

סוג הבדיקה: בגרות לבתי ספר על-יסודיים
 מועד הבדיקה: קיץ תשע"ו, מועד ב
 מספר השאלה: 899381

מְדֻעַי הַמְּחַשֵּׁב

על פי תכנית הרפורמה ללמידה משמעותית

הוראות לנבחן

- א. משך הבדיקה: שלוש שעות.
- ב. מבנה השאלה ופתחה הערכה: בשאלון זה שלושה פרקים.
פרק ראשון — בפרק זה שלוש שאלות, ענה על פי
 $(15 \times 1) + (10 \times 1)$ — 25 נקודות
פרק שני — בפרק זה שלוש שאלות,
 (25×2) — 50 נקודות
פרק שלישי — בפרק זה שאלות באربעה מסלולים שונים.
 (25×1) — 25 נקודות
סה"כ — 100 נקודות

- ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ מחשב הנitin לתכנות.

ד. הוראות מיוחדות:

- (1) את כל התכניות שאתה נדרש לכתוב בשפת מחשב בפרקים הראשון והשני כתוב בשפה אחת בלבד — Java או C#.
- (2) רשות על הכירכה החיצונית של המחברת אליו שפה אתה כותב — Java או C#.
- (3) רשות על הכירכה החיצונית של המחברת את שם המסלול שלמדת.
 המסלול הוא אתך מרבעת המסלולים האלה:
 מערכות מחשב ואסambilי, מבוא לחקיר ביצועים, מודלים חישוביים, תכנות מונחה עצמים.

הערה: בתכניות שאתה כותב לא יורד לך נקודות, אם כתובות אותן גדולה במקום אותן קטנה או להפוך.

כתב במחברת הבדיקה בלבד, בעמודים נפרדים, כל מה שברצונך לכתב בטיוטה (ראשי פרקים, חישובים וכדומה).
 רשום "טיוטה" בראש כל עמוד טיוטה. רישום טיווטות כלשון על דפים שמחוץ למחברת הבדיקה עלול לגרום לפסילת הבדיקה!

הנחהיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות ולנבחנים כאחד.

בהתלה!

/המשך מעבר לדף/

השאלות

בשאלון זה שלושה פרקים.

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

פרק ראשון (25 נקודות)

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

פתרונות ב- Java: בכל שאלה שנדרשת בה קליטה, הנח שבתכנותית כתובה הוראה:

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

ענה על שאלה 1 – חובב (10 נקודות)

1. נתונה פעולה ראשית במחלקה **Stam**. בפעולה הראשית הוגדר מערך חד-ממדי arr

מтиיפוס **שלם**.

כתבו ב- Java או ב- C#, בפעולה הראשית,قطع תכנית שיקלוט מספר שלם num, וידפיס את

סכום כל האיברים במערך שערכם קטן מ- num .

עונה על אחת מהשאלות 2 - 3 (15 נקודות)

.2. הגדרו מחלקה A בעלת שתי תכונות: n – מטיפוס שלם ו n2 – מטיפוס שלם.

במחלקה הוגדרו: שתי פועלות בונות,

. ולכל תכונה פועלות get ו set ב Java ו C# . Set ב C# .

במחלקה זו הפועלה ToString ב Java ו C# מוחירה מחרוזת שהיא תרגיל החיבור של ערכי n ו n2 , והתוצאה שלו.

לדוגמה:

אם הערך שב n הוא המספר 3 והערך שב n2 הוא המספר 9, תחזיר הפועלה את המחרוזות:
."3 + 9 = 12"

כמו כן הפועלה (void Add(int k) ב Java ו void add(int k) ב C#) מוסיפה את הערך k לערך של כל אחת מן התכונות n ו n2 .

הדפסת עצם מזמנת ב Java את הפועלה ToString של המחלקה וב C# את הפועלה של המחלקה.

לפניך הפועלה הראשית במחלקה Program כתובה ב Java וב C# .
הפועלה משתמשת במחלקה A .

Java

```
public class Program
{
    public static void main(String[] args)
    {
        A a = new A();
        a.setN1(4);
        a.setN2(5);
        A a1 = new A(a.getN1(), a.getN2());
        System.out.println (a);
        a1.add(4);
        System.out.println(a);
        System.out.println(a1);
    }
}
```

C#

```
public class Program
{
    public static void Main()
    {
        A a = new A();
        a.SetN1(4);
        a.SetN2(5);
        A a1 = new A(a.GetN1(), a.GetN2());
        Console.WriteLine(a);
        a1.Add(4);
        Console.WriteLine(a);
        Console.WriteLine(a1);
    }
}
```

(שים לב: המשך השאלה בעמוד הבא.)

- א. עליך להגדיר את המחלקה A על פי התת-סעיפים ו-ז' כך שהפעולה הראשית תבוצע
בלי שגיאות הידור או ריצה.
- i כתוב ב- Java או ב- C# את כותרת המחלקה ואת התכונות שלה.
 - ii כתוב ב- Java או ב- C# את הכותרות של שתי הפעולות הבונוס של המחלקה A.
 - iii מימוש ב- Java את כל הפעולות get ו- set או ב- C# את כל הפעולות Get ו- Set הנחוצות לריצה של הפעולה הראשית.
 - iv מימוש ב- Java את הפעולה add או ב- C# את הפעולה Add .
 - v מימוש ב- Java את הפעולה toString או ב- C# את הפעולה ToString .
- ב. הצג את העצמים הנוצרים עקב הפעלת הפעולה הראשית. לכל עצם רשום את השמות של
כל התכונות שלו ואת הערכים של כל התכונות.
כמו כן הצג את השינויים בעצמים בעקבות הוראה add(4) ב- Java או ב- C# Add(4) ב- Java.
רשום את הפלט של הפעולה הראשית.

/המשך בעמוד 5

- .3 לפניך ממשק המחלקה **Button** המייצגת כפתור. לכפתור שלוש תכונות:
מספר הרשום עליו – num, גודל הכפתור – size וצבע הכפתור – color.

कותרת הפעולה ב- Java	תיאור הפעולה
public Button(int num)	פעולה הבונה כפתור שרשום עליו המספר num , בגודל 5 , בצבע "black" .
public Button(int num , int size , String color)	פעולה הבונה כפתור שרשום עליו המספר num , בגודל size , בצבע color .
public int getSize()	פעולה המחזיר את גודל הכפתור.
public void addToSize(int x)	פעולה המגדילה את גודל הכפתור ב- x .
public boolean isSameSize(Button other)	פעולה המחזיר true אם הכפתור other זהה בגודלו לכפתור הנוכחי, אחרת – הפעולה מחזירה false .

कותרת הפעולה ב- C#	תיאור הפעולה
public Button(int num)	פעולה הבונה כפתור שרשום עליו המספר num , בגודל 5 , בצבע "black" .
public Button(int num , int size , string color)	פעולה הבונה כפתור שרשום עליו המספר num , בגודל size , בצבע color .
public int GetSize()	פעולה המחזיר את גודל הכפתור.
public void AddToSize(int x)	פעולה המגדילה את גודל הכפתור ב- x .
public bool IsSameSize(Button other)	פעולה המחזיר true אם הכפתור other זהה בגודלו לכפתור הנוכחי, אחרת – הפעולה מחזירה false .

- .א. ממש במחלקה **Button** , ב- C# או ב- Java , את הפעולה הבונה . Button(int num) .
 .ב. ממש במחלקה **Button** , ב- Java או ב- C# , את הפעולה המגדילה את גודל הכפתור ב- x .
 .ג. ממש במחלקה **Button** , ב- C# או ב- Java , את הפעולה הבודקת אם גודל הכפתור זהה לגודל הכפתור הנוכחי. other
 (שים לב: סעיף ד של השאלה בעמוד הבא).

.7. לפניך כתעת מפעולה ראשית במחלקה Program הכתוב ב- Java וב- C# .
 עקוב אחר ביצוע כתעת הפעולה, ורשותם את הפלט שיתקבל. לכל עצם רשותם את השמות של כל התוכנות שלו, את הערכיהם של כל התוכנות ואת השינויים שהיו בהם.

Java

```
Button a1 = new Button(1, 12, "red");
Button a2 = new Button(2, 12, "green");
Button a3 = new Button(3, 12, "blue");
Button a4 = new Button(4);
a1.addTosize(2);
a4.addTosize(9);
if (a1.isSameSize(a3))
    System.out.println("***");
if (a4.isSameSize(a1))
    System.out.println("$$$");
if (a2.isSameSize(a3))
    System.out.println("###");
```

C#

```
Button a1 = new Button(1, 12, "red");
Button a2 = new Button(2, 12, "green");
Button a3 = new Button(3, 12, "blue");
Button a4 = new Button(4);
a1.AddTosize(2);
a4.AddTosize(9);
if (a1.IsSameSize(a3))
    Console.WriteLine("***");
if (a4.IsSameSize(a1))
    Console.WriteLine("$$$");
if (a2.IsSameSize(a3))
    Console.WriteLine("###");
```

/המשך בעמוד 7

פרק שני (50 נקודות)

שים לב: בכל שאלה שנדרש בה שימוש אתה יכול להשתמש בעולות של המחלקות **טור, מחסנית, עץ בינרי וחוליה**, בלי למשמש אותן. אם אתה משתמש בעולות נוספות, عليك למשמש אותן.

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

4. רשימה L תיקרא **מושולשת** אם היא מקיימת את התנאים הבאים:

- * הרשימה אינה ריקה.
 - * מספר האיברים בה מתחלק ב-3 בלי שארית.
 - * האיברים בשליש הראשון של הרשימה מכילים את אותם ערכים שמכילים האיברים בשליש השני של הרשימה ואוטם ערכים שמכילים האיברים בשליש השלישי של הרשימה.
 - * הערכים מסוודרים באותו סדר בכל אחד מהשלישים.
- לדוגמה: הרשימה L1 שלפניך היא רשימה **מושולשת** באורך 12.

L1: 

כתב ב- C# או ב- Java פועלה חיצונית שתתקבל רשימה L שהאיברים שלה הם מטיבוס שלם.

אם L היא רשימה **מושולשת**, הפעולה תחזיר true .

אחרת — הפעולה תחזיר false .

שים לב: לשאלה זו שני נוסחים: אחד בו Java (עמודים 9-8), ואחד בו C# (עמודים 11-10). עבוד על פי השפה שלמדת.

.5

לפתרונות ב- Java

לפניך הפעולות sod ו what המקבלות מערך a שאיבריו מטיפוס שלם, ממוקן בסדר עולה, ומספר שלם k. לשתי הפעולות אותה טענת יציאה.

```
public static boolean sod(int[] a , int k)
{
    for (int i = 0; i < a.length-1; i++)
    {
        int j = i+1;
        while (j < a.length)
        {
            if (a[i] + a[j] == k)
                return true;
            j++;
        }
    }
    return false;
}
```

```
public static boolean what(int[] a , int k)
{
    int left = 0 , right = a.length-1;
    while (left < right)
    {
        if (a[left] + a[right] == k)
            return true;
        if (a[left] + a[right] < k)
            left++;
        else
            right--;
    }
    return false;
}
```

a	2	4	7	12	18
---	---	---	---	----	----

- .א. עוקב בעזרת טבלת מעקב אחר ביצוע הפעולה `sod` בעבר המערך הנתון a
והמספר $k = 11$. רשום את הערך המוחזר.
בטבלת המיעקב יש לכלול עמודות בעבר: $[j]$, $a[i]$, j , ועמודה נוספת שבה יצוין אם התנאי שבקוד if מתקיים או אינו מתקיים.
- .ב. עוקב בעזרת טבלת מעקב אחר ביצוע הפעולה `sod` בעבר המערך הנתון a
והמספר $k = 10$. רשום את הערך המוחזר.
בטבלת המיעקב יש לכלול את העמודות שפורטו בסעיף א.
- .ג. מהי טענת היציאה של הפעולה `sod` ?
מהי סיבוכיות זמן הריצה של הפעולה `sod` ? נמק את תשובתך.
- .ד. עוקב בעזרת טבלת מעקב אחר ביצוע הפעולה `what` בעבר המערך הנתון a
והמספר $k = 11$. רשום את הערך המוחזר.
בטבלת המיעקב יש לכלול עמודות בעבר: $[a[left], a[right]]$, $left$, $right$,
ושתי עמודות נוספות לכליacht מפקודות if . בכל עמודה יצוין אם התנאי בפקודת if
מתקיים או אינו מתקיים.
- .ג. מהי סיבוכיות זמן הריצה של הפעולה `what` ? נמק את תשובתך.
מי מבין שתי הפעולות — `sod` או `what` — יעילה יותר? נמק את תשובתך.
- .ח. טענת הכניסה של הפעולות `sod` ו- `what` שונות כך שאפשר להעביר אליה
מערך a לא ממויין.
- (1) האם טענת היציאה של הפעולה `sod` תשתנה? נמק את תשובתך.
(2) האם טענת היציאה של הפעולה `what` תשתנה? נמק את תשובתך.

C# – לפוטרים ב-

לפניך הפעולות Sod ו- What המקובלות מערך a שאיבריו מתtipוס שלם, ממוקן בסדר עולה, ומספר שלם k. לשתי הפעולות אותה טענת יציה.

```
public static bool Sod(int[] a , int k)
{
    for (int i = 0; i < a.Length-1; i++)
    {
        int j = i+1;
        while (j < a.Length)
        {
            if (a[i] + a[j] == k)
                return true;
            j++;
        }
    }
    return false;
}
```

```
public static bool What(int[] a , int k)
{
    int left = 0 , right = a.Length-1;
    while (left < right)
    {
        if (a[left] + a[right] == k)
            return true;
        if (a[left] + a[right] < k)
            left++;
        else
            right--;
    }
    return false;
}
```

a	2	4	7	12	18
---	---	---	---	----	----

- א. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה Sod בעבור המערך הנתון a
והמספר $11 = k$. רשום את הערך המוחזר.
בטבלת המעקב יש לכלול עמודות בעבור: $[j] , i , a[i] , j$, ועמודה נוספת שבה יצוין אם התנאי שבקודת if מתקיים או אינו מתקיים.
- ב. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה Sod בעבור המערך הנתון a
והמספר $10 = k$. רשום את הערך המוחזר.
בטבלת המעקב יש לכלול את העמודות שפורטו בסעיף א.
מהי טענת הייצאה של הפעולה Sod ?
מהי סיבוכיות זמן הריצה של הפעולה Sod ? נמק את תשובתך.
- ג. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה What בעבור המערך הנתון a
והמספר $11 = k$. רשום את הערך המוחזר.
בטבלת המעקב יש לכלול עמודות בעבור: $[j] , left, right, a[left], a[right]$, ושתי עמודות נוספות לכל אחת מפקודות if . בכל עמודה יצוין אם התנאי בפקודת if מתקיים או אינו מתקיים.
מהי סיבוכיות זמן הריצה של הפעולה What ? נמק את תשובתך.
- ד. מי מבין שתי הפעולות — Sod או What — עיליה יותר? נמק את תשובתך.
טענת הנכינה של הפעולות Sod ו- What שונות כך שאפשר להעביר אליהן מערך a לא ממויין.
(1) האם טענת הייצאה של הפעולה Sod תשתנה? נמק את תשובתך.
(2) האם טענת הייצאה של הפעולה What תשתנה? נמק את תשובתך.

שים לב: לשאלה זו שני נוסחים: אחד ב- Java (עמודים 12-13), ואחד ב- C# (עמודים 14-15). עמוד על פי השפה של마다.

לפותרים ב- Java

המשחק 'שחק נא' הוא משחק לשחקן אחד בשני שלבים. השחקן מקבל 52 קלפים מעורבבים, שונים זה מזה. על כל קלף יש מספר בין 1 ל- 13, וצורה של אחת מארבע צורות. הצורות מיוצגות על ידי המספרים 1 עד 4.

בשלב הראשון השחקן מחלק את הקלפים לארבע ערכות על פי הצורה של הקלף, כך שכל ערמה יש קלפים עם אותה צורה. הקלפים מונחים זה על גבי זה.

בשלב השני השחקן מגיריל מספר בין 1 ל- 4 המייצג צורה של קלף. השחקן פונה לערמה שעלה כל הקלפים בה נמצאת הצורה המיוצגת על ידי המספר שהוגרל. הוא מרים את הקלף שבראש הערמה, ו מעביר אותו לערמה חמישית.

שלב זה יבוצע עד שהמשחק יסתתיים.

המשחק מסתיים כאשר מוגרל מספר המייצג ערמה שאין בה קלפים. ניחנו הוא מצב שבו סכום המספרים שעל הקלפים בערמה חמישית מתחלק ב- 100 ללא שארית.

לאחר שהמשחק מסתיים בודקים אם יש ניצחון.

לצורך מיומוש המשחק הוגדרו המחלקות: **Card** המייצגת קלף אחד, **Deck** המייצגת את חמש הערכות הנדרשות במשחק, ו- **Test** המנהלת את המשחק.

כמו כן נכתבת במחלקה **Test** הפעולה `public static boolean game(Card[] cards)`, המתקבלת כמספרם של 52 קלפים שונים המסודרים באופן אקראי. הפעולה מנהלת את המשחק עד סיוםו.

הפעולה מחזירה `true` אם המשחק הסתיים בניצחון, ואחרת — מחזירה `false`.

לפניך המחלקה **Card** המייצגת קלף שעליו מספר (`value`) בין 1 ל- 13 וצורה (`shape`) בין 1 ל- 4.

```
public class Card
{
    private int value;
    private int shape;

    public Card (int value, int shape)
    {
        this.value = value;
        this.shape = shape;
    }

    public int getValue () { return this.value; }

    public int getShape () { return this.shape; }
}
```

- א. i.** כתוב ב- Java את כותרת המחלקה **Deck** ואת התכונות שלה.
- א. ii.** כתוב ב- Java במחלקה **Deck** את כותרות הפעולות האלה:
- פעולה בונה המגדירה את חמש הערמות להיות ריקות.
 - פעולה insert מקבלת קלף ומוסיפה אותו לערומה הנכונה על פי השלב הראשון במשחק.
 - פעולה move המגרילה מספר המציג צורה. אם ערומה הקלפים שצורתה הוגלה היא ריקה, הפעולה תחזיר false. אחרת — הפעולה תעביר לערומה החמישית את הקלף שבראש הערומה שמספרה הוגול ותחזיר true.
 - פעולה sum המזירה את הסכום הנוכחי של המספרים שעל הקלפים בערומה החמישית.
- ב. סימן לב:** בתת-סעיף זה אין צורך למש את הפעולות.
- . ב.** ממש ב- Java את הפעולה הבונה במחלקה **Deck**.
- . ג.** ממש ב- Java את הפעולה sum במחלקה **Deck**.
- . ד.** ממש ב- Java את הפעולה game במחלקה **Test**.
- אתה יכול להשתמש בפעולות insert ו- move של המחלקה **Deck**, בלי למש אותן. אם אתה משתמש בפעולות נוספות נוספות, עליך למש אותן.

לפתרונות ב- C#

המשחק 'שחק נא' הוא משחק לשחקן אחד בשני שלבים. השחקן מקבל 52 קלפים מעורבבים, שונים זה מזה. על כל קלף יש מספר בין 1 ל- 13, וצורה של אחת מארבע צורות. הוצאות מיוצגות על ידי המספרים 1 עד 4.

בשלב הראשון השחקן מחלק את הקלפים לארבע ערכות על פי הצורה של הקלף, כך שכל ערמה יש קלפים עם אותה צורה. הקלפים מונחים זה על גבי זה.

בשלב השני השחקן מגיריל מספר בין 1 ל- 4 המייצג צורה של קלף. השחקן פונה לערמה שעלה כל הקלפים בה נמצאת הצורה שמיוצגת על ידי המספר שהוגרל. הוא מרים את הקלף שבראש הערמה, ו מעביר אותו לערמה חמישית.

שלב זה יבוצע עד שהמשחק יסתתיים.

המשחק מסתיים כאשר מוגריל מספר המייצג ערמה שאין בה קלפים.

ניחסו הוא מצב שבו סכום המספרים שעלה הקלפים בערמה חמישית מתחלק ב- 100 ללא שארית.

לאחר שהשחקן מסתיים בודקים אם יש ניצחון.

לצורך מיימוש המשחק הוגדרו המחלקות: **Card** המייצגת קלף אחד, **Deck** המייצגת את חמיש הערמות הנדרשות במשחק, ו- **Test** המנהלת את המשחק.

כמו כן נכתבת במחלקה **Test** הפעולה Game(Card[] cards), המקבילהurremotot הערמות הנדרשות במשחק, ו- **Test** המנהלת את המשחק. הפעולה Game(Card[] cards) מקבלת 52 קלפים שונים המסודרים באופן אקראי. הפעולה מנהלת את המשחק עד סיוםו. הפעולה מחזירה true אם המשחק הסתיים בניצחון, ואחרת — מחזירה false.

לפניך המחלקה **Card** המייצגת קלף שעליו מספר (value) בין 1 ל- 13 וצורה (shape) בין 1 ל- 4.

```
public class Card
{
    private int value;
    private int shape;

    public Card (int value, int shape)
    {
        this.value = value;
        this.shape = shape;
    }

    public int GetValue () { return this.value; }

    public int GetShape () { return this.shape; }
}
```

- א.**
- i. כתוב ב- C# את כותרת המחלקה **Deck** ואת התכונות שלה.
 - ii. כתוב ב- C# במחלקה **Deck** את כותרות הפעולות האלה:
 - פעולה **Build** המגדירה את חמש הערכות להיות ריקות.
 - פעולה **Insert** מקבלת קלף ומוסיפה אותו לערימה הנכונה על פי השלב הראשוני במשחק.
 - פעולה **Move** המגרילה מספר המיצג צורה. אם ערימת הקלפים שצורתה הוגלה היא ריקה, הפעולה תחזיר `false`. אחרת — הפעולה תעביר לערימה החמישית את הקלף שבראש הערימה שמספרה הוגרל ותחזיר `true`.
 - פעולה **Sum** המחזירה את הסכום הנוכחי של המספרים שעל הקלפים בערימה החמישית.
- ב.** שים לב: בתת-סעיף זה אין צורך למש את הפעולות.
- ג.** ממש ב- C# את הפעולה הבונה במחלקה **Deck**.
- ד.** ממש ב- C# את הפעולה **Sum** במחלקה **Deck**.
- אתה יכול להשתמש בפעולות **Insert** ו- **Move** של המחלקה **Deck** בלי למש אותן. אם אתה משתמש בפעולות נוספות, עליך למש אותן.

פרק שלישי (25 נקודות)

בפרק זה שאלות ארבעה מסלולים:
מערכות מחשב וasm, עמודים 16-17.
מבוא לחקיר ביצועים, עמודים 18-24.
מודלים חישוביים, עמוד 25-36.
תכנות מונחה עצמים ב- Java, עמודים 31-32; תכנות מונחה עצמים ב- C#, עמודים 32-36.
ענה על שאלה אחת במסלול שלמה.

מערכות מחשב וasm

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

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

א. לפניך קטע תכנית באסמלבי:

שים לב: הנתונים הם הקסדצימליים.

START:	MOV	AX	,	C83BH
	MOV	BX	,	A89CH
	SHL	AX	,	1
	OR	AL	,	33H
	NOT	BL		
	ADD	AX	,	BX

עקב בעורת טבלת מעקב אחר ביצוע התכנית.
בטבלת המעקב פרט את התוכן של AX, BX, ZF, SF, CF בכל שלב.

.ב. אין קשר לסעיף א.)

במקרה הנתונים הוגדר מערך:

ARR DB 5 DUP (?)

. לפניך 4 קטעים i-iv באסמבלי שمطلوبם לאותחל לי 0 את כל תאי המערך ARR. עקוב בעזרת טבלת מעקב אחרי הביצוע של בלאחד מהקטעים i-iv שלפניך, וקבע אם הוא מבצע את הנדרש או אינו מבצע את הנדרש.

i	MOV	SI , 0
	MOV	CX , 4
A1:	MOV	ARR[SI] , 0
	INC	SI
	LOOP	A1

ii	MOV	CX , 5
	LEA	BX , ARR
	MOV	AL , 0
A1:	MOV	[BX] , AL
	INC	BX
	LOOP	A1

iii	MOV	BX , 5
	DEC	BX
A1:	MOV	ARR[BX] , 0
	DEC	BX
	JNZ	A1

iv	MOV	DI , 0
A1:	MOV	ARR[DI] , 0
	INC	DI
	CMP	DI , 5
	JC	A1

.8. באוגר AX מאוחסן מספר ביברי. כתוב קטע תכנית באסמבלי, שייציב באוגר BL את מספר הפעמים שהרצף 1011 מופיע במספר הביברי שבאוגר AX.

לדוגמא: אם באוגר AX מאוחסן המספר הביברי: 0110110010110110 _____

יוצב באוגר BL המספר 3.

/ המשך בעמוד 18

מבוא לחקור ביצועים

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

.9. נתונה בעיית תכנון ליניארי:

$$\text{Max } \{z = 5x_1 - x_2\}$$

בכפוף לאיוצים אלה:

$$2x_1 + 3x_2 \leq 18$$

$$x_1 - x_2 \leq 2$$

$$x_1 \leq 3$$

$$x_2 \geq 0$$

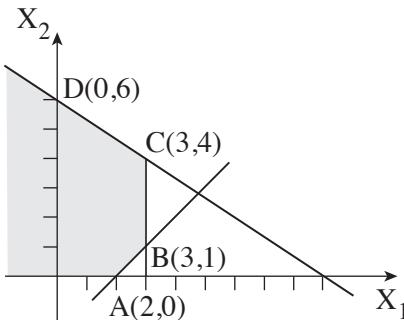
x_1 אינו מוגבל בסימן.

נתון גם כי הפתרון (1, 3) הוא פתרון אופטימלי של הבעיה.

כל אחד מהסעיפים א-ה שלפני מתיחס לבעיית התכנון הליניארי הנתונה.

הסעיפים איןם תלויים זה זהה. ענה על בל הסעיפים.

.א. לפניה סרטוט של תחום הפתרונות האפשריים של הבעיה הנתונה.



העתק למחברתך את הסרטוט, ובצע את הצעדים האלה:

צעד 1: הוסף לסרטוט שבמחברתך את היטל הגובה של פונקציית המטרה

$$\text{בעבור } z = 5.$$

חשב את שיעורי נקודות החיתוך של היטל זה עם הצירים x_1 ו- x_2 , וסמן אותן על הסרטוט.

צעד 2: הוסף לסרטוט שבמחברתך את היטל הגובה של פונקציית המטרה

$$\text{בעבור } z = 15.$$

חשב את שיעורי נקודות החיתוך של היטל זה עם הצירים x_1 ו- x_2 , וסמן אותן על הסרטוט.

задание 3: Семантический анализ в задаче оптимизации в линейном программировании.

ам из трех вариантов решений (1, 2, 3) какое оптимальное?

Любая ситуация в начальной задаче добавляет следующий ограничение:

$$x_1 - x_2 \geq 2$$

Сделать в задаче новые ограничения, и семантика только на предметах возможных решений.

Любая ситуация в начальной задаче добавляет следующее ограничение:

в пределах возможных решений есть точка, отличная от нуля.

ам из трех вариантов оптимальное?

Любая ситуация в начальной задаче добавляет следующее ограничение:

$$3x_1 + 2x_2 \leq 14$$

ученик утверждает, что оптимальное значение (1, 3) не изменилось.

ам ученик прав? Найдите в задаче новые возможные решения.

после добавления ограничения.

меньшим только ограничение на вторую координату для.

внутри единицы ? - это один из них. Переформулируйте единицу в соответствии с заданным ограничением.

i) оптимальное значение будет (1, 3).

ii) оптимальное значение будет (3, 4).

iii) оптимальное значение будет:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \lambda \begin{pmatrix} 3 \\ 1 \end{pmatrix} + (1 - \lambda) \begin{pmatrix} 3 \\ 4 \end{pmatrix}$$

если $0 \leq \lambda \leq 1$

iv) оптимальное значение не будет ограничено.

- . 10. בשאלת זו שישה סעיפים, א-ו, שאינם תלויים זה זהה. ענה על כל הסעיפים.
- א. לפניך בעיית תובלה ובה 2 מקורות ו 4 יעדים. העליות ליחידה מכל מקור לכלי יעד נתונות בטבלה שלפניך.

מקורות	יעדים				היצוא
	1	2	3	4	
1	8	9	4	6	100
2	14	12	13	8	80
ביקוש	10	40	30	70	

על פי הטבלה הנתונה לא ניתן להפעיל את שיטת הפינה הצפון-מערבית כדי למצוא פתרון אפשרי לבעיית התובלה. הסבר מדוע.

- . ב. בטבלה שלפניך נתון חלק מפתרון בסיסי אפשרי לבעיית תובלה: $x_{11} = 10$, $x_{12} = 2$.

מקורות	יעדים			היצוא
	1	2	3	
1	4	5	10	12
	10	2		
2	6	3	6	12
3	6	2	6	6
ביקוש	10	10	10	

העתק את הטבלה למחברתך, והשלם אותה לפי שיטת הפינה הצפון-מערבית.

6. בטבלה של פניך נתון פתרון בסיסי אפשרי לביעית תובלה. הערך שנקבע ל- u_1 הוא 0.

העתק את הטבלה למחברתך, חשב את הערך של v_1, v_2, v_3, u_2, u_3 , ורשום את הערכים שחייבים במקומות המתאים בטבלה.

מקורות	יעדים			היצע	u_i
	1	2	3		
1	4 10	5 2	10	12	0
2	6	3 2	6 10	12	
3	6	2 6	6	6	
ביקוש	10	10	10		
v_j					

(שים לב: המשך השאלה בעמוד הבא.)

/המשך בעמוד 22/

. 7. בטבלה של פניך נתון פתרון בסיסי אפשרי לביעית תובלה, ונתונים ערכיים

של $v_1, v_2, v_3, u_1, u_2, u_3$.

מקורות	יעדים			היצוא	u_i
	1	2	3		
1	4 20	5 4	10	24	0
2	2	6 16	3 4	20	1
3	6	2	6 8	8	4
ביקוש	20	20	12		
v_j	4	5	2		

האם הפתרון הוא אופטימלי? נמק את תשובה.

ה. בטבלה שלפניך נתון פתרון לביעית תובלה,
ונתונים ערכים של $v_1, v_2, v_3, u_1, u_2, u_3$, שמתאים לפתרון זה.

מקורות	יעדים			היצע	u_i
	1	2	3		
1	10 20	25	30	20	10
2	10 30	22	14 20	50	10
3	16 40	20 40	20 20	60	16
ביקוש	50	40	40		
v_j	0	4	4		

לפניך היגדים 1-3 שرك אחד מהם נכון. העתק למחברתך את ההיגד הנוכחי, ונמק את בחרתך.

- i הפתרון הנתון אינו פתרון אפשרי.
- ii הפתרון הנתון הוא פתרון בסיסי אפשרי אך לא 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.)

ו.

בטבלה של פניך נתון חלק מפתרון בסיסי אפשרי לביעית תובלה שחוושב על פי שיטת הפינה הצפון מערבית, ונתונים ערכיים של $v_1, v_2, v_3, u_1, u_2, u_3$ שהושבו על פי פתרון זה.

מקורות	יעדים			היצע	u_i
	1	2	3		
1	12 20	15	17	20	2
2	10	18 10	14	10	0
3	20	10 5	18 10	15	-8
ביקוש	20	15	10		
v_j	10	18	26		

מה צריך להוסיף לטבלה כדי לקבל פתרון בסיסי אפשרי המתאים לכל הנתונים שבטבלה?

העתק למחברתך את התשובה הנכונה מבין האפשרויות -iv, ונמك את תשובהך.

. $X_{12} = 0$ i

. $X_{21} = 0$ ii

. $X_{13} = 0$ iii

. $X_{31} = 0$ iv

מודלים חישוביים

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

11. לפניך השפה L מעלה א"ב $\{ \$, \$0^{i_1} \$0^{i_2} \$ \dots 0^{i_k} \$ \}$.

$$L = \left\{ 0^3 \$ 0^{i_1} \$ 0^{i_2} \$ \dots 0^{i_k} \$ \mid \begin{array}{l} k \geq 1 \\ \text{כל } m \text{ בין } 1 \text{ ל- } k : i_m \geq 0 \\ \text{ו- } i_m \text{ מחלק ב- } 3 \text{ ללא שארית} \end{array} \right\}$$

א. כתוב את המילה הקצרה ביותר בשפה L .

ב. בנה אוטומט סופי דטרמיניסטי שיקבל את השפה L .

12. בניית מכונה טיורינג המחשבת את הפונקציה $f(n, m)$. המכונה זו מוצאת את המספר הקטן מבין שני מספרים m ו- n .

המכונה מקבלת כקלט שני מספרים m ו- n שלמים גדולים מ-0. שני המספרים רשומים על הסרט כמספרים אונריים (מספר אונרי m הוא מספר המוצג על ידי m תווים של 1) ובין שני המספרים רשות הסימן #.

הפלט של המכונה הוא המספר הקטן מבין שני המספרים, והוא יירשם על הסרט כמספר אונרי בין שני סימני \$. סימן \$-\$ השמאלי יירשם במקום הסימן #, והמספר יירשם מימינו.

במהלך הפעולה המכונה יכולה להיעזר בסימנים a, b.

לדוגמה: בעבר הקלט $m = 3$ ו- $n = 5$, לפני החישוב סרט הזיכרון יראה כך:

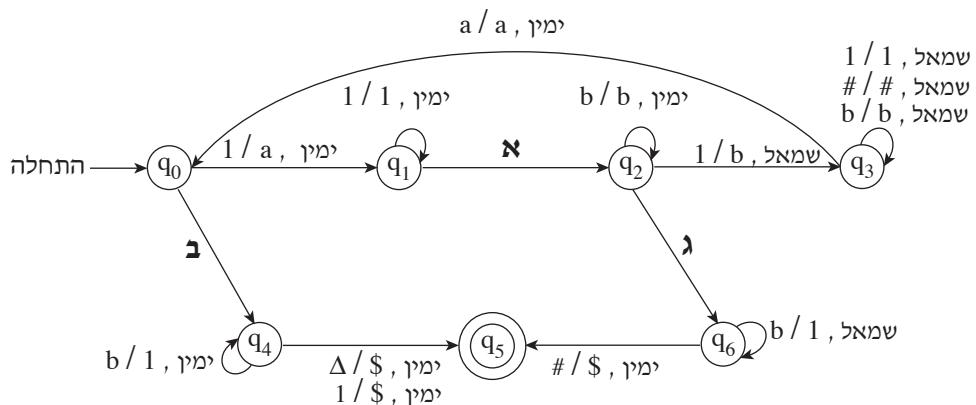
	+	1	1	1	#	1	1	1	1	1	Δ	Δ	Δ	...
--	---	---	---	---	---	---	---	---	---	---	---	---	---	-----

לאחר סיום החישוב יראה הסרט כך:

	+	\$	1	1	1	\$...
--	---	-----	-----	-----	----	---	---	---	----	-----

הערה: התאים המסומנים ב... הם תאימים שאין חשיבות לתוכנן.

לפניך סרטוט חלקי של המכונה.



א. בסרטוט יש שלושה מעברים מסומנים באותיות א-ג.

במעברים אלה חסרים סימני הקלט והווראות.

העתק למחברתך את הסרטוט, והשלם את שלושת המעברים החסרים כך שהמכונה תחשב את הפונקציה $f(n, m)$.

ב. הראה את תהליך החישוב של המכונה בעבר הקלט $n = 1, m = 1$.

ככל שלב רשם את מצב הסרט, סמן היכן נמצא ראש המכונה, ורשם באיזה מהמצבים

q6 - q המכונה נמצאת. המספר בעמוד 27 /

תכנות מונחה עצמים

אם למדת מסלול זה ואתה כותב ב- Java, ענה על אחת מהשאלות 13-14. 25) נקודות).
לפניך חלק פרויקט העוסק בכלל תחבורה וככל את המחלקות האלה:

מייצגת כלי תחבורה	מחלקה Vehicle
מייצגת רכבת שהיא כלי תחבורה	מחלקה Train
מייצגת סירה שהיא כלי תחבורה	מחלקה Boat
מייצגת מטוס שהוא כלי תחבורה	מחלקה Airplane
מייצגת חברת שיש לה כלי תחבורה מסווגים שונים	מחלקה TransportationCompany

Public class **Vehicle**

```
{
    private String type;           // מיקום (יבשה / מים / אויר)
    private String way;            // סוג הדרך (כביש / מסילה / נהר / ...)
    private int maxSpeed;          // מהירות מקסימלית
}
```

```
public Vehicle(String type, String way, int maxSpeed)
```

```
{
    this.type = type;
    this.way = way;
    this.maxSpeed = maxSpeed
}
}
```

public class **Train** extends **Vehicle**

```
{
    private int numOfCarriages;     // מספר הקרונות
    public Train(int maxSpeed, int numOfCarriages)
    {
        super("land", "tracks", maxSpeed);
        this.numOfCarriages = numOfCarriages;
    }
}
```

```
magdila b-ך את מספר הקרונות ברכבת //  

{
    this.numOfCarriages = this.numOfCarriages + n;
}
}
```

public class **Boat** extends **Vehicle**

```
{  
    public Boat(String way, int maxSpeed)  
    {  
        super("water", way , maxSpeed);  
    }  
}
```

public class **Airplane** extends **Vehicle**

```
{  
    private int maxHeight;           // גובה טיסה מקסימלי  
    public Airplane(int maxSpeed, int maxHeight)  
    {  
        super("sky", "air", maxSpeed)  
        this.maxHeight = maxHeight;  
    }  
}
```

public class **TransportationCompany**

```
{  
    private Vehicle[] vehicles = new Vehicle[50];   // מערך כל התחבורה בחברה  
    private int counter = 0;                          // מספר כל התחבורה שיש בפועל  
    public TransportationCompany()  
    {  
    }
```

מוסיפה כל תחבורה למספר כל התחבורה של (v) // החברה. הנח שיש מקום להוסיף כל תחבורה. //

```
    this.vehicles[counter] = v;  
    this.counter++;  
}  
}
```

(שים לב: המשך השאלה בעמוד הבא.)
/המשך בעמוד 29/

א. ממש ב- Java מחלקה ראשית Program ובה פעולה ראשית, שתבצע את המשימות האלה:

- i** בנייתו של עצם מטיפוס חברת כליה תחבורה – **TransportationCompany**
- ii** הוספה של סירה אחת ורכבת אחת לחברת **company1**.
- iii** בחור לתוכנות ערכיהם כרצונו.

ב. במחלקה **TransportationCompany** הוגדרה הפעולה:

```
public void display()
{
    for (int i=0; i<this.counter; i++)
    {
        System.out.println((i+1) + ":" + this.vehicles[i]);
    }
}
```

משב Java פועלות שיאפשרו ביצוע תקין של הפעולה `display()`, כך שבubo כל כליה תחבורה יודפסו כל התוכנות שלו. הגדר את הפעולות באופן המתאים ביותר לעקרונות של תכונות מונחה עצמים (הכמה – encapsulation, הירושה – inheritance – polymorphism –).

בעבור כל פעולה שאתה ממש, רשום לאיזו מחלקה היא שייכת. אין לשנות את הפעולה `display()`.

ג. ממש ב- Java פעולה, שתקבל מספר שלם `n` ותוסיף `n` קרונות לכל הרכבות שישיות להחברה שיש לה כליה תחבורה מסווגים שונים. تعد את הפעולה, ורשום באיזו מחלקה יש להגדיר אותה. אין לשנות את הפעולות הקיימות בפרויקט.

.14. לפניך המחלקות **AA** ו- **BB** :

```

public class AA
{
    private String st;

    public AA()          { this.st = "excellent"; }
    public AA(String st) { this.st = st; }
    public String getSt() { return this.st; }
    public void setSt(String st) { this.st = st; }
    public String toString() { return "st = " + this.st; }
}

public class BB extends AA
{
    private int num;

    public BB()          { super(); this.num = 1; }
    public BB(int num , String st) { super(st); this.num = Math.abs(num); }
    public int getNum()   { return this.num; }
    public void setNum(int num) { this.num = num; }
    public String toString() { return super.toString() + " num = " + this.num; }
}

```

- א.** הגדר במחלקה **AA** פועלה בוליאנית בשם `isLike(Object obj)` המקבלת עצם `obj` כערך. אם העצם `obj` הינו מטיפוס **AA** וגם תוכן המחרוזת `st` של `obj` מטיפוס `Object`. אם העצם `obj` הינו מטיפוס **AA** וגם תוכן המחרוזת `st` של `obj` זהה לתוכן המחרוזת `st` של העצם הנוכחי — הפועלה תחזיר `true`, אחרת — תחזיר `false`.
- ב.** הגדר במחלקה **BB** פועלה הדורשת את הפועלה שהגדרת בסעיף א. אם העצם `obj` הינו מטיפוס **BB** וגם ערך התכונה `num` שלו זהה לערך התכונה `num` של העצם הנוכחי — הפועלה תחזיר `true`, אחרת — תחזיר `false`.
- /המשך בעמוד 31/

ג. לפניך קטע מפעולה ראשית:

```
AA a = new AA("excellent");
BB b = new BB();
a = b;
if (a.isLike(b)) System.out.println(a);
```

האם קטע התכנית תקין?

אם כן — מה יהיה פלט הקטע? רשוםஇயோ גרסה של הפעולה `isLike` תופעל — זו של `AA` או זו של `BB`.

אם לא — הסבר מהי השגיאה ומתי היא תתגללה: בזמן קומpileציה או בזמן ריצה.

ד. לפניך קטע מפעולה ראשית:

```
AA aa = new AA();
BB bb = new BB(2 , "excellent");
bb = aa;
if (bb.isLike(aa)) System.out.println(bb);
```

האם קטע התכנית תקין?

אם כן — מה יהיה פלט הקטע? רשוםஇயோ גרסה של הפעולה `isLike` תופעל — זו של `AA` או זו של `BB`.

אם לא — הסבר מהי השגיאה ומתי היא תתגללה: בזמן קומpileציה או בזמן ריצה.

ה. כתוב פעולה חיצונית בשם `longString` מקבלת מערך של עצמים מטיפוס `Object`.

הפעולה מחזירה מחרוזת המורכבת משרשור התוכנה `st` של עצמים מטיפוס `AA` ב眯ירר, באופן זהה:

— אם לעצם יש בק התוכנה `st`, תשורשר המחרוזת שבתוכנה `st` פעם אחת.

— אם לעצם יש גם התוכנה `ssst`, המחרוזת שבתוכנה `st` תשורשר `ssst` פעמיים.

— אם אין ב眯ירר אף עצם מטיפוס `AA`, תוחזר מחרוזת ריקה.

תכנות מונחה עצמים

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

15. לפניך חלק מפרויקט העוסק בכלי תחבורה וכ כולל את המחלקות האלה:

מייצגת <u>כלי תחבורה</u>	המחלקה Vehicle
מייצגת רכבת שהיא <u>כלי תחבורה</u>	המחלקה Train
מייצגת סירה שהיא <u>כלי תחבורה</u>	המחלקה Boat
מייצגת מטוס שהוא <u>כלי תחבורה</u>	המחלקה Airplane
מייצגת חברה שיש לה <u>כלי תחבורה</u> מסווגים שונים	המחלקה TransportationCompany

Public class **Vehicle**

{

```
private string type;           // מיקום (יבשה / מים / אויר)
private string way;           // סוג הדרך (כביש / מסילה / נהר / ...)
private int maxSpeed;         // מהירות מקסימלית
```

```
public Vehicle(string type, string way, int maxSpeed)
```

{

```
    this.type = type;
    this.way = way;
    this.maxSpeed = maxSpeed
```

}

}

public class **Train : Vehicle**

{

```
private int numOfCarriages      // מספר الكرונות
public Train(int maxSpeed, int numOfCarriages) : base("land", "tracks", maxSpeed)
```

{

```
    this.numOfCarriages = numOfCarriages;
```

}

```
public void IncNumOfCarriages(int n) // מגדילה ב- n את מספר الكرונות ברכבת
{
```

```
    this.numOfCarriages = this.numOfCarriages + n;
```

}

}

(שים לב: המשך השאלה בעמודים הבאים.)

public class Boat : Vehicle

{

```
public Boat(string way, int maxSpeed) : base("water", way, maxSpeed)
```

{

}

}

public class Airplane : Vehicle

{

private int maxHeight; // גובה טישה מקסימלי

```
public Airplane(int maxSpeed, int maxHeight) : base("sky", "air", maxSpeed)
```

{

```
this.maxLength = maxLength;
```

}

}

public class **TransportationCompany**

{

```
private Vehicle[] vehicles = new Vehicle[50]; // מערך כל הרכבים בחברה
```

```
private int counter = 0; // מספר כל התחבורה שיש בפועל
```

public TransportationCompany()

{

}

```
מוסיפה כלי תחבורה למערך כלי התחבורה של החברה. הנח שיש מקום להוסיף כלי תחבורה. //
```

```
this.vehicles[counter] = v;
```

```
this.counter++;
```

}

}

(שים לב: המשך השאלה בעמוד הבא).

א. ממש ב- C# מחלקה ראשית Program ובה פעולה ראשית, שתבצע את המשימות האלה:

TransportationCompany – **i** בנייתו של עצם מטיפוס חברת כלית תחבורה –

. company1
הנקרא

. company1 **ii** הוספה של סירה אחת ורכבת אחת לחברת 1

בחר לתוכנות ערכיהם כרצונו.

ב. במחלקה **TransportationCompany** הוגדרה הפעולה:

```
public void Display()
{
    for (int i=0; i<this.counter; i++)
    {
        Console.WriteLine((i+1) + ":" + this.vehicles[i]);
    }
}
```

ממש ב- C# פעולות שיאפשרו ביצוע תיקין של הפעולה () , כך שבמעבר כל כלית תחבורה יודפסו כל התוכנות שלו. הגדר את הפעולות באופן המתאים ביותר לעקרונות של תכונות מונחה עצמים (הכמה – encapsulation , הורשה – inheritance – polymorphism –).

בעבור כל פעולה שאתה ממש, רשום לאיזו מחלקה היא שייכת.
ן לשנות את הפעולה () . Display

ג. ממש ב- C# פעולה, שתקבל מספר שלם n ותוסיף n קرونות לכל הרכבות ששויות לחברת שיש לה כלית תחבורה מסווגים שונים. תעוד את הפעולה, ורשום באיזו מחלקה יש להגדיר אותה. ן לשנות את הפעולות הקיימות בפרויקט.

.16. לפניך המחלקות **AA** ו- **BB**:

public class **AA**

{

 private string st;

 public AA() { this.st = "excellent"; }

 public AA(string st) { this.st = st; }

 public string GetSt() { return this.st; }

 public void SetSt (string st) { this.st = st; }

 public override string ToString() { return "st = " + this.st; }

}

public class **BB** : **AA**

{

 private int num;

 public BB() : base() { this.num = 1; }

 public BB(int num, string st) : base(st) { this.num = Math.Abs(num); }

 public int GetNum() { return this.num; }

 public void SetNum(int num) { this.num = num; }

 public override string ToString() { return base.ToString() + " num = " + this.num; }

}

.א. הגדר במחלקה **AA** פועלה בוליאנית הניתנת לדרישת, בשם (Object obj), המתקבל עצם obj מטיפוס **Object**. אם העצם obj הינו מטיפוס **AA** וגם תוכן המחרוזת st של obj זהה לתוכן המחרוזת st של העצם הנוכחי — הפועלה תחזיר `true`, אחרת — תחזיר `false`.

.ב. הגדר במחלקה **BB** פועלה הדורשת את הפועלה שהגדרת בסעיף א. אם העצם obj הינו מטיפוס **BB** וגם ערך התכונה num שלו זהה לערך התכונה num של העצם הנוכחי — הפועלה תחזיר `true`, אחרת — תחזיר `false`.

(שים לב: המשך השאלה בעמוד הבא).

.ג.

לפניך קטע מפעולה ראשית:

```
AA a = new AA("excellent");
BB b = new BB();
a = b;
if (a.IsLike(b)) Console.WriteLine(a);
```

האם קטע התכנית תקין?

אם כן — מה יהיה פלט הקטע? רשוםஇயோ גרסה של הפעולה IsLike תופעל — זו של AA או זו של BB.

אם לא — הסבר מהי השגיאה ומתי היא תתגלה: בזמן קומpileציה או בזמן ריצה.

.ד.

לפניך קטע מפעולה ראשית:

```
AA aa = new AA();
BB bb = new BB(2 , "excellent");
bb = aa;
if (bb.IsLike(aa)) Console.WriteLine(bb);
```

האם קטע התכנית תקין?

אם כן — מה יהיה פלט הקטע? רשוםஇயோ גרסה של הפעולה IsLike תופעל — זו של AA או זו של BB.

אם לא — הסבר מהי השגיאה ומתי היא תתגלה: בזמן קומpileציה או בזמן ריצה.

.ה.

כתב פוליה חיונית בשם LongString המקבלת מערך של עצמים מטיפוס Object.

הפעולה מחזירה מחרוזת המורכבת משרשרת התוכונה st של עצמים מטיפוס AA ב眯ירך, באופן זהה:

- אם לעצם יש בק התוכונה st , תשורשר המחרוזות שבתוכונה st פעם אחת.
- אם לעצם יש גם התוכונה משת , המחרוזות שבתוכונה st תשורשר מעת פעמיים.
- אם אין ב眯ירך אף עצם מטיפוס AA , תוחזר מחרוזת ריקה.

בצלחה!