

שימו לב: בבחינה זו יש הנחיות מיוחדות.
יש לענות על השאלות על פי הנחיות אלה.

מדעי המחשב

הוראות

א. משך הבחינה: שלוש שעות.

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

פרק ראשון	-	$(10 \times 1) + (15 \times 1)$	-	25	נקודות
פרק שני	-	(25×2)	-	50	נקודות
פרק שלישי	-	(25×1)	-	25	נקודות
סך הכול	-		-	100	נקודות

ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ ממחשבון שיש בו אפשרות תכנות.

ד. הוראות מיוחדות:

(1) רשמו על הכריכה החיצונית של המחברת את שם המסלול שלמדתם.

המסלול הוא אחד מארבעת המסלולים האלה:

מערכות מחשב ואסמבלי, אלגוריתמים, מודלים חישוביים, תכנות מונחה עצמים.

(2) את כל התוכניות שיש לכתוב בשפת מחשב בפרקים הראשון והשני כתבו בשפה אחת בלבד –

Java או C#.

הערה: לא יורדו נקודות אם תכתבו בתוכניות אות גדולה במקום אות קטנה או להפך.

יש לכתוב במחברת הבחינה בלבד. יש לרשום "טיוטה" בראש כל עמוד המשמש טיוטה.

כתובת טיוטה בדפים שאינם במחברת הבחינה עלולה לגרום לפסילת הבחינה.

השאלות בשאלון זה מנוסחות בלשון רבים, אף על פי כן על כל תלמידה וכל תלמיד להשיב עליהן באופן אישי.

בהצלחה!

השאלות

בשאלון זה שלושה פרקים.

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

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

לפיתרים בשפת Java: בכל שאלה שנדרשת בה קליטה, הניחו שבתוכנית כתובה ההוראה:

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

פרק ראשון (25 נקודות)

ענו על שאלה 1 – חובה (10 נקודות).

1. כתבו פעולה חיצונית ששמה multiply בשפת Java או Multiply בשפת C# המקבלת שני מערכים מטיפוס שלם,

arr1 ו- arr2, ששוניים בגודלם. נקרא לגודל המערך הקטן יותר k ולגודל המערך הגדול יותר m. על הפעולה להחזיר

מערך חדש מטיפוס שלם בגודל m (גודל המערך הגדול יותר מבין המערכים arr1 ו- arr2).

על המערך המוחזר להיות לפי הפירוט שלהלן:

ערכו של כל אחד מ- k התאים הראשונים יהיה שווה למכפלה של הערכים בתאים המקבילים לו במערכים arr1 ו- arr2.

שאר ערכי התאים במערך המוחזר יהיו זהים לערכי התאים המקבילים להם במערך הגדול.

דוגמה: בעבור המערכים arr1 ו- arr2 שלהלן:

	0	1	2	3	4
arr1	1	-4	4	9	2

	0	1	2	3	4	5	6
arr2	9	2	0	-1	3	11	23

הפעולה תחזיר את המערך הזה:

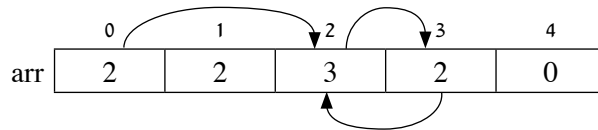
	0	1	2	3	4	5	6
	9	-8	0	-9	6	11	23

הערה: הניחו שאין תאים ריקים בשני המערכים, ושערכי המערכים תקינים.

ענו על אחת מן השאלות 2-3 (15 נקודות).

2. "סריקה לפי ערכי התאים" היא סריקת מערך המתחילה באינדקס 0 ועוברת לאינדקס שמספרו זהה לערך התא הנוכחי שנסרק (התא שבאינדקס 0), ומשם ממשיכה לתא הבא באותו אופן, וכן הלאה. אם אחד מערכי התאים הוא 0 והסריקה מגיעה אליו, היא מסתיימת, אחרת הסריקה אינה מסתיימת.

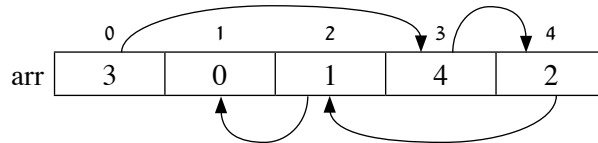
דוגמה:



סדר האינדקסים בסריקה זו הוא: 0, 2, 3, 2, 3, ... (יש לקרוא את המספרים משמאל לימין).
הניחו שערכי התאים אינם חורגים מגבולות המערך.

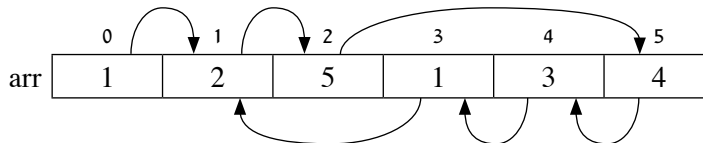
- מערך מושלם הוא מערך שבו ב"סריקה לפי ערכי התאים" מתקיימים גם התנאי הראשון וגם התנאי השני שלהלן:
1. כל התאים של המערך נסרקים.
 2. הסריקה מסתיימת (מפני שבמערך יש תא שערכו 0 והסריקה מגיעה אליו).

דוגמה למערך מושלם:



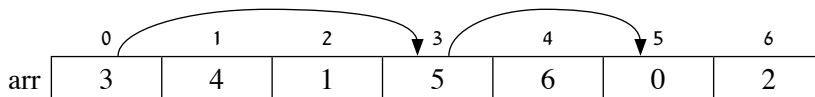
המערך בדוגמה זו הוא מערך מושלם מפני שהסריקה עוברת על כל התאים, ומסתיימת.

דוגמה למערך שאינו מושלם:



מערך זה אינו מערך מושלם מפני שהסריקה אינה מסתיימת.

דוגמה נוספת למערך שאינו מושלם:



מערך זה אינו מערך מושלם מפני שהסריקה מסתיימת בלי שנסרקו התאים באינדקסים 6, 4, 2, 1.

ממשו את הפעולה הנתונה:

Java – public static boolean isPerfect (int[] arr)

C# – public static bool IsPerfect (int[] arr)

הפעולה תחזיר true אם המערך arr שהתקבל הוא מערך מושלם, אחרת היא תחזיר false.
הערה: אין חובה לשמור על ערכי המערך.

/המשך בעמוד 4/

הניחו שערכי התאים אינם חורגים מגבולות המערך.

3.

פיקסל (Pixel) הוא יחידת מידע המתארת נקודה בתמונה דיגיטלית.

כל פיקסל מורכב משילוב של שלושה צבעים: אדום, ירוק וכחול.

כל אחד משלושת הצבעים האלה (אדום, ירוק וכחול) מקבל ערך מספרי מ-0 עד 255 (כולל). המספר מייצג את עוצמת הצבע בנקודה. באמצעות שילוב של שלושת הצבעים בערכים שונים מ-0 עד 255 אפשר לקבל כל גוון בקשת הצבעים.

לקבלת נקודה שהצבע שלה הוא אדום (בין אדום בהיר לאדום כהה), ערכי הצבעים צריכים להיות:

אדום – מ-1 עד 255, ירוק – 0, כחול – 0.

לקבלת נקודה שהצבע שלה הוא ירוק (בין ירוק בהיר לירוק כהה), ערכי הצבעים צריכים להיות:

אדום – 0, ירוק – מ-1 עד 255, כחול – 0.

לקבלת נקודה שהצבע שלה הוא כחול (בין כחול בהיר לכחול כהה), ערכי הצבעים צריכים להיות:

אדום – 0, ירוק – 0, כחול – מ-1 עד 255.

לקבלת נקודות בשאר הצבעים (לא כולל שחור ולבן), הערכים של לפחות שני צבעים מתוך הצבעים אדום, ירוק וכחול צריכים להיות גדולים מ-0.

לקבלת נקודה שהצבע שלה הוא שחור, ערכי הצבעים אדום, ירוק וכחול צריכים להיות 0.

לקבלת נקודה שהצבע שלה הוא לבן, ערכי הצבעים אדום, ירוק וכחול צריכים להיות 255.

נתונה המחלקה **Pixel** – פיקסל, ולה שלוש תכונות:

- red – צבע אדום, מספר שלם מ-0 עד 255
- green – צבע ירוק, מספר שלם מ-0 עד 255
- blue – צבע כחול, מספר שלם מ-0 עד 255

הניחו שיש פעולות `get/Get` ו-`set/Set` בעבור תכונות המחלקה.

נתון ממשק חלקי של המחלקה `Pixel`:

כותרת הפעולה	תיאור הפעולה
<code>public Pixel (int red, int green, int blue)</code>	פעולה בונה המקבלת ערכים בעבור תכונות המחלקה
Java – <code>public boolean isRed ()</code> C# – <code>public bool IsRed ()</code>	פעולה המחזירה <code>true</code> אם הנקודה היא בצבע אדום, אחרת היא מחזירה <code>false</code>
Java – <code>public boolean isGreen ()</code> C# – <code>public bool IsGreen ()</code>	פעולה המחזירה <code>true</code> אם הנקודה היא בצבע ירוק, אחרת היא מחזירה <code>false</code>
Java – <code>public boolean isBlue ()</code> C# – <code>public bool IsBlue ()</code>	פעולה המחזירה <code>true</code> אם הנקודה היא בצבע כחול, אחרת היא מחזירה <code>false</code>

(שימו לב: המשך השאלה בעמוד הבא.)

- א. ממשו את הפעולה `isRed/IsRed`.
 הניחו שערכי התכונות תקינים (מ-0 עד 255).
- ב. בסעיף זה אפשר להשתמש בפעולות `isGreen / IsGreen` ו- `isBlue / IsBlue` בלי לממש אותן. אפשר להוסיף פעולות נוספות למחלקה `Pixel`, אך חובה לממש אותן.
 נתונה המחלקה **Structure** – ולה תכונה אחת בלבד:
`arr` – מערך חד-ממדי מטיפוס **Pixel** בגודל `N`.
 (1) כתבו פעולה פנימית במחלקה `Structure` ששמה `isBalanced` בשפת `Java` או `IsBalanced` בשפת `C#`. הפעולה תחזיר `true` אם במערך `arr` מספר הנקודות בצבע האדום זהה למספר הנקודות בצבע הירוק וזהה למספר הנקודות בצבע הכחול. אחרת הפעולה תחזיר `false`.
שימו לב: הפעולה תחזיר `true` גם אם יש נקודות בצבעים שאינם אדום, ירוק וכחול, כל עוד מספרי הנקודות האדומות, הירוקות והכחולות זהים.
 (2) מערך "שחור לבן" הוא מערך שבו כל הנקודות הן רק בצבעים שחור ולבן, ויש בו לפחות נקודה אחת בצבע לבן ולפחות נקודה אחת בצבע שחור.
 כתבו פעולה פנימית במחלקה `Structure` ששמה `isBlackWhite` בשפת `Java` או `IsBlackWhite` בשפת `C#`. הפעולה תחזיר `true` אם המערך `arr` הוא מערך "שחור לבן", אחרת היא תחזיר `false`.

פרק שני (50 נקודות)

שימו לב: בכל שאלה שנדרש בה מימוש אפשר להשתמש בפעולות של המחלקות: תור, מחסנית, עץ בינרי וחוליה, בלי לממש אותן. אם משתמשים בפעולות נוספות, יש לממש אותן.

ענו על שתיים מן השאלות 4-7 (לכל שאלה - 25 נקודות).

4. נתונה המחלקה **Range** - טווח, ולה שתי תכונות:

- low - מספר מטיפוס שלם
- high - מספר מטיפוס שלם

המספר high גדול או שווה ל- low ($high \geq low$).

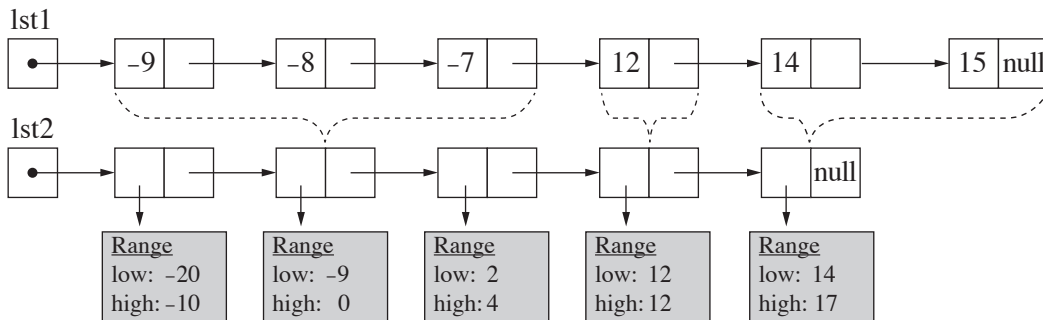
הניחו שיש פעולות get/Set ו- set/Set בעבור תכונות המחלקה.

מספר כלשהו, x, "מוכל" בעצם מטיפוס Range אם הוא נמצא בטווח המספרים שבין low ובין high

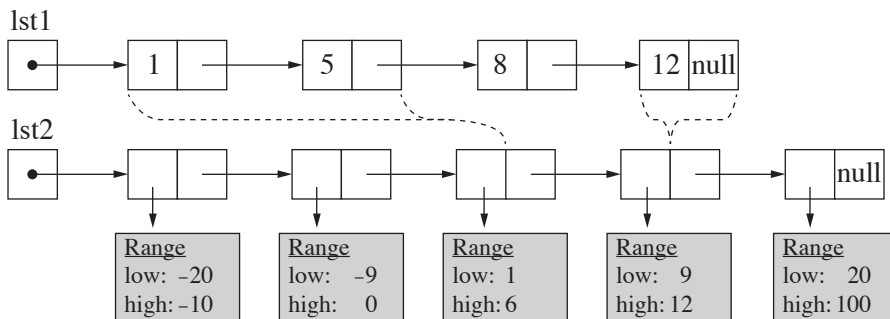
($high \geq x \geq low$).

שרשרת חוליות - lst1 מטיפוס שלם "מוכלת" בשרשרת חוליות - lst2 מטיפוס Range אם בעבור כל מספר בשרשרת lst1 קיימת חוליה בשרשרת lst2 המכילה אותו.

דוגמה לשרשרת lst1 המוכלת בשרשרת lst2:



דוגמה לשרשרת lst1 שאינה מוכלת בשרשרת lst2:



הסבב: המספר 8 שבשרשרת lst1 אינו "מוכל" בשום חוליה בשרשרת lst2.

(שימו לב: המשך השאלה בעמוד הבא.)

ממשו את הפעולה החיצונית שלהלן:

Java – public static boolean isIncluded (Node<Integer> lst1, Node<Range> lst2)

C# – public static bool IsIncluded (Node<int> lst1, Node<Range> lst2)

הפעולה מחזירה true אם lst1 "מוכלת" ב-lst2, אחרת היא מחזירה false. הפעולה חייבת לעבוד בסיבוכיות זמן ריצה של $O(N)$.

הערה: N הוא אורך השרשרת הארוכה יותר מבין שתי השרשראות.

הנחות:

- lst1 ו-lst2 אינם null.
- בשרשרת lst2 כל העצמים מטיפוס Range אינם null.
- **השרשרת lst1 ממוינת בסדר עולה.**
- **השרשרת lst2 ממוינת בסדר עולה**, כלומר, ערך ה-high של כל חוליה קטן מערך ה-low של החוליה הבאה אחריה בשרשרת (כפי שמופיע בדוגמאות בעמוד הקודם).

5. נתונה המחלקה **TwoStack**, ולה שתי תכונות:

- numbers – מחסנית מטיפוס שלם
- sums – מחסנית מטיפוס שלם

היחס בין המחסנית sums למחסנית numbers הוא כמפורט להלן:
 המספר בסוף המחסנית sums שווה למספר בסוף המחסנית numbers,
 המספר השני מסוף המחסנית sums שווה לסכום שני המספרים האחרונים במחסנית numbers,
 המספר השלישי מסוף המחסנית sums שווה לסכום שלושת המספרים האחרונים במחסנית numbers,
 וכן הלאה עד המספר בראש המחסנית sums, השווה לסכום כל המספרים במחסנית numbers.

דוגמה:



א. ממשו את הפעולה הפנימית שלהלן:

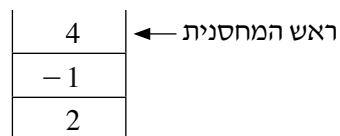
Java – `public Stack<Integer> getNums (int x)`

C# – `public Stack<int> GetNums (int x)`

הפעולה מקבלת מספר x השווה לאחד המספרים במחסנית sums, ומחזירה מחסנית חדשה מטיפוס שלם, שבה מופיעים המספרים מן המחסנית numbers שסכומם שווה למספר x .
 הניחו שהמספר x קיים במחסנית sums ומופיע בה רק פעם אחת.

דוגמה:

בעבור המחסניות שבדוגמה לעיל ו- $x = 5$ (הוא המספר השלישי מלמטה במחסנית sums), הפעולה תחזיר את המחסנית שלהלן:



הערות: – אפשר לשנות את המחסניות של המחלקה.

– אין חשיבות לסדר המספרים במחסנית המוחזרת.

– ההגדרה למחסנית מטיפוס שלם `Stack<int>/Stack<Integer>` שקולה להגדרה `StackInt`.

אפשר להשתמש בכל אחת מהן, לבחירתכם.

(שימו לב: המשך השאלה בעמוד הבא.)

ב. ממשו את הפעולה הפנימית שלהלן:

Java – public void eraseNum (int x)

C# – public void EraseNum (int x)

הפעולה מוחקת את המספר x מן המחסנית numbers ומתקנת את המחסנית sums בהתאם. הניחו שהמספר x קיים במחסנית numbers והוא מופיע בה רק פעם אחת. הערה: יש לשמור על סדר המספרים שנשארו במחסנית numbers.

דוגמה:

בעבור המחסניות שלפניכם ו- $x = 3$:

<u>sums</u>		<u>numbers</u>
8	← ראש המחסנית	6
2		- 3
5		3
2		2

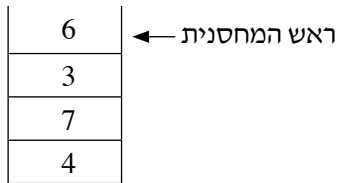
לאחר הרצת הפעולה ייראו המחסניות כך:

<u>sums</u>		<u>numbers</u>
5	← ראש המחסנית	6
- 1		- 3
2		2

6. שימו לב! לשאלה זו שני נוסחים: בשפת Java בעמודים 10–11 ובשפת C# בעמודים 12–13.

לפתרים בשפת Java

א. נתונה הפעולה stackSod1 והמחסנית st :



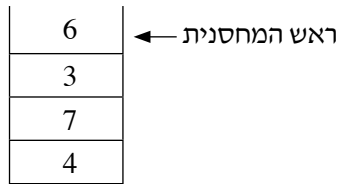
```
public static void stackSod1(Stack<Integer> st, int element)
{
    if(st.isEmpty())
        st.push(element);
    else
    {
        int val = st.pop();
        stackSod1(st, element);
        st.push(val);
    }
}
```

הערה: ההגדרה למחסנית מטיפוס שלם Stack<Integer> שקולה להגדרה StackInt.

- (1) סרטטו את המחסנית כפי שהיא תיראה לאחר זימון הפעולה stackSod1(st, 9). יש להראות מעקב.
- (2) מהי מטרת הפעולה stackSod1?
- (3) מהי סיבוכיות זמן הריצה של הפעולה stackSod1?

(שימו לב: המשך השאלה בעמוד הבא.)

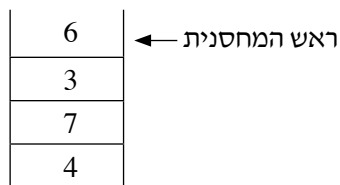
ב. נתונה הפעולה stackSod2 והמחסנית st :



```
public static void stackSod2 (Stack<Integer> st)
{
    if(!st.isEmpty())
    {
        int val = st.pop();
        stackSod2(st);
        stackSod1(st, val);
        st.push(val);
    }
}
```

- (1) סרטטו את המחסנית כפי שהיא תיראה לאחר זימון הפעולה stackSod2(st). יש להראות מעקב.
הערה: בסעיף זה אין צורך לבצע מעקב אחרי הפעולה stackSod1.
- (2) מהי מטרת הפעולה stackSod2 ?
- (3) מהי סיבוכיות זמן הריצה של הפעולה stackSod2 ?

א. נתונה הפעולה StackSod1 והמחסנית st :



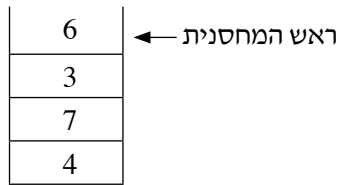
```
public static void StackSod1(Stack<int> st, int element)
{
    if(st.IsEmpty())
        st.Push(element);
    else
    {
        int val = st.Pop();
        StackSod1(st, element);
        st.Push(val);
    }
}
```

הערה: ההגדרה למחסנית מטיפוס שלם Stack<int> שקולה להגדרה StackInt.

- (1) סרטטו את המחסנית כפי שתיראה לאחר זימון הפעולה StackSod1(st, 9). יש להראות מעקב.
- (2) מהי מטרת הפעולה StackSod1 ?
- (3) מהי סיבוכיות זמן הריצה של הפעולה StackSod1 ?

(שימו לב: המשך השאלה בעמוד הבא.)

ג. נתונה הפעולה StackSod2 והמחסנית st :



```
public static void StackSod2 (Stack<int> st)
{
    if(!st.IsEmpty())
    {
        int val = st.Pop();
        StackSod2(st);
        StackSod1(st, val);
        st.Push(val);
    }
}
```

(1) סרטטו את המחסנית כפי שתיראה לאחר זימון הפעולה StackSod2(st). יש להראות מעקב.

הערה: בסעיף זה אין צורך לבצע מעקב אחרי הפעולה StackSod1.

(2) מהי מטרת הפעולה StackSod2 ?

(3) מהי סיבוכיות זמן הריצה של הפעולה StackSod2 ?

7. שאלה בנושא עץ בינארי

בשאלה זו אפשר להשתמש בפעולה החיצונית eraseFirst / EraseFirst שלהלן בלי לממש אותה.

דוגמאות	תיאור הפעולה	כותרת הפעולה
- בעבור המחרוזת str = "hello", הפעולה תחזיר את המחרוזת "ello". - בעבור המחרוזת str = "temp", הפעולה תחזיר את המחרוזת "emp". - בעבור המחרוזת str = "m", הפעולה תחזיר מחרוזת ריקה "". - בעבור המחרוזת הריקה str = "", תהיה שגיאה.	הפעולה מחזירה תת-מחרוזת של str, ללא התו הראשון. אם המחרוזת - str ריקה לפני זימון הפעולה, תהיה שגיאה.	בשפת Java: public static String eraseFirst (String str) בשפת C#: public static string EraseFirst (string str)

ממשו את הפעולה החיצונית שלהלן:

Java – public static boolean wordFromRoot (BinNode<Character> tree, String str)

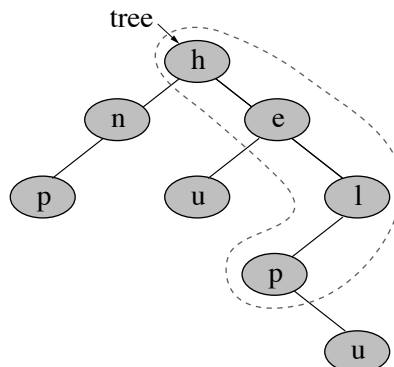
C# – public static bool WordFromRoot (BinNode<char> tree, string str)

הפעולה מקבלת מחרוזת str המכילה לפחות תו אחד, והפניה לעץ בינארי של תווים - tree שאינו null. הפעולה תחזיר true אם קיים מסלול המתחיל בשורש העץ שבו בצף התווים זהה למחרוזת str. אחרת הפעולה תחזיר false.

הערה: אות קטנה ואות גדולה אינן זהות זו לזו.

דוגמה:

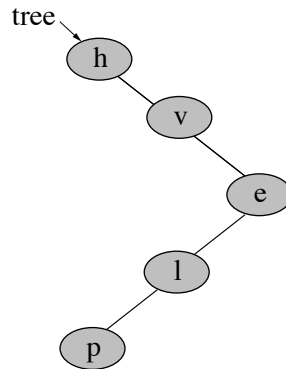
בעבור העץ הנתון והמחרוזת "help" הפעולה תחזיר true.



(שימו לב: המשך השאלה בעמוד הבא.)

דוגמה:

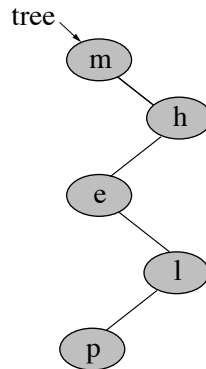
בעבור העץ הנתון והמחרוזת "help" הפעולה תחזיר false.



הסבר: לא קיים בעץ רצף תווים הזהה למחרוזת "help".

דוגמה:

בעבור העץ הנתון והמחרוזת "help" הפעולה תחזיר false.



הסבר: אף על פי שקיים בעץ רצף תווים הזהה למחרוזת "help", הפעולה תחזיר false, כי הרצף אינו מתחיל בשורש העץ.

פרק שלישי (25 נקודות)

בפרק זה שאלות בארבעה מסלולים:

מערכות מחשב ואסמבלי, עמודים 16-18.

אלגוריתמים, עמודים 19-22.

מודלים חישוביים, עמוד 23.

תכנות מונחה עצמים בשפת Java, עמודים 24-27; תכנות מונחה עצמים בשפת C#, עמודים 28-31.

יש לענות על שאלה אחת במסלול שלמדתם.

מערכות מחשב ואסמבלי

אם למדתם מסלול זה, יש לענות על אחת מן השאלות 8-9 (25 נקודות).

8. בשאלה זו שני סעיפים, א-ב, שאין ביניהם קשר. יש לענות על שני הסעיפים.

א. נתון קטע תוכנית בשפת אסמבלי.

```

MOV SI, 30H
MOV CL, 4
MOV CH, 4
MOV BH, 0
A1: MOV AX, [SI]
MOV BL, AL
ROR BL, CL
DEC BL
MOV [SI], BX
ADD SI, 2
DEC CH
CMP CH, 0
JNE A1
    
```

נתונה מפת תאי הזיכרון מ־30H עד 36H לפני ביצוע קטע התוכנית:

address	30H	32H	34H	36H
value	25H	3AH	7BH	93H

שימו לב: תאי הזיכרון הם בגודל מילה.

(1) עקבו בעזרת טבלת מעקב אחר קטע התוכנית.

בטבלת המעקב יש לכלול עמודה לכל אחד מן האוגרים SI, CL, CH, BL, BH, AX.

נוסף על כך, יש לכתוב מפת זיכרון מתאימה כפי שתיראה בסוף קטע התוכנית.

(2) הסבירו מה מבצע קטע התוכנית.

(שימו לב: המשך השאלה בעמוד הבא.)

ב. להלן קטע תוכנית הכתוב בשפות Java ו- C#.

```
if ( x < y + z )
{
    do {
        if ( y >= z || x < y )
            x += y;
        z --;
    } while ( z > 0 );
}
```

המשתנים x , y ו- z הם מטיפוס שלם.

כתבו קטע מתאים בשפת אסמבלי.

הניחו שהמשתנים x , y ו- z מכוונים (signed) ומאוחסנים באוגרים AX, BX ו- CX בהתאמה.

הערה: אין צורך להתייחס לבעיות גלישה אפשריות.

9.

"זוגות משני הצדדים" הם זוגות איברים במערך בגודל זוגי, כמפורט להלן:

הזוג הראשון מורכב מן האיבר הראשון ומן האיבר האחרון במערך,
 הזוג השני מורכב מן האיבר השני ומן האיבר שלפני האיבר האחרון במערך,
 הזוג השלישי מורכב מן האיבר השלישי ומן האיבר שלפני שני האיברים האחרונים במערך,
 וכן הלאה עד זוג האחרון, המורכב משני האיברים שנמצאים באמצע המערך.

"מערך זוגי יורד" הוא מערך בגודל זוגי המקיים את התנאי שלהלן:

סכום ה"זוגות משני הצדדים" מסודר בסדר יורד, כלומר, סכום הזוג הראשון גדול מסכום הזוג השני, סכום הזוג השני גדול מסכום הזוג השלישי, וכן הלאה, עד הזוג האחרון, שסכומו הוא הקטן ביותר.

דוגמה למערך זוגי יורד:

0	1	2	3	4	5	6	7
8	6	7	1	2	2	5	5

הסבר: המערך הוא "זוגי יורד" מפני שסכום הזוג הראשון $(8+5)$ גדול מסכום הזוג השני $(6+5)$, הגדול מסכום הזוג השלישי $(7+2)$, הגדול מסכום הזוג האחרון $(1+2)$.

דוגמה למערך שאינו זוגי יורד:

0	1	2	3	4	5
6	8	1	2	3	5

הסבר: המערך אינו "זוגי יורד" מפני שסכום הזוג הראשון $(6+5)$ אינו גדול מסכום הזוג השני $(8+3)$, אלא שווה לו.

במקטע הנתונים הוגדרו הנתונים האלה:

ARR DB 20 DUP (?)

REZ DB ?

כתבו קטע תוכנית הבודקת אם מערך ARR הוא מערך זוגי יורד.

אם כן, התוכנית מאחסנת את המספר 1 במשתנה REZ, אחרת היא מאחסנת בו את המספר 0.

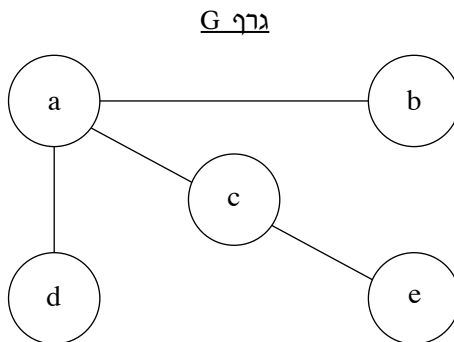
הערה: אין צורך לבדוק שהמערך בגודל זוגי.

אלגוריתמים

אם למדתם מסלול זה, יש לענות על אחת מן השאלות 10–11 (25 נקודות).

10. בשאלה זו שני סעיפים, א–ב, שאין ביניהם קשר. יש לענות על שני הסעיפים.

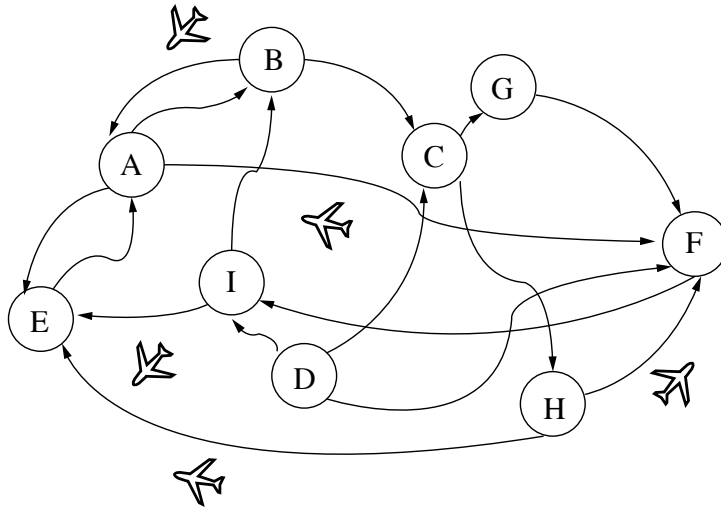
א. להלן גרף $G = (V, E)$ שאינו מכוון:



- (1) האם הגרף קשיר? נמקו.
- (2) האם הגרף דו־צדדי? נמקו.
- (3) האם הגרף מלא (שלם)? נמקו.
- (4) האם הגרף הוא עץ? נמקו.

(שימו לב: המשך השאלה בעמוד הבא.)

ב. חברת תעופה מפעילה בשגרה קווי תעופה בין המדינות A, B, C, D, E, F, G, H, I על פי הסרטוט שלהלן:



(1) סרטטו את רשימת הסמיכות המייצגת את קווי התעופה בין המדינות.

בעקבות שינויים בחברת התעופה הוחלט לבטל כמה מקווי התעופה במדינות שלהלן:

מדינה G – ביטול הטיסות הנכנסות והטיסות היוצאות

מדינה H – ביטול הטיסות היוצאות למדינה F

מדינה D – ביטול הטיסות היוצאות למדינה C

מדינה A – ביטול הטיסות היוצאות למדינה E

(2) סרטטו את רשימת הסמיכות המתקבלת לאחר הביטולים המצוינים לעיל.

(3) I. אביטל צריכה להגיע ממדינה מסוימת למדינה אחרת.

לשם כך היא רוצה לבדוק אם היא יכולה להגיע לאותה מדינה, ואם היא אכן יכולה, היא רוצה למצוא

מסלול טיסה כלשהו המגיע לאותה מדינה.

מהו האלגוריתם שעליה להשתמש בו למטרות אלה? הסבירו.

הערה: אם יש כמה אלגוריתמים מתאימים, בחרו באחד מהם.

II. הפעילו את האלגוריתם שכתבתם עליו בסעיף I(3) כדי למצוא את מסלול הטיסה ממדינה A

למדינה E. כתבו את מסלול הטיסה שמצאתם.

יש לבצע מעקב מפורט בכל שלב בהתאם לאלגוריתם שהפעלתם.

11. בשאלה זו ארבעה סעיפים, א-ד, שאין ביניהם קשר. יש לענות על כל הסעיפים.
- א. המסר: $It's\ my\ bank\ account$ הומר לסיביות (ביטים), חוץ מן המילה my . המירו את המילה my לסיביות. הערך ה- $ASCII$ של האות m הוא 109 בבסיס עשרוני, והערך ה- $ASCII$ של האות y הוא 121 בבסיס עשרוני.
- ב. להלן אוסף סיביות m , המייצג את הערך ה- $ASCII$ של המילה Hey , ואוסף סיביות key , שהוא פנקס חד-פעמי (OTP).
- $$m - 010010000110010101111001$$
- $$key - 110110100100010111101101$$
- הצפינו את המילה Hey באמצעות הפעולה xor עם key .
- ג. רונית ויובל הצפינו מסר כלשהו m , שאורכו n הוא זוגי, באמצעות הפעולה xor עם מפתח key , וקיבלו מסר מוצפן $cipher$. למען הזהירות, הם חצו את key לשניים. את החצי הראשון של המפתח, key_1 (שהוא באורך $n/2$), לקחה רונית איתה למקום אחד, ואת החצי האחר key_2 (שהוא באורך $n/2$), לקח איתו יובל למקום אחר.
- כדי לפענח את המסר, ביצעה חברתם הלית את הפעולות האלה לפי הסדר:
- היא לקחה מיובל את key_2 , והוסיפה לתחילתו כמות של אפסים באורך של $n/2$ (וכך נהיה אורכו של key_2 באורך n).
 - היא ביצעה $cipher\ xor\ key_2$ וקיבלה אוסף סיביות. נקרא לאוסף שהיא קיבלה $halfMessage$.
 - היא לקחה מרונית את key_1 , והוסיפה לסופו כמות של אפסים באורך של $n/2$ (וכך נהיה אורכו של key_1 באורך n).
 - היא ביצעה $halfMessage\ xor\ key_1$ וקיבלה אוסף סיביות. נקרא לאוסף שהיא קיבלה $fullMessage$.
- כתבו אם הפענוח תקין (כלומר, אם $fullMessage$ הוא המסר m). נמקו.
- הערה: תשובה ללא נימוק לא תזוכה בנקודות.

(שימו לב: המשך השאלה בעמוד הבא.)

7. בוב ואליס השתמשו בפעולה CreatePrg/createPrg כדי לייצר מפתח PRG, ואיתו הצפינו מסר באמצעות הפעולה xor. הפעולה מקבלת מחרוזת של סיביות - seed, ואת אורך המסר שרוצים להצפין - len, ומחזירה פנקס חד-פעמי (OTP).

הפעולה CreatePrg/createPrg כתובה לפניכם בשפות Java ו-C#. איב יודעת מהי הסיבית האחרונה של ה-seed. האם היא יכולה לפצח את המסר בניסיון אחד? נמקו.

הערה: תשובה ללא נימוק לא תזוכה בנקודות.

בשפת C# :	בשפת Java :
<pre>public static string CreatePrg (string seed, int len) { string str; if (seed[seed.Length - 1] == '1') str="0"; else str="1"; for (int i = 1; i < len; i ++) { if (str[str.Length - 1] == '0') str += "1"; else str += "0"; } return str; }</pre>	<pre>public static String createPrg (String seed, int len) { String str; if (seed.charAt(seed.length() - 1) == '1') str="0"; else str="1"; for (int i = 1; i < len; i ++) { if (str.charAt(str.length() - 1) == '0') str += "1"; else str += "0"; } return str; }</pre>

מודלים חישוביים

אם למדתם מסלול זה, יש לענות על אחת מן השאלות 12–13 (25 נקודות).

12. נתונה השפה L מעל הא"ב $\{a, b\}$:

$$L = \{a^n b^m \mid n > 0, m \geq 0, m \% 4 = n \% 2\}$$

א. לפניכם 10 מילים. בעבור כל אחת מהן, ציינו אם המילה שייכת לשפה L , ונמקו.

a, ab, aaabbb, aaabbbb, aabb, aaab, aa, bbbb, aaaa, abbb

ב. בנו אוטומט דטרמיניסטי מלא המקבל את השפה L .

13. נתונה השפה L_1 מעל הא"ב $\{a, b, c\}$:

$$L_1 = \{c^{1+k+n} b^k a^{2n} \mid n, k \geq 1\}$$

א. (1) מהי המילה הקצרה ביותר בשפה L_1 ?

(2) בְּנו אוטומט מחסנית המקבל את השפה L_1 .

ב. נגדיר את השפה L_2 מעל הא"ב $\{a, b, c, d\}$:

$$L_2 = L_1 \cdot d \cdot R(L_1)$$

בעבור כל אחת מן המילים שלפניכם, ציינו אם המילה שייכת לשפה L_2 , ונמקו.

cccbaaddaabccc

cccbaadcccbaa

cccbaadaaaabccc

הערה: אין צורך לבנות אוטומט בעבור השפה L_2 .

תכנות מונחה עצמים בשפת Java

אם למדתם מסלול זה ואתם כותבים בשפת Java, יש לענות על אחת מן השאלות 14–15 (25 נקודות).

14. בעל חנות כולבו מעוניין למכור את מוצריו גם באופן מקוון (אונליין). בשלב הראשון החליט בעל החנות למכור באופן

מקוון רק חולצות ומכנסיים.

לשם כך נבנו המחלקות האלה:

– המחלקה **Clothing**, המייצגת בגד

– המחלקה **Shirt**, המייצגת חולצה

– המחלקה **Pants**, המייצגת זוג מכנסיים

```
public class Clothing {
    private int id;           // מספר זיהוי של המוצר
    private String fabric;    // סוג הבד
    private String color;    // צבע הבגד
    private double price;    // מחיר
}

public class Shirt extends Clothing {
    private String size;     // מידה (S, M, L, XL, XXL)
}

public class Pants extends Clothing {
    private int length;     // אורך באינצ'ים
    private int width;     // רוחב באינצ'ים
}
```

הניחו שיש פעולות get ו־ set בעבור התכונות בכל המחלקות.

א. המחלקה **Cart** מייצגת סל קניות. למחלקה שלוש תכונות:

name – שם הקונה

arr – מערך המכיל גם חולצות וגם מכנסיים

current – מספר המוצרים הנוכחי בקנייה

(1) כתבו את כותרת המחלקה Cart ואת התכונות שלה.

(2) סרטטו תרשים הייררכייה המתאר את הקשר בין המחלקות: Clothing, Shirts, Pants, Cart.



(שימו לב: המשך השאלה בעמוד הבא.)

ב. בשלב השני החליט בעל החנות למכור באופן מקוון גם ספרים.

לפניכם המחלקה **Book** המייצגת ספר:

```
public class Book {
    private int id;           // מספר זיהוי של המוצר
    private String bookName; // שם הספר
    private String author;   // שם הסופר
    private double price;    // מחיר
}
```



הניחו שיש פעולות `get` ו-`set` בעבור תכונות המחלקה.

כדי שהמערך `arr` במחלקה `Cart` יוכל להכיל גם ספרים, נעשו הפעולות האלה:

- הוגדרה המחלקה **Product** - מוצר.
- טיפוס המשתנה `arr` במחלקה `Cart` שונה ל-`Product`.
- נכתבו במחלקות המתאימות (כולל במחלקה `Book`) יחסי הורשה חדשים בינן ובין המחלקה `Product`.
- חלק מן התכונות שהופיעו עד עכשיו בכמה מן המחלקות האחרות (כולל המחלקה `Book`) הועברו למחלקה `Product`, בהתאם לעקרונות תכנות מונחה עצמים (מעבר לתכונות אלו אין תכונות נוספות במחלקה `Product`).

(1) סרטטו תרשים הייררכייה מעודכן המתאר את הקשר בין המחלקות:

`Product`, `Cart`, `Book`, `Pants`, `Shirt`, `Clothing`.

יש לסמן ירושה באמצעות החץ  והכלה באמצעות הסימן .

(2) פתבו את כותרת המחלקה `Product` ותכונותיה.

(3) פתבו מחדש את המחלקות שחל בהן שינוי בעקבות ההוספה של המחלקה `Product`.

שימו לב: יש להתייחס גם ליחסי הורשה חדשים וגם לשינוי שחל בתכונות המחלקה.

ג. בשלב האחרון, החליט בעל החנות להכריז על מבצע זמני: ברכישה מקוונת כל המוצרים יימכרו בהנחה של 10%, ובעבור רכישת בגדים (חולצות ומכנסיים) תהיה הנחה נוספת של 10 שקלים (הניחו שהמחיר נשאר חיובי לאחר ההנחה).

נתונה הפעולה הפנימית במחלקה `Product`:

```
public double getDiscountPrice()
```

זימון הפעולה מחזיר את מחיר המוצר (חולצה/מכנסיים/ספר) לאחר ההנחה - בהתאם לכל מוצר.

ממשו את הפעולה במחלקה `Product`, והוסיפו פעולה/ות במחלקה/ות המתאימה/ות כדי לבצע את הנדרש.

ציינו בעבור כל פעולה שהוספתם באיזו מחלקה היא נוספה.

אין להשתמש בפעולה `instanceof` ובפעולות המחלקה `Object` ואין לשנות את תכונות המחלקות.

15. נתונות שלוש מחלקות – Mammal (יונק), Antelope (אנטילופה) ו־ Beaver (בונה):

```

public class Mammal {
    protected int weight;

    public Mammal (int w) {
        weight = w;
    }
    public int getWeight () {
        return weight;
    }
    public boolean isSame (Mammal other) {
        System.out.println ("In Mammal");
        return (this == other);
    }
}

.....

public class Antelope extends Mammal {
    public Antelope (int w) { super (w); }
    public boolean isSame(Antelope other) {
        System.out.println ("In Antelope");
        return ((other != null) && (this.weight == other.weight));
    }
}

.....

public class Beaver extends Mammal {
    public Beaver (int w) { super (w); }
    public boolean isSame (Mammal other) {
        System.out.println ("In Beaver");
        return ((other != null) && (other instanceof Beaver) && (this.weight == ((Beaver)other).weight));
    }
}

.....

```

(שימו לב: המשך השאלה בעמוד הבא.)

נתונה המחלקה Program :

```

public class Program {
    public static void main (String[] args) {
        Antelope a1 = new Antelope (10);
        Object a2 = new Antelope (10);
        Beaver b1 = new Beaver (10);
        Mammal b2 = new Beaver (10);
        *****
    }
}

```

המחלקה Program נמצאת בחבילה (Package) שונה מן המחלקה Mammal .

הציבו כל אחת מן השורות 1–10 שלהלן בפעולה main , במקום שמסומן לעיל בכוכביות ***** .

פתבו במחברת את מספר השורה וציינו אם הקוד תקין או לא תקין. אם הקוד תקין – כתבו את הפלט, ואם הוא

לא תקין – הסבירו מדוע.

1. System.out.println (a1.weight);
2. System.out.println (((Beaver)a2).getWeight());
3. System.out.println (a1.isSame (a2));
4. System.out.println (a2.isSame (a1));
5. System.out.println (b1.isSame (b2));
6. System.out.println (b2.isSame (b1));
7. System.out.println (a1.isSame ((Beaver)b2));
8. System.out.println (a1.isSame ((Antelope)a2));
9. System.out.println (b1.isSame ((Antelope)a2));
10. System.out.println (b1.isSame ((Beaver)a2));

תכנות מונחה עצמים בשפת C#

אם למדתם מסלול זה ואתם כותבים בשפת C#, יש לענות על אחת מן השאלות 16–17 (25 נקודות).

16. בעל חנות כולבו מעוניין למכור את מוצריו גם באופן מקוון (אונליין). בשלב הראשון החליט בעל החנות למכור באופן

מקוון רק חולצות ומכנסיים.

לשם כך נבנו המחלקות האלה:

– המחלקה **Clothing**, המייצגת בגד

– המחלקה **Shirt**, המייצגת חולצה

– המחלקה **Pants**, המייצגת זוג מכנסיים

```
public class Clothing {
    private int id;           // מספר זיהוי של המוצר
    private string fabric;    // סוג הבד
    private string color;    // צבע הבגד
    private double price;    // מחיר
}
public class Shirt : Clothing {
    private string size;     // מידה (S, M, L, XL, XXL)
}
public class Pants : Clothing {
    private int length;     // אורך באינצ'ים
    private int width;     // רוחב באינצ'ים
}
```

הניחו שיש פעולות Get ו-Set בעבור התכונות בכל המחלקות.

א. המחלקה **Cart** מייצגת סל קניות. למחלקה שלוש תכונות:

name – שם הקונה

arr – מערך המכיל גם חולצות וגם מכנסיים

current – מספר המוצרים הנוכחי בקנייה

(1) כתבו את כותרת המחלקה Cart ואת התכונות שלה.

(2) סרטוטו תרשים הייררכייה המתאר את הקשר בין המחלקות: Clothing, Shirts, Pants, Cart.



(שימו לב: המשך השאלה בעמוד הבא.)

ג. בשלב השני החליט בעל החנות למכור באופן מקוון גם ספרים.

לפניכם המחלקה **Book** המייצגת ספר:

```
public class Book {
    private int id;           // מספר זיהוי של המוצר
    private string bookName; // שם הספר
    private string author;   // שם הסופר
    private double price;    // מחיר
}
```

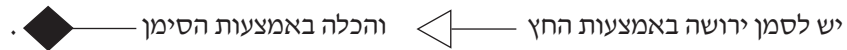
הניחו שיש פעולות Get ו- Set בעבור תכונות המחלקה.

כדי שהמערך arr במחלקה Cart יוכל להכיל גם ספרים, נעשו הפעולות האלה:

- הוגדרה המחלקה **Product** – מוצר.
- טיפוס המשתנה arr במחלקה Cart שונה ל- Product.
- נכתבו במחלקות המתאימות (כולל במחלקה Book) יחסי הורשה חדשים בינן ובין המחלקה Product.
- חלק מן התכונות שהופיעו עד עכשיו בכמה מן המחלקות האחרות (כולל המחלקה Book) הועברו למחלקה Product, בהתאם לעקרונות תכנות מונחה עצמים (מעבר לתכונות אלו אין תכונות נוספות במחלקה Product).

(1) סרטו תרשים הייררכייה מעודכן המתאר את הקשר בין המחלקות:

Product , Cart , Book , Shirt , Clothing .



(2) כתבו את כותרת המחלקה Product ותכונותיה.

(3) כתבו מחדש את המחלקות שחל בהן שינוי בעקבות ההוספה של המחלקה Product.

שימו לב: יש להתייחס גם ליחסי הורשה חדשים וגם לשינוי שחל בתכונות המחלקה.

ג. בשלב האחרון, החליט בעל החנות להכריז על מבצע זמני: ברכישה מקוונת כל המוצרים יימכרו בהנחה של 10%,

ובעבור רכישת בגדים (חולצות ומכנסיים) תהיה הנחה נוספת של 10 שקלים (הניחו שהמחיר נשאר חיובי לאחר ההנחה).

נתונה הפעולה הפנימית במחלקה Product:

```
public virtual double GetDiscountPrice()
```

זימון הפעולה מחזיר את מחיר המוצר (חולצה/מכנסיים/ספר) לאחר ההנחה – בהתאם לכל מוצר.

ממשו את הפעולה במחלקה Product, והוסיפו פעולה/ות במחלקה/ות המתאימה/ות כדי לבצע את הנדרש.

ציינו בעבור כל פעולה שהוספתם באיזו מחלקה היא נוספה.

אין להשתמש בפעולות is ו- as ובפעולות המחלקה Object ואין לשנות את תכונות המחלקות.

17. נתונות שלוש מחלקות – Mammal (יונק), Antelope (אנטילופה) ו־ Beaver (בונה):

```
public class Mammal {
    protected int weight;

    public Mammal (int w) {
        weight = w;
    }
    public int GetWeight () {
        return weight;
    }
    public virtual bool IsSame (Mammal other) {
        Console.WriteLine ("In Mammal");
        return (this == other);
    }
}
```

```
public class Antelope : Mammal {
    public Antelope (int w) : base (w) {}
    public bool IsSame(Antelope other) {
        Console.WriteLine ("In Antelope");
        return ((other != null) && (this.weight == other.weight));
    }
}
```

```
public class Beaver : Mammal {
    public Beaver (int w) : base (w) {}
    public override bool IsSame (Mammal other) {
        Console.WriteLine ("In Beaver");
        return ((other != null) && (other is Beaver) && (this.weight == ((Beaver)other).weight));
    }
}
```

(שימו לב: המשך השאלה בעמוד הבא.)

נתונה המחלקה Program :

```
public class Program {
    public static void Main (string[] args) {
        Antelope a1 = new Antelope (10);
        Object a2 = new Antelope (10);
        Beaver b1 = new Beaver (10);
        Mammal b2 = new Beaver (10);
        *****
    }
}
```

הציבו כל אחת מן השורות 1–10 שלהלן בפעולה Main, במקום שמסומן לעיל בכוכביות ***** .
 כתבו במחברת את מספר השורה וציינו אם הקוד תקין או לא תקין. אם הקוד תקין – כתבו את הפלט, ואם הוא
 לא תקין – הסבירו מדוע.

1. Console.WriteLine (a1.weight);
2. Console.WriteLine (((Beaver)a2).GetWeight());
3. Console.WriteLine (a1.IsSame (a2));
4. Console.WriteLine (a2.IsSame (a1));
5. Console.WriteLine (b1.IsSame (b2));
6. Console.WriteLine (b2.IsSame (b1));
7. Console.WriteLine (a1.IsSame ((Beaver)b2));
8. Console.WriteLine (a1.IsSame ((Antelope)a2));
9. Console.WriteLine (b1.IsSame ((Antelope)a2));
10. Console.WriteLine (b1.IsSame ((Beaver)a2));

בהצלחה!