית, במקומות ההוראה המסומנת ב-(*) נכתבה ההוראה: האם ביצוע קטע התוכנית ישנה? נמיך את תשובתך.

ב. (אין קשר לסעיף א.).

לפניך קטע תוכנית באסמבלי.

MOV AX , 14H
MOV BL , 0AH
DIV BL
MOV CL , 4
ROL AL , CL
OR AL , AH

(1) עקוב בעזרת טבלת מעקב אחר הביצוע של קטע התוכנית.

.CL , BL , AH , AL , AX בטבלת המעקב יש לכלול עמודה לכל אחד מהאוגרים

(2) מה מבצע קטע התוכנית?

/המשך בעמוד 14/

- .6. כדי לבדוק אם מספר מתחולק ב- 6 בלי שארית מחלקים אותו ב- 2.
אם הוא מותחלק ב- 2 בלי שארית – מחלקים את התוצאה ב- 3.
אם התוצאה מתחולקת ב- 3 בלי שארית – המספר מתחולק ב- 6 בלי שארית.
א. כתוב באסמבלי שגרה (פrozדורה) בשם DIVBYTWO שתתקבל באמצעות המחסנית
מספר בן 16 סיביות, ותאחסן 0 ב- AL אם המספר מתחולק ב- 2 בלי שארית.
אחרת – השגרה תאחסן 1 ב- AL.
בכתיבת השגרה אין להשתמש בהוראה DIV.
- .ב. כתוב באסמבלי שגרה (פrozדורה) בשם DIVBYTHREE שתתקבל באמצעות המחסנית
מספר בן 16 סיביות, ותאחסן 0 ב- AL אם המספר מתחולק ב- 3 בלי שארית.
אחרת – השגרה תאחסן 1 ב- AL.
בכתיבת השגרה אפשר להשתמש בכל הוראה.
- .ג. כתוב באסמבלי שגרה (פrozדורה) בשם DIVBYSIX שתתקבל באמצעות המחסנית
מספר בן 16 סיביות, ותאחסן 0 ב- AL אם המספר מתחולק ב- 6 בלי שארית.
אחרת – השגרה תאחסן 1 ב- AL.
עליך להשתמש בשגרות שכתבת בסעיפים א-ב.
.ד. במקטע הנתונים הוגדר מערך ARR בן 8 איברים שכל אחד מהם בגודל מילה. המערך
מכיל מספרים.
כתוב באסמבלי קטע תכנית שידפיס את מספר האיברים במערך שהערכיהם המואחסנים
בינם מתחולקים ב- 6 בלי שארית.
עליך להשתמש בשגרה שכתבת בסעיף ג.

- .7. בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניהם.
- א. כתוב באסמליל קטע תכנית להצפנה טו.
- הטו מוגדר כך במקטיע 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 שמאליה 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 לא נכון. אם ההיגד אינו נכון, הסבר מדוע.

(1) לפני קטע תכנית באסמלילי.

MOV AX , 8

MOV BX , 2

DIV BX

בסיום קטע התכנית האוגר AX יוכל בהכרח 4.

(2) לפני קטע תכנית באסמלילי.

MOV AL , 56

ADD AL , 200

JZ STOP

INC AL

STOP:

בסיום קטע התכנית האוגר AX יוכל 1.

(שים לב: המשך הסעיף בעמוד הבא).

(3) לפניך קטע תוכנית באסמבלי.

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

בסיום קטע התוכנית האוגר AX יכיל 3.

(4) לפניך קטע תוכנית באסמבלי.

```
MOV CX , 3  
MOV AX , 1  
DO:  
    SHL AX , 1  
    LOOP DO
```

בסיום קטע התוכנית האוגר AX יכיל 8.

(5) לפניך קטע תוכנית באסמבלי.

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

המספר הנמצא ב-AX בהתאם גدول מהמספר הנמצא ב-BX.

(6) לפניך הוראה:

```
OR AL , 3
```

לאחר ביצוע ההוראה ערכו של AL תמיד אי-זוגי.

- .8. בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שנייהם.
- א. באוגר AL מאוחסן המספר 2 ובאוגר BL מאוחסן המספר 5. יש לאחסן באוגר DX את סכום המספרים מ- AL עד BL (כולל) לפניר 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	

(שים לב: סעיף ב של השאלה בעמוד הבא.)

ב. (אין קשר לסעיף א.).

לפניך קטע תוכנית באסמבלי.

שים לב: המספרים הם הקסדצימליים.

```
START: MOV AX ,0C72AH  
       MOV BX ,0A98DH  
       SHR AX ,1  
       OR AL ,17H  
       NOT BH  
       ADD AX ,BX
```

עקבות בעזרת טבלת מעקב אחר ביצוע קטע התוכנית.

בטבלת המעקב יש לכלול עמודה לכל אחד מהדגלים:

,ZF,SF,OF,CF

ולכל אחד מהאוגרים:

.BX,AX

/המשך בעמוד 19/

מבוא לחקור ביצועים

אם למדת מסלול זה, ענה על שתיים מהשאלות 9-12 (לכל שאלה – 25 נקודות).

.9. נתונה בעיית תכנון לינארי:

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

בכפוף לאיוצים האלה:

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

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

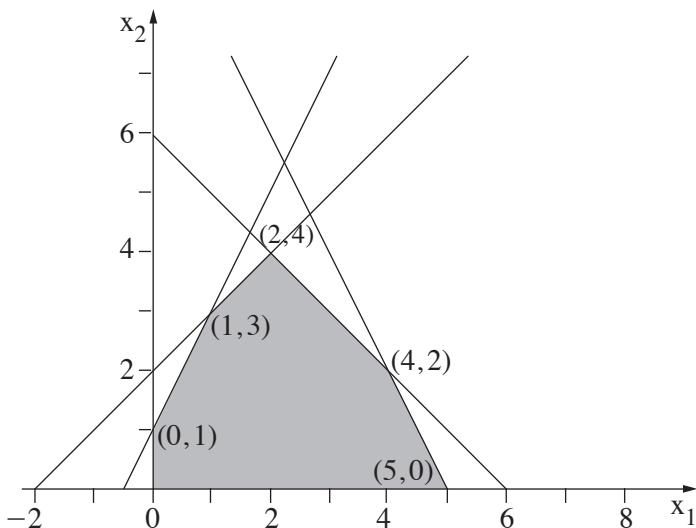
$$(3) \quad -x_1 + x_2 \leq 2$$

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

$$x_1 \geq 0$$

$$x_2 \geq 0$$

לפניך סרטוט של תחום הפתרונות האפשריים של הבעיה הנתונה.



(שים לב: סעיפים השאלה בעמוד הבא.)
/ המשך בעמוד 20

כל אחד מהסעיפים א-ד מתייחס לביעית התכנון הלינארי הנתונה.
הסעיפים אינם תלויים זה זהה. ענה על **כל** הטעיפים.

- א.** לפניך שני תת-סעיפים (1)-(2) **שאינם תלויים זה זהה**. בכל אחד מהם נתון ערך מסוים של הפרמטר k .

$$k = -1 \quad (1)$$

$$k = -3 \quad (2)$$

בעבור כל אחד מהתת-סעיפים (1)-(2) נתונים ארבעה היגדים ו-ז' 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 ו-ז' הוא נכון.

בלאحد מהתת-סעיפים (1)-(2) קבוע איזה היגד הוא נכון, העתק אותו למחברתך,
ונמק את קביעתך.

- אם קבועה שהיגד ז' הוא נכון — עלייך למצוא את הפתרון האופטימלי היחיד
בתת-סעיף זהה, ואת ערכיה של פונקציית המטריה בפתרון שמצוות.
 — אם קבועה שהיגד ז' הוא נכון — עלייך למצוא את הפתרון האופטימלי הכללי בעיה,
ואת ערכיה של פונקציית המטריה בתחום הפתרונות האופטימליים.

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

- ג.** מבטלים את האילוץ $0 \leq x_1 \leq \infty$ שבבעיה הנתונה בתחילת השאלה, כלומר $x_1 \leq \infty$.
 האם קיים ערך של k שבבעורו הפתרון האופטימלי יהיה לא חסום? נמק את תשובתך.
ד. במקום האילוץ $0 \leq x_2 \leq \infty$ שבבעיה הנתונה בתחילת השאלה כתוב: $4 \geq x_2$.
 האם בעיה יש פתרון אופטימלי לאחר שינוי האילוץ? אם כן — כתוב מהו הפתרון,
ואם לא — הסבר מדוע אין פתרון אופטימלי.

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

א. $G = (V, E)$ הוא גרף מכוון המיוצג על ידי רישימת הסמוכיות 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.

— מצא את רכיבי הקשרות החזקה (Strong Connected Components)

רק"ח) שבגרף הנתון, בעבר כל רק"ח שמצאת רשותם את קבוצת הקדוקדים שלו.

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

(4) מהו המספר המינימלי של קשתות שיש להוסיף לגרף הנתון כדי שהגרף יוכל רק"ח אחד בלבד? מהי הקשת או מה הן הקשתות שיש להוסיף?

.ב. אין קשר לסעיף א.)

א. $G = (V, E)$ הוא גרף לא מכוון המיוצג על ידי רישימת הסמוכיות שלפניך:

a	$\rightarrow b \rightarrow c \rightarrow \ $
b	$\rightarrow a \rightarrow c \rightarrow d \rightarrow e \rightarrow \ $
c	$\rightarrow d \rightarrow a \rightarrow b \rightarrow \ $
d	$\rightarrow c \rightarrow b \rightarrow e \rightarrow \ $
e	$\rightarrow b \rightarrow d \rightarrow \ $

(1) סרטט את הגרף G המיוצג על ידי רישימת הסמוכיות.

(2) כמה רכיבי קשרות (Connected Components) יש בגרף G, ומה הם?

(3) הפעל אלגוריתם סריקה לעומק (DFS) על הגרף G החל בקדקוד a.

סרטט במחברתך רק את העץ הפורש (DFS) / הייר הפורש (DFS) שמתקיים.

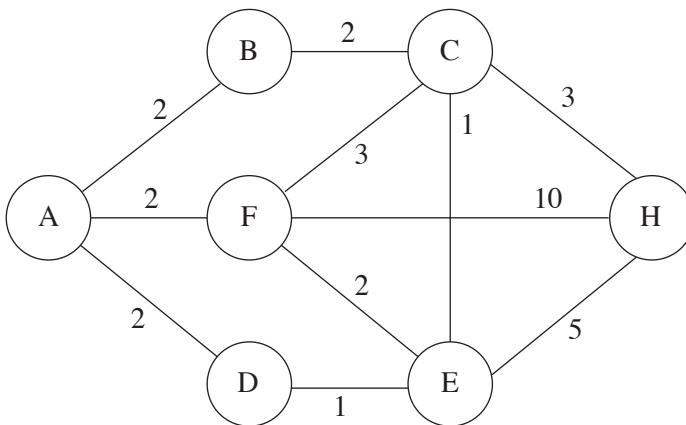
התבסס על היצוג הנתון על ידי רישימת הסמוכיות.

(4) הפעל אלגוריתם סריקה לרוחב (BFS) על הגרף G החל בקדקוד a.

סרטט במחברתך רק את העץ הפורש (BFS) / הייר הפורש (BFS) שמתקיים.

התבסס על היצוג הנתון על ידי רישימת הסמוכיות.

- בשאלה זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניהם.
- א. לפניך רשות (V, E) כאשר V הוא קבוצת הקודקודים ברשות ו- E קבוצת הקשתות ברשות. פונקציית המשקל $E \rightarrow R^+$: $w: E \rightarrow R^+$ (מספר גדול מ-0) לכל קשת ברשות.



- (1) מצא את כל המסלולים הקצרים ביותר ברשות הנתונה, החל מקדוקוד A לקדוקוד H.
הציג תיאור סכמטי של כל אחד מן המסלולים שמצאת.
- (2) מצא עץ פורש מינימלי לפי האלגוריתם של ק魯וסקל, מבין כל העצים הפורשים המכילים את שתי הקשתות האלה: (A, B) ו (B, C).
- סרטט במחברתך את העץ הפורש המינימלי המבוקש.

(אין קשר לסעיף א.). ב.

נתונה רשות $G = (V, E)$ כאשר V הוא קבוצת הקודדים ברשות ו- E קבוצת הקשתות ברשות. פונקציית המשקל $w: E \rightarrow R^+$ קובעת משקל (מספר גדול מ- 0) לכל קשת ברשות.

כמו כן נתונים קשת $e = (a, b)$, וקודוקדים s ו- t ברשות הנתונה.

לפניך אלגוריתם אשר מחזיר את הערך 1, אם הקשת הנתונה e נמצאת על כל המסלולים הקצרים ביותר, מקדקוד s לקודוק t . אחרת – האלגוריתם מחזיר את הערך 0.

באלגוריתם חסרים שני ביטויים, המסומנים (1) ו- (2) .

אלגוריתם:

צעד 1: הרץ את האלגוריתם של **דיקטורה** על הרשות G , ושמור במשתנה $M1$ את אורך המסלול הקצר ביותר מקדקוד s לקודוק t .

צעד 2: בנה רשות חדשה $E1 = G1 = (V1, E1)$ שבה $V1 = V$ שבה $E1 = (1)$.

צעד 3: הרץ את האלגוריתם של **דיקטורה** על הרשות $G1$, ושמור במשתנה $M2$ את אורך המסלול הקצר ביותר מקדקוד s לקודוק t .

צעד 4: אם $(2) < M1$ החזר 1.

אחרות – החזר 0.

העתק למחברתך את המספרים (1) ו- (2) , ועל יד כל אחד מהם כתוב את הביטוי החסר שהוא מייצג.

- . בא. בטבלה שלפניך נתנו חלק מפתרונות בסיסי אפשרי לבעיית התובלה: $x_{11} = 10$.
- . ב. בשאלת זו חמשה סעיפים, א-ה, שאינם תלויים זה זהה. ענה על כל הסעיפים.

מקורות	יעדים				היצע
	1	2	3	4	
1	20 10	20	30	21	20
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 מעירבית.
- (2) משנים את המחיר בתא (1,3) מ- 10 ל- 20. האם ישנה הפתרון הבסיסי האפשרי שמצאת בתת-סעיף (1)? נמק.
- בטבלה שלפניך נתנו פתרון בסיסי אפשרי לבעית תובלה, ונתנו ערכו של u_3 .

מקורות	יעדים			היצע	u_i
	1	2	3		
1	10	8	3 10	10	
2	12	15	9 3	3	
3	2 20	7 40	1 30	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.

ג. בטבלה שלפניך נתונה בעיית תובלה, ונתונים ערכיהם של $u_1, u_2, u_3, v_1, v_2, v_3$.

מקורות	יעדים			היצע	u_i
	1	2	3		
1	8	8	1	30	1
2	10	12	6	40	6
3	2	7	2	74	1
ביקוש	30	64	50		
v_j	1	6	0		

(1) העתק את הטבלה למחברתך וכותב בה פתרון בסיסי אפשרי תוך התחשבות בערכים של $v_3, v_2, v_1, u_1, u_2, u_3$.

(2) משנים בטבלה הנתונה את עלות התובלה ממוקור 1 ליעד 2 מ- 8 ל- 7. לאור שינוי זה קבע איזה מבין ההיגדים –ו-ו שלפניך הוא היחיד הנכון, והעתק אותו למחברתך. נמק את קביעותך.

i הפתרון הבסיסי לא משתנה, ונשאר פתרון ייחד המתאים לערכי u_i ו- v_j הנתונים.

ii יש שני פתרונות בסיסיים אפשריים **בלבד** המתאימים לערכי u_i ו- v_j הנתונים.

iii יש אינסוף פתרונות בסיסיים אפשריים המתאימים לערכי u_i ו- v_j הנתונים.
לאור השינוי אי אפשר למצוא פתרון בסיסי אפשרי המתאים לערכי u_i ו- v_j הנתונים.

(שם לב: המשך השאלה בעמוד הבא.)

.ד. בטבלה שלפניך נתון פתרון אפשרי לביעית תובלה, ונתונים ערכיהם של

$u_1, u_2, u_3, v_1, v_2, v_3$ שמתאימים לפתרון זה.

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

קבע איזה מבין ההיגדים ו- v_j שלפניך הוא היחיד הנכון, והעתק אותו למחברתך. נמק את קביעותך.

- i הפתרון הנתון אינו פתרון בסיסי אפשרי.
- ii הפתרון הנתון הוא פתרון בסיסי אפשרי אך אינו פתרון אופטימלי.
- iii הפתרון הנתון הוא פתרון אופטימלי יחיד.
- iv הפתרון הנתון הוא פתרון אופטימלי אך אינו פתרון אופטימלי יחיד.

(שים לב: המשך השאלה בעמוד הבא.)

ה. בטבלה של פניך נתון פתרון לא אופטימלי שהתקבל לאחר k איטרציות בעבר בעיתות טובלה

מסויימת, ונתונים ערכיהם של $u_1, u_2, u_3, v_1, v_2, v_3, v_4$.

מקורות	יעדים				היצע	u_i
	1	2	3	4		
1	4 6	4 4	14	25	10	3
2	6	1 5	13 10	14 3	18	0
3	0	8	14	12 2	2	-2
ביקוש	6	9	10	5		
v_j	1	1	13	14		

עליך לבצע איטרציה נוספת, כלומר איטרציה $k + 1$.

(1) מהו המשטנה שיוצא מהבסיס באיטרציה זו?

(2) סרטט במחברתך טבלה חדשה, ורשום בה את הפתרון שיתקבל לאחר איטרציה זו.

מודלים חישוביים

אם למדת מסלול זה, ענה על שתיים מהשאלות 13-16 (לכל שאלה – 25 נקודות).

13. לפניך השפות הרגולריות L_1 ו- L_2 .

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

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

נתונה השפה L מעלה הא"ב $\{a, b\}$.

$$L = \{a^n b^k \mid n \geq 0, k \geq 0\}$$

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 .

14. לפניך השפות L_1, L_2, L_3, L_4 , מעלה הא"ב $\{0, 1\}$.

$$L_1 = \{w \mid 010\text{ מכילה את הרץ}\}$$

$$L_2 = \{w \mid 00\text{ אינה מכילה את הרץ}\}$$

$$L_3 = \{w \mid \text{מספר ה-0ים בו שונה ממספר ה-1ים בו}\}$$

$$L_4 = \{w \mid w\text{ הוא אי-זוגי}\}$$

לפניך טענות (1)-(6).

לכל אחת מהטענות קבע אם היא נכונה או אינה נכונה.

אם הטענה נכונה — הסבר מדוע.

אם הטענה אינה נכונה — הסבר מדוע או כתוב מילה המפריכה את הטענה והסביר מדוע היא מפריכה אותה.

$$(1) \in \text{ שיצת לשפה } L_1 \cup L_3 \cup L_4$$

$$(2) \in 00100 \text{ שיצת לשפה } L_1 \cap \overline{L_4}$$

$$(3) \in \overline{L_2} = \{w \mid 11 \text{ w}\}$$

$$(4) \subseteq L_1 \subseteq L_3$$

$$(5) L_4 = R(L_4)$$

$$(6) L_3 \cap L_4 = L_2$$

.15. נתונה השפה L מעל הא"ב $\{a, b, c\}$.

$$L = \{a^{2n}c^kb^{3n+1} \mid n > 0, k > 0\}$$

- A. כתוב את המילה הקצרה ביותר בשפה L .
 B. בנה אוטומט מחסנית שיקבל את השפה L .

.16. נתונה פעולה הכתובת ב- Java ו- C#.

הפעולה מקבלת שלושה מספרים שלמים גדולים מ- 0.

Java

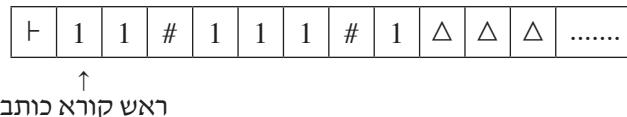
```
public static int foo(int x , int y , int z)
{
    if ((x % 3) == 0) return x;
    if ((x % 3) == 1) return y;
    return z;
}
```

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;
}
```

כתב 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 אונריית. בין מספר למספר מפheid הסימן $#$.
 למשל אם הפעולה מקבלת 2 ליד x , 3 ליד y , ו- 1 ליד z , הסרט יראה כך:



הפלט הוא הערך שהפעולה מחזירה והוא ייכתב על הסרט במקום כלשהו כערך אונריי בין שני סימני $$$.

תכונות מונחה עצמיים

אם למדת מסלול זה ואתה כותב ב- Java, ענה על שתיים מהשאלות 17-20.

(כל שאלה – 25 נקודות)

17. בשאלת זו שני סעיפים א-ב שאין קשור ביניהם. ענה על שניהם.

א. נתנות חמיש מחלקות B , Z , Y , X , Run. התת-סעיפים (1)-(4) שלפניך מתיחסים

لمחלקות אלה. התת-סעיפים אינם קשורים זה לזה. ענה על כולם.

(1) בכל אחת מהמחלקות B , Z , Y , X הוגדרה הפעולה

ובמחלקה Run הוגדרה הפעולה

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

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

בכל אחת מהמחלקות X , Y , Z הוגדרה הפעולה

ובמחלקה Run הוגדרה הפעולה

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

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

בכל אחת מהמחלקות B , Y , X , Z הוגדרה הפעולה

ובמחלקה Run הוגדרה הפעולה

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

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

בכל אחת מהמחלקות B , Z , Y , X הוגדרה הפעולה

ובמחלקה Run הוגדרה הפעולה

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

.ב. (אין קשר לסעיף א.)

לפניך המחלקה **Singer** שירשת את המחלקה **Artist** והמשק **IPerform** שהמחלקה **Singer** מימושת.

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 כוורתה המחלקה **Artist** , את התכונות ואת כוורות הפעולות המתחיבות מהמחלקה **Singer** והמשק **IPerform** הנתונים. אין צורך לממש את הפעולות של המחלקה **Artist**

בגן חיות נמצאות החיות האלה: אריות Lion, ברווזנים Platypus, נחשים Snake וצבים Turtle.

אריות וברווזנים הם יונקים, נחשים וצבים הם זוחלים.
הנקבות של כל הזוחלים בגן החיות מטילות ביצים, וגם נקבת הברווזן מטילה ביצים.
רוצים לבנות מערכת ממוחשבת שתציג את החיים בגן החיות. המערכת תראה תמונות וanimaciyah.
לשם כך הוגדרו המחלקות Animal – חייה, Mammal – יונק, Reptile – זוחל.
לכל חייה יש תכונה: שם – name, ופעולות: () move void turn() ו- .void move()
לחיות שטויות ביצים יש התכונה: מספר ממוצע של ביצים בהטלה – numOfEggs –
ופעלת: .int getNumOfEggs()

א. סרטט את היררכיית המחלקות של גן החיות ואת הממשקים, אם צריך, באופן המתאים
ב尤טור לעקרונות של תכונות מונחה עצמים (הכמסה – encapsulation,
הירושה – inheritance, פולימורפיזם – polymorphism).
סמן בסרטוט את הקשרים בין המחלקות ובין הממשקים, אם
יש ממשקים.
אין צורך לככלו הסרטוטים תכונות ופעולות.

ב. לכל אחת מן המחלקות שבסרטוט כתוב ב- Java את כותרת המחלקה ואת התכונות שלה,
וכותרות לכל הפעולות.

לכל אחד מן הממשקים שבסרטוט כתוב ב- Java את הכותרת שלו, וכותרות לכל
הפעולות.
אין צורך לכתוב את הפעולות get ו- set ופעולות בונות.
בכל המחלקות הוגדרה הפעולה String details()

פעלה זו מחזירה מחזורת המורכבת שם החיה, כאשר לפניהם שם החיה מצוינים שמות
המחלקות בהיררכיית הירושה של החיה החל מ- Animal.
לדוגמא:

בעבור העצם In מטיפוס Lion ששמו Simba, הפעלה details מחזירה את המחרוזת
(משמעות לימין):

Animal , Mammal , Lion , Simba

בעבור העצם קְלָם מטיפוס Platypus ששמו Perry, הפעלה details מחזירה את
המחרוזת (משמעות לימין):

Animal , Mammal , Platypus , Perry

ממש ב- Java את הפעולה details במחלקות **Platypus**, **Lion**, **Mammal**, **Animal** ו- **details**. ממש את הפעולה באמצעות המאפיינים המתאימים ביטר לערךונות של כדי שיווחרו המחרוזות המתוארות. ממש את הפעולה באופן המתאים ביטר לעקרונות של תכונות מונחה עצמים (הכמסה — encapsulation, הורשה — inheritance, פולימורפיזם — polymorphism).

הפעולה `toString` קוראת לפעולה `details`. אם החיה מטילה ביצים הפעולה `toString` מוסיפה למחרוזת את המילים Number of eggs ואת מספר הביצים המופיע בהטללה, ומהזירה את המחרוזת.

ממש ב- Java את הפעולה `toString` בכל מחלוקת שהיא מופיעה בה, באופן המתאים ביטר לעקרונות של תכונות מונחה עצמים (הכמסה — encapsulation, הורשה — inheritance — polymorphism).

/המשך בעמוד 34/

.Tester המחלקות **Bill** , **PhoneBill** , **Shopping** , **MyBills** , ופועלה ראשית במחלקה **.19**
 עקוב אחר ביצוע הפועלה הראשית, וכותב את הפלט המתkeletal.
 במקבב יש לכלול את ערכי המשתנים, ובverb כל עצם את ערכי התוכנות שלו.

```
public class Bill {
    public final static char DOLLAR = 'd';
    public final static char SHEKEL = 's';
    private static double dollarRate = 4.00;
    private double total;
    private char curr;
    public double getTotal() {
        if (curr == DOLLAR)
            return total * dollarRate;
        return total;
    }
    public Bill(double total, char curr) {
        this.total = total;
        this.curr = curr;
    }
    public String toString() {
        String s = getTotal() + " Shekels";
        return s;
    }
}
```

```
public class PhoneBill extends Bill {
    private String company;
    public PhoneBill(String company, double value, char curr) {
        super(value, curr);
        this.company = company;
    }
    public PhoneBill(PhoneBill pb) {
        super(pb.getTotal(), SHEKEL);
        company = pb.company;
    }
    public String toString() {
        String s = "Pay: " + super.toString() + " to: " + company;
        return s;
    }
}
```

```

public class Shopping extends Bill {
    private String name;
    private String list;
    public Shopping(String name, String list, double value, char curr) {
        super(value, curr);
        this.name = name;
        this.list = list;
    }
    public Shopping(Shopping s) {
        super(s.getTotal(), SHEKEL);
        name = s.name;
        list = s.list;
    }
    public String toString() {
        String s = name + " bought: " + list + ". Pay: " + super.toString();
        return s;
    }
}

```

```

public class MyBills {
    private int num;
    private Bill[] bills;
    private final static int MAX = 4;
    public MyBills() {
        bills = new Bill[MAX];
        num = 0;
    }
    public void addBill(Bill b) {
        if (num < MAX) {
            if (b instanceof PhoneBill)
                bills[num] = new PhoneBill((PhoneBill) b);
            else if (b instanceof Shopping)
                bills[num] = new Shopping((Shopping) b);
            else num--;
            num++;
        }
    }
}

```

(שים לב: המשך השאלה בעמוד הבא.)
ההמשך בעמוד הבא./36

```
public double total() {  
    double t = 0;  
    for (int i = 0; i < num; i++)  
        t = t + bills[i].getTotal();  
    return t;  
}  
public void print() {  
    for (int i = 0; i < num; i++)  
        System.out.println(bills[i].toString());  
}  
}  
  
public class Tester {  
    public static void main(String[] args) {  
        MyBills baba = new MyBills();  
        Shopping s = new Shopping("Baba", "Books", 250, Bill.DOLLAR);  
        PhoneBill t = new PhoneBill("TeleBaba", 2000, Bill.SHEKEL);  
        PhoneBill u = new PhoneBill("Galil Telecom", 200, Bill.DOLLAR);  
        baba.addBill(t);  
        baba.addBill(s);  
        baba.addBill(u);  
        baba.print();  
        System.out.println(baba.total());  
    }  
}
```

המשך בעמוד /37

שלושת הטעיפים א-ג מתייחסים למחלקות האלה, אך איןם 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+";"; }  
}
```

(**שים לב: סעיפי השאלה בעמודים הבאים.**)

א. לפני סדרת הוראות שהפלט בעבורה צריך להיות:

```
xA=1 xB=20; xA=1 xB=20; xA=1 xB=20; xA=1 xB=20;
```

בסדרת הוראות 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. לפני סדרת הוראות. הצג את העצמים הנוצרים, ולכל עצם הצג את ערכיו התוכנויות 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();
```

ג. לפניה פעולה ראשית.

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

לפני 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 // פעולה הראשית. —

קבע אם הקטע תקין או אינו תקין. —

אם הקטע תקין — כתוב את הפלט המתקיים 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 שתיים מהשאלות 21-24.
(כל שאלה – 25 נקודות)

.21. בשאלת זו שני סעיפים א-ב שאין קשר ביניהם. ענה על שניהם.

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

public virtual void Foo(){} הוגדרה הפעולה (1)
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 יירושות את המחלקה B .

בכל אחת מהמחלקות X , Y , Z הוגדרה הפעולה
public virtual void Foo() {} הוגדרה הפעולה
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 יירושות את המחלקה B .

בכל אחת מהמחלקות X , Y , Z הוגדרה הפעולה
public virtual void Foo(){} הוגדרה הפעולה
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 יירושות את המחלקה B .

בכל אחת מהמחלקות B , Y , X , 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.

.ב. אין קשר לסעיף א.)

לפניך המחלקה **Singer** שיורשת את המחלקה **Artist** והמשק **IPerform** שהמחלקה **Singer** ממשחת.

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 כוורת המחלקה **Artist**, את התכונות ואת הפעולות המותחיבות מהמחלקה

.**Artist** והמשק **IPerform** הנתונים. אין צורך לממש את הפעולות של המחלקה **Singer**

אין לשנות את המחלקה **Singer** ואת המשק **IPerform**

בגנ חיות נמצאות החיות האלה: אריות Lion, ברווזנים Platypus, נחשים Snake וצבים Turtle.

אריות וברווזנים הם יונקים, נחשים וצבים הם זוחלים.

הנקבות של כל הזוחלים בגן החיות מטילות ביצים, וגם נקבת הברווזן מטילה ביצים.

רוצים לבנות מערכת ממוחשבת שתציג את החיים בגן החיות. המערכת תראה תמונות וanimatzia.

לשם כך הוגדרו המחלקות Animal – חייה, Mammal – יונק, Reptile – זוחל.

לכל חייה יש תכונה: שם – name, ופעולות: void Turn() ו- void Move().

לחיות שטילות ביצים יש התכונה: מספר ממוצע של ביצים בהטלה – numOfEggs –

ופעלת: .int GetNumberOfEggs()

א. סרטט את היררכיית המחלקות של גן החיות ואת הממשקים, אם צריך, באופן המתאים ביוטר לעקרונות של תכונות מונחה עצמים (הכמסה – encapsulation,

הירושה – inheritance, פולימורפיזם – polymorphism –).

סמן בסרטוט את הקשרים בין המחלקות ובין הממשקים, אם יש ממשקים.

אין צורך לכלול הסרטוטים תכונות ופעולות.

ב. לכל אחת מן המחלקות שבסרטוט כתוב ב- C# את כותרת המחלקה ואת התכונות שלה, וכותרות לכל הפעולות.

לכל אחד מן הממשקים שבסרטוט כתוב ב- C# את הכותרת שלו, וכותרות לכל הפעולות.

אין צורך לכתוב את הפעולות Get ו- Set ופעולות בונוט. string Details()

בכל המחלקות הוגדרה הפעולה

.Animal_details()

פעלה זו מחזירה מחרוזת המורכבת שם החיה, כאשר לפניה שם החיה מצוינים שמות המחלקות בהיררכיית הירושה של החיה החל מ-

לדוגמה:

בעבור העצם In מטיפוס Lion ששמו Simba, הפעולה Details מחזירה את המחרוזת (משמאלי לימי):

Animal , Mammal , Lion , Simba

בעבור העצם קלאס מטיפוס Platypus ששמו Perry, הפעולה Details מחזירה את המחרוזת (משמאלי לימי):

Animal , Mammal , Platypus , Perry

מימוש ב- C# את הפעולה Details במחלקות **Platypus**, **Lion**, **Mammal**, **Animal** ו- **Rabbit**. מימוש את הפעולה באופן המתאים ביותר לעקרונות של כדי שיווחזו המחרוזות המתוארכות. מימוש את הפעולה באופן המתאים ביותר לעברון לעקרונות של תכונות מונחה עצמים (הכמסה — encapsulation, הורשה — inheritance, ומחזירה את המחרוזות. — polymorphism).

2. הפעולה ToString קוראת לפעולה Details. אם החיה מטילה ביצים הפעולה ToString מוסיפה למחרוזת את המילים Number of eggs ואת מספר הביצים הממוצע בהטלה, וממחזירה את המחרוזת.

מימוש ב- C# את הפעולה ToString בכל מחלוקת שהיא מופיעה בה, באופן המתאים ביותר לעקרונות של תכונות מונחה עצמים (הכמסה — encapsulation, הורשה — inheritance, ומחזירה את המחרוזות. — polymorphism).

/המשך בעמוד 44/

.Tester המחלקות MyBills , Shopping , PhoneBill , Bill לפניך .
 עקוב אחר ביצוע הפעולה הראשית, וכתוב את הפלט המתkeletal.
 בעקבות ייש לכלול את ערכי המשתנים, ובעבור כל עצם את ערכי התוכנות שלו.

```
public class Bill {
```

```
    public const char DOLLAR = 'd';
    public const char SHEKEL = 's';
    private static double dollarRate = 4.00;
    private double total;
    private char curr;
    public double GetTotal() {
        if (curr == DOLLAR)
            return total * dollarRate;
        return total;
    }
    public Bill(double total, char curr) {
        this.total = total;
        this.curr = curr;
    }
    public override string ToString() {
        string s = GetTotal() + " Shekels";
        return s;
    }
}
```

```
public class PhoneBill : Bill {
```

```
    private string company;
    public PhoneBill(string company, double value, char curr) : base(value, curr) {
        this.company = company;
    }
    public PhoneBill(PhoneBill pb) : base(pb.GetTotal(), SHEKEL){
        company = pb.company;
    }
    public override string ToString() {
        string s = "Pay: " + base.ToString() + " to: " + company;
        return s;
    }
}
```

```

public class Shopping : Bill {
    private string name;
    private string list;
    public Shopping(string name, string list, double value, char curr) : base(value, curr){
        this.name = name;
        this.list = list;
    }
    public Shopping(Shopping s):base (s.GetTotal(), SHEKEL) {
        name = s.name;
        list = s.list;
    }
    public override string ToString() {
        string s = name + " bought: " + list + ". Pay: " + base.ToString();
        return s;
    }
}
public class MyBills {
    private int num;
    private Bill[] bills;
    private const int MAX = 4;
    public MyBills() {
        bills = new Bill[MAX];
        num = 0;
    }
    public void AddBill(Bill b) {
        if (num < MAX) {
            if (b is PhoneBill)
                bills[num] = new PhoneBill((PhoneBill) b);
            else if (b is Shopping)
                bills[num] = new Shopping((Shopping) b);
            else num--;
            num++;
        }
    }
}

```

(שים לב: המשך השאלה בעמוד הבא).

```
public double Total() {  
    double t = 0;  
    for (int i = 0; i < num; i++)  
        t = t + bills[i].GetTotal();  
    return t;  
}  
public void Print() {  
    for (int i = 0; i < num; i++)  
        Console.WriteLine(bills[i].ToString());  
}  
}  
  
public class Tester {  
    public static void Main(string[] args) {  
        MyBills baba = new MyBills();  
        Shopping s = new Shopping("Baba", "Books", 250, Bill.DOLLAR);  
        PhoneBill t = new PhoneBill("TeleBaba", 2000, Bill.SHEKEL);  
        PhoneBill u = new PhoneBill("Galil Telecom", 200, Bill.DOLLAR);  
        baba.AddBill(t);  
        baba.AddBill(s);  
        baba.AddBill(u);  
        baba.Print();  
        Console.WriteLine(baba.Total());  
    }  
}
```

.24 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.)

ההמשך בעמוד /48

א. לפניך סדרת הוראות שהפלט עבורה צריך להיות:
xA=1 xB=20; xA=1 xB=20; xA=1 xB=20; xA=1 xB=20;

בסדרת הוראות 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 ערכיו התיוכנות 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();
```

```

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 // פעולה 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 (קומpileציה) או שגיאת זמן ריצה.

ב hasilחה !