

מדעי המחשב
פתרון בחינת הבטאות
פרק ראשון - (יסודות)

שאלה 1

דרך I

```
//--- פעולה המחזירה את הציון הממוצע ---  
public double average()  
{  
    return (arrTest[0] + arrTest[1] + arrTest[2]) / 3.0;  
}
```

Java - II דרך

```
//--- פעולה המחזירה את הציון הממוצע ---  
public double average()  
{  
    int sum = 0;  
    for (int i = 0 ; i < arrTest.length ; i++)  
        sum = sum + arrTest[i];  
  
    return (double) sum / arrTest.length;  
}
```

C# - II דרך

```
//--- פעולה המחזירה את הציון הממוצע ---  
public double Average()  
{  
    int sum = 0;  
    for (int i = 0; i < arrTest.Length; i++)  
        sum = sum + arrTest[i];  
  
    return (double)sum / arrTest.Length;  
}
```

אלף 2

Java

א

```
//--- פעולה בונה לשחקן ---
public Actor(String id, String gender, int numFilms)
{
    this.id = id;
    this.gender = gender;
    this.numFilms = numFilms;
}
/ }
```

ב.

```
//--- פעולה המשווה בין מספר הסרטים ---
//--- של השחקן הנוכחי והשחקן האחר ---
public int compare (Actor other)
{
    if (this.numFilms > other.numFilms)
        return 1;
    if (this.numFilms < other.numFilms)
        return 2;
    return 3;
}
```

ג.

```
//--- ט. כניסה: מערך שחקנים ומספר שלם num ---
//--- ט. יציאה: מודפס מספר השחקנים ששיחקו ביותר מ- num סרטים ---
public static void busyActors (Actor [] arr, int num)
{
    int count = 0;
    for (int i = 0 ; i < arr.length ; i++)
        if (arr[i] != null && arr[i].getNumFilms() > num)
            count ++;

    System.out.println("count = " + count);
}
```

C#

א.

```
//--- פעולה בונה לשחקן ---  
public Actor(string id, string gender, int numFilms)  
{  
    this.id = id;  
    this.gender = gender;  
    this.numFilms = numFilms;  
}
```

ב.

```
//--- פעולה המשווה בין מספר הסרטים ---  
//--- של השחקן הנוכחי והשחקן האחר ---  
public int Compare(Actor other)  
{  
    if (this.numFilms > other.numFilms)  
        return 1;  
    if (this.numFilms < other.numFilms)  
        return 2;  
    return 3;  
}
```

ג.

```
//--- ט. כניסה: מערך שחקנים ומספר שלם num ---  
//--- ט. יציאה: מודפס מספר השחקנים ששיחקו ביותר מ- num סרטים ---  
public static void busyActors(Actor[] arr, int num)  
{  
    int count = 0;  
    for (int i = 0 ; i < arr.Length ; i++)  
        if (arr[i] != null && arr[i].GetNumFilms() > num)  
            count ++;  
  
    Console.WriteLine("count = " + count);  
}
```

עבודה 3

Java

```
//--- א. פעולה בונה לתכנית טלוויזיה ---
public TvProgram(int code, int day, boolean isSport)
{
    this.code = code;
    this.day = day;
    this.isSport = isSport;
}
```

ב.

```
public class TvWeek
{
    //--- תכונות ---
    public static int maxProg = 100;    // גודל המערך
    private TvProgram [] arrProg;      // מערך התכניות
    private int current;               // מספר התכניות בפועל

    //--- הוספת תכנית למערך התכניות ---
    public void add (TvProgram tvProg)
    {
        this.arrProg[this.current] = tvProg;
        this.current ++;
    }

    //--- מספר תכניות הספורט בשבוע הנוכחי ---
    public int countSportProgram ()
    {
        int count = 0;
        for (int i = 0 ; i < this.current ; i++)
            if (this.arrProg[i].isSport())
                count ++;

        return count;
    }
}
```

C#

```
//--- א. פעולה בונה לתכנית טלוויזיה ---  
public TvProgram(int code, int day, bool isSport)  
{  
    this.code = code;  
    this.day = day;  
    this.isSport = isSport;  
}
```

ב.

```
class TvWeek  
{  
    //--- תכונות ---  
    public static int maxProg = 100;    // גודל המערך  
    private TvProgram[] arrProg;        // מערך התכניות  
    private int current;                 // מספר התכניות בפועל  
  
    //--- הוספת תכנית למערך התכניות ---  
    public void Add(TvProgram tvProg)  
    {  
        this.arrProg[this.current] = tvProg;  
        this.current++;  
    }  
  
    //--- מספר תכניות הספורט בשבוע הנוכחי ---  
    public int CountSportProgram()  
    {  
        int count = 0;  
        for (int i = 0; i < this.current; i++)  
            if (this.arrProg[i].IsSport())  
                count++;  
  
        return count;  
    }  
}
```

פרק שני - (מבני נתונים)

Java

אלף 4

.א

```

//--- פעולה המסדרת את הטבעות על המוט ---
//--- כך שכל הטבעות הגדולות בתחתית המוט ---
//--- וכל הטבעות הקטנות מעליהן ---
public void sort()
{
    Pole pL = new Pole(); // מוט עזר לטבעות הגדולות
    Pole pS = new Pole(); // מוט עזר לטבעות הקטנות
    Ring ring;

    //--- העברת הטבעות למוטות העזר לפי הגודל ---
    while (! this.isEmpty())
    {
        ring = this.remove();
        if (ring.getSize().equals("L"))
            pL.add(ring);
        else
            pS.add(ring);
    }

    //--- החזרת הטבעות הגדולות ---
    while (! pL.isEmpty())
        this.add(pL.remove());

    //--- החזרת הטבעות הקטנות ---
    while (! pS.isEmpty())
        this.add(pS.remove());
}

```

ב. סיבוכיות זמן הריצה היא $O(n)$.נימוק: העברת n טבעות מהמוט למוטות העזר: $O(1) * n$ סה"כ n צעדיםהחזרת n טבעות משני מוטות העזר: $O(1) * n$ סה"כ n צעדיםסה"כ: $f(n) = 2n \Rightarrow O(n)$

C#

אלף 4

.א

```

//--- פעולה המסדרת את הטבעות על המוט ---
//--- כך שכל הטבעות הגדולות בתחתית המוט ---
//--- וכל הטבעות הקטנות מעליהן ---
public void Sort()
{
    Pole pL = new Pole(); // מוט עזר לטבעות הגדולות
    Pole pS = new Pole(); // מוט עזר לטבעות הקטנות
    Ring ring;

    //--- העברת הטבעות למוטות העזר לפי הגודל ---
    while (!this.IsEmpty())
    {
        ring = this.Remove();
        if (ring.GetSize().Equals("L"))
            pL.Add(ring);
        else
            pS.Add(ring);
    }

    //--- החזרת הטבעות הגדולות ---
    while (!pL.IsEmpty())
        this.Add(pL.Remove());

    //--- החזרת הטבעות הקטנות ---
    while (!pS.IsEmpty())
        this.Add(pS.Remove());
}

```

ב. סיבוכיות זמן הריצה היא $O(n)$.

נימוק: העברת n טבעות מהמוט למוטות העזר: $O(1) * n$ שה"כ n צעדים

החזרת n טבעות משני מוטות העזר: $O(1) * n$ שה"כ n צעדים

סה"כ: $O(n) \Rightarrow f(n) = 2n$

שאלה 5

Java:

```

//--- א. השלמת הפעולה ---
//--- פעולה המחזירה הפנייה לחוליה השמאלית ביותר ברשימה הדו-כיוונית ---
public static BinNode<Integer> firstLeft(BinNode<Integer> pos)
{
    while (pos.hasLeft()) // while (pos.getLeft() != null)
        pos = pos.getLeft();
    return pos;
}

```

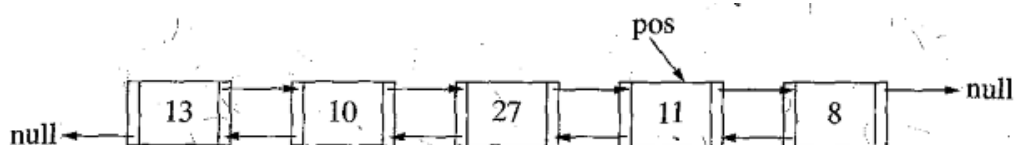
C#:

```

//--- א. השלמת הפעולה ---
//--- פעולה המחזירה הפנייה לחוליה השמאלית ביותר ברשימה הדו-כיוונית ---
public static BinNode<int> FirstLeft(BinNode<int> pos)
{
    while (pos.HasLeft()) // while (pos.GetLeft() != null)
        pos = pos.GetLeft();
    return pos;
}

```

ב. (1) טבלת מעקב:



| left value | right value | sum | I left \neq right | II left.getRight \neq right | III left.val + right.val == sum | I && II && III | left == right | left.get Right == sum | ערך מוחזר |
|------------|-------------|-----|------------------------|----------------------------------|------------------------------------|----------------|---------------|-----------------------|-----------|
| 13 | 8 | 21 | T | T | T | true | | | |
| 10 | 11 | | T | T | T | true | | | |
| 27 | 27 | | F | | | false | true | | false |

(2) ניתן להחליף את התנאי.

נימוק: הפעולה מחזירה אמת אם סכום כל זוגות איברים הנמצאים במרחק שווה מקצות הרשימה שווים (סימטרי-סכומים).

הלולאה עוצרת כאשר הגיעה לאמצע הרשימה או כאשר נמצא זוג איברים שסכומם שונה.

עבור רשימה באורך אי-זוגי, מוחזר אמת אם האיבר האמצעי שווה לסכום זה, אחרת מוחזר שקר. בכל מקרה אחר מוחזר אמת אם סכום שני האיברים במקומות left ו-right שווה ל-sum, אחרת מוחזר שקר.

ולכן, ניתן להחליף את הקטע המסומן בהוראה:

"החזר: סכום האיברים במקומות left ו-right שווה ל-sum"

תרגיל 6

דרך I

האם-מסלול-עולה? (tr)

```

    }
    פעולה המחזירה אמת אם קיים בעץ מסלול המתחיל בשורש העץ ומסתיים באחד העלים, וערך
    הצמתים ממין בסדר עולה, ושקר אחרת.
    {
        הנחה: העץ לא לא ריק שערכיו מספרים שלמים השונים זה מזה.

        אם (tr == null) החזר שקר // מסלול לא יכול להסתיים ב-null
        אם (עלה? (tr)) החזר אמת // בדיקת המסלול הסתיימה בהצלחה

        אם (יש ל-tr בן שמאלי שערכו גדול מערך הצומת של tr)
            אמת ← left
            אחרת שקר ← left

        אם (יש ל-tr בן ימני שערכו גדול מערך הצומת של tr)
            אמת ← right
            אחרת שקר ← right

        אם (אחד הצדדים תקין)
            החזר left וגם האם-מסלול-עולה? (בן שמאלי של tr) או
            right וגם האם-מסלול-עולה? (בן ימני של tr)

        החזר שקר // אף אחד מהבנים לא נמצא במסלול ממין

```

יש לממש גם את הפעולה עלה? (tr)

```

//--- האם עלה? ---
public static boolean isLeaf(BinNode<Integer> tr)
{
    if (tr == null) return false;
    return ! tr.hasLeft() && ! tr.hasRight();
}

//--- הפעולה מחזירה אמת אם קיים מסלול עולה המתחיל בשורש ---
//--- ומסתיים באחד העלים, ושקר אחרת ---
public static boolean upPath (BinNode<Integer> tr)
{
    if (tr == null) return false;
    if (isLeaf(tr)) return true;

    boolean left = tr.hasLeft() && tr.getLeft().getValue() > tr.getValue();
    boolean right = tr.hasRight() && tr.getRight().getValue() > tr.getValue();

    if (left || right)
        return left && upPath (tr.getLeft()) ||
            right && upPath (tr.getRight());
    return false;
}

```

דרך II

פתרון של: אביטל Evi גרינולד

תנאי קדם: הפעולה מקבלת עץ בינרי של מספרים שלמים טבעיים (שלמים גדולים מ 0) שונים זה מזה
 תנאי בתר: הפעולה מחזירה 'אמת' אם קיים מסלול עולה משורש עד עלה, שקר – אחרת.

```
public static boolean upPath(BinNode<Integer> tr)
{
    if (tr == null) return false;
    if (tr.getLeft() == tr.getRight()) return true;
    return upPath(tr.getLeft(), tr.getValue()) || upPath(tr.getRight(), tr.getValue());
}
```

תנאי קדם: הפעולה מקבלת עץ בינרי של מספרים שלמים טבעיים (שלמים גדולים מ 0) שונים זה מזה
 ומספר טבעי x

תנאי בתר: הפעולה מחזירה 'אמת' אם קיים מסלול עולה משורש העץ עד עלה שערך השרש שלו גדול מ x, שקר – אחרת.

```
private static boolean upPath(BinNode<Integer> tr, int x)
{
    if (tr == null) return false;
    if (tr.getValue() <= x) return false;
    if (!tr.hasLeft() && !tr.hasRight()) return true;
    return upPath(tr.getLeft(), tr.getValue()) || upPath(tr.getRight(), tr.getValue());
}
```

פרק שלישי

מערכות מחשב ואסמבלר

הפתרון לפרק זה נכתב ע"י: רונית (מרציאנו) גל-אור

תרגיל 7

א.

```

tav db ?
; count bits in tav
count:
    mov al,tav
    mov cl,0
again:
    shr al,1
    adc cl,0
    cmp al,0
    jne again
continue:
    mov ch,cl ; keep cl
; check if Even number
    and ch,1
    jz even
; not even
    rol tav,cl
    jmp sof
even:
    ror tav,cl
sof: nop

```

ב.

(1) לא נכון

הפקודה `div bx` מבצעת חילוק של `dx ax` ב `bx` ולכן אם ב `dx` יש ערך השונה מ 0, התשובה ב `ax` לא תהיה 4.

```

mov ax,8
mov bx,2
div bx

```

(2) לא נכון בסיס הקטע `al=1` לא ידוע מה יש ב `ah`

```

mov al,56
add al,200
jz stop
inc al
stop:

```

(3) לא נכון בסים הקטע ax=3000h

| | | | | | | | | |
|-------|----|----|----|----|----|----|----|----|
| תוכן | 01 | 00 | 02 | 00 | 03 | 00 | 04 | 00 |
| כתובת | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 |

ax=3000h

```
array dw 1,2,3,4
mov bx,array
add bx,2
mov ax,[bx]
```

(4) נכון

```
mov cx,3
mov ax,1
do:
shl ax,1
loop do
```

(5) נכון

```
mov ax,11000001b
mov bx,01000001b
```

מכיוון שהנתון בשאלה הוא בית והוא נכנס ל אוגר מילה, תמיד ההשוואה ביניהם תהיה ש ax גדול יותר

(6) נכון

```
or al,3
```

or אם ביט 1 תמיד ייתן 1 לא משנה אם הביט המקביל הוא 0 או 1

תרגיל 8

א.

| cx | | ax | | si |
|-----|-----|-----|-----|-------|
| ch | cl | ah | al | |
| 00h | 04h | 45h | 45h | 0000h |
| | 04h | 04h | 05h | 0001h |
| 00h | 04h | 00h | 14h | 0002h |
| 00h | 03h | 26h | 26h | 0003h |
| | 04h | 02h | 06h | 0004h |
| 00h | 03h | 00h | 0ch | |
| 00h | 02h | 32h | 32h | |
| | 04h | 03h | 02h | |
| 00h | 02h | 00h | 06h | |
| 00h | 01h | 82h | 82h | |
| | 04h | 08h | 02h | |
| 00h | 01h | 00h | 10h | |
| 00h | 00h | | | |

| | | | | | |
|-------------|------|-----|-----|-----|-----|
| sp | 100h | | | | |
| | 0ffh | 00h | 00h | 00h | 00h |
| sp sp sp sp | 0feh | 04h | 03h | 02h | 01h |
| | 0fdh | | | | |
| | 0fch | | | | |
| | 0fbh | | | | |
| | 0fah | | | | |
| | 0f0h | | | | |
| | 0f9h | | | | |

תמונת הזיכרון

vec1 db 45h,26h,32h,82h
vec2 db 0,0,0,0

לפני

vec1 db 45h,26h,32h,82h
vec2 db 14h,0ch,6h,10h

אחרי

(2) הקטע מכפיל את ספרת העשרות באחדות ואת התוצאה בבסיס 16 שם במערך השני בתאים מקבילים.

(3) אם נחליף את את הפקודה: mul ah בפקודה imul ah לא יהיה שינוי בתוצאה, אבל, גם ה cf וגם ב of ידלקו, למשל, אם מכפילים בתא במערך vec1 והיה הערך 0ffh והיינו מבצעים כפל של 0fh * 0fh, התוצאה לא נכנסת בבית במספרים מכוונים ולכן נדלקים cf כי התוצאה לא נכנסת בייצוג ו of כי הראה גלישה במספרים מכוונים.
התוצאה שנראה במערך זה, מכיוון ש 0e1(h) – 225(10) אמנם נכנס בבית אבל רק במספרים לא מכוונים.

ב. באוגר al מאוחסן המספר 2 ובאוגר bl מאוחסן המספר 5.
יש לאחסן באוגר dx את סכום המספרים מ al עד bl כולל.

קטע 1

לא מבצע

```
mov al,2
mov bl,5

mov dx,0
mov ah,0
mov cl,al
sub cl,bl

again:
add dx,ax
inc al
loope again
nop
```

| cx | | dx | | ax | | bx | |
|----|------|-----|-----|-----|-----|----|-----|
| ch | cl | dh | dl | ah | al | bh | bl |
| | 02h | 00h | 00h | 00h | 02h | | 05h |
| | 0fdh | 00h | 02h | | 03h | | |
| | 0fch | | | | | | |

ואז יוצא מהלולאה בגלל ה loope
בנוסף שימו לב לערך של cx

קטע 2

מבצע

```
mov al,2
mov bl,5

xor dx,dx
mov bh,0

again:
add dx,bx
dec bl
cmp bl,al
jge again
nop
```

קטע 3

לא מבצע

```
mov al,2
mov bl,5

xor dx,dx
xor ax,ax
mov bx,0

again:
add dx,ax
add ax,1
cmp ax,bx
jl again
nop
```

al ו-bl מתאפסים ישר בהתחלה

מבוא לחקר ביצועים

עלה 9

עלה 10

מודלים חישוביים
הפתרון לפרק זה נכתב ע"י רחל לודמר.

תרגיל 11

$$L = L_1 \cdot L_2 \cup \overline{L_1} \cdot \overline{L_2}$$

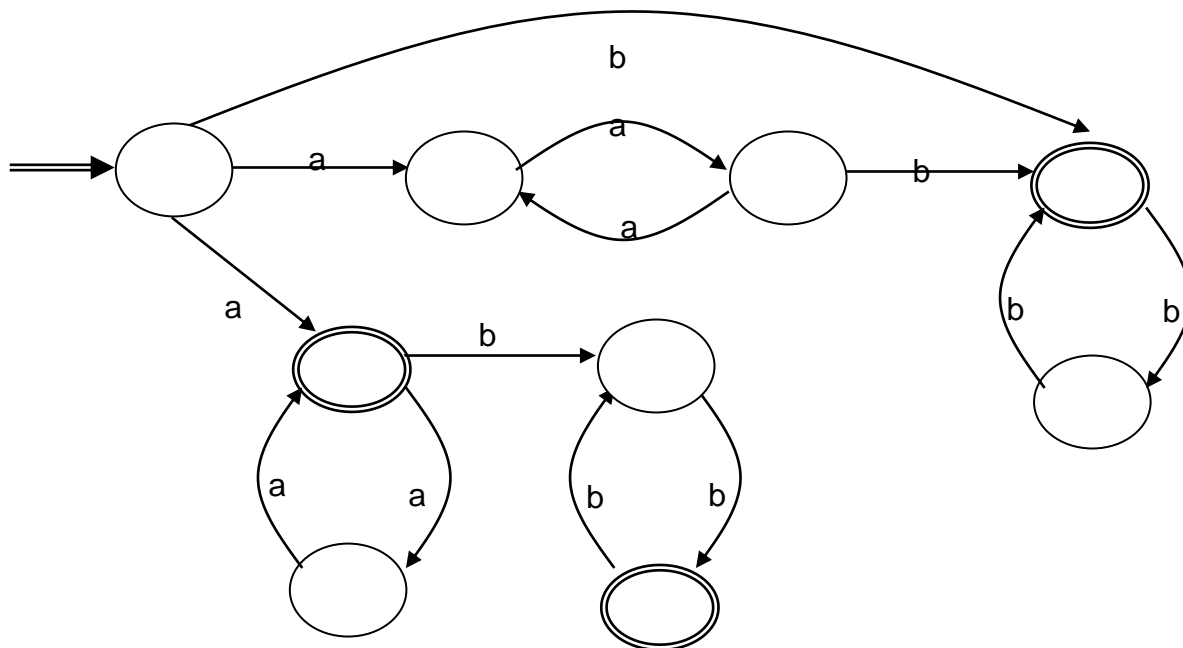
א.

כאשר:

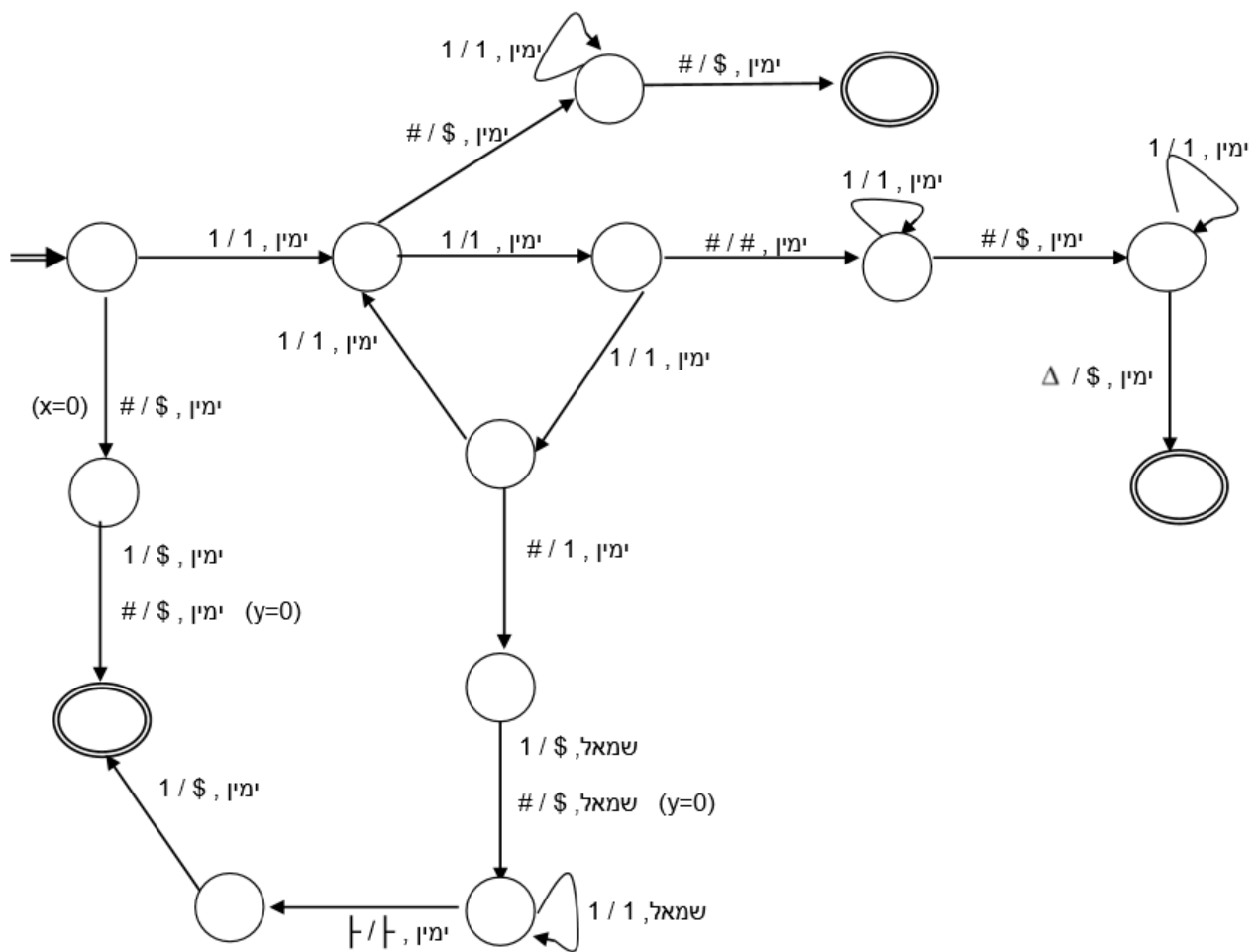
$$\overline{L_1} = \{a^k \mid k \% 2 = 1, k > 0\}$$

$$\overline{L_2} = \{b^k \mid k \% 2 = 0, k \geq 0\}$$

ב.



שאלה 12



תכנות מונחה עצמים Java

תרגיל 13

- א. (1) שגיאת קומפילציה (הידור). הפעולה foo() אינה מוגדרת עבור עצם מהטיפוס Object
- (2) שגיאת קומפילציה (הידור). הפעולה foo() אינה מוגדרת עבור עצם מהטיפוס B
- (3) תקין
- (4) שגיאת קומפילציה (הידור). הפעולה foo() אינה מוגדרת עבור עצם מהטיפוס Object

ב.

```

public class Artist
{
    //--- תכונות ---
    protected String name1;
    private String name2;
    protected double sal;
    public static int num = 2;

    //--- פעולות בונות ---
    public Artist (double sal, String name) {}

    public Artist (double sal, String name1, String name2) {}

    //--- פעולות נוספות ---
    public int train() {}

    public double price() {}
    public double calc (double x) {}
    public void print() {}
}

```

תרגיל 14

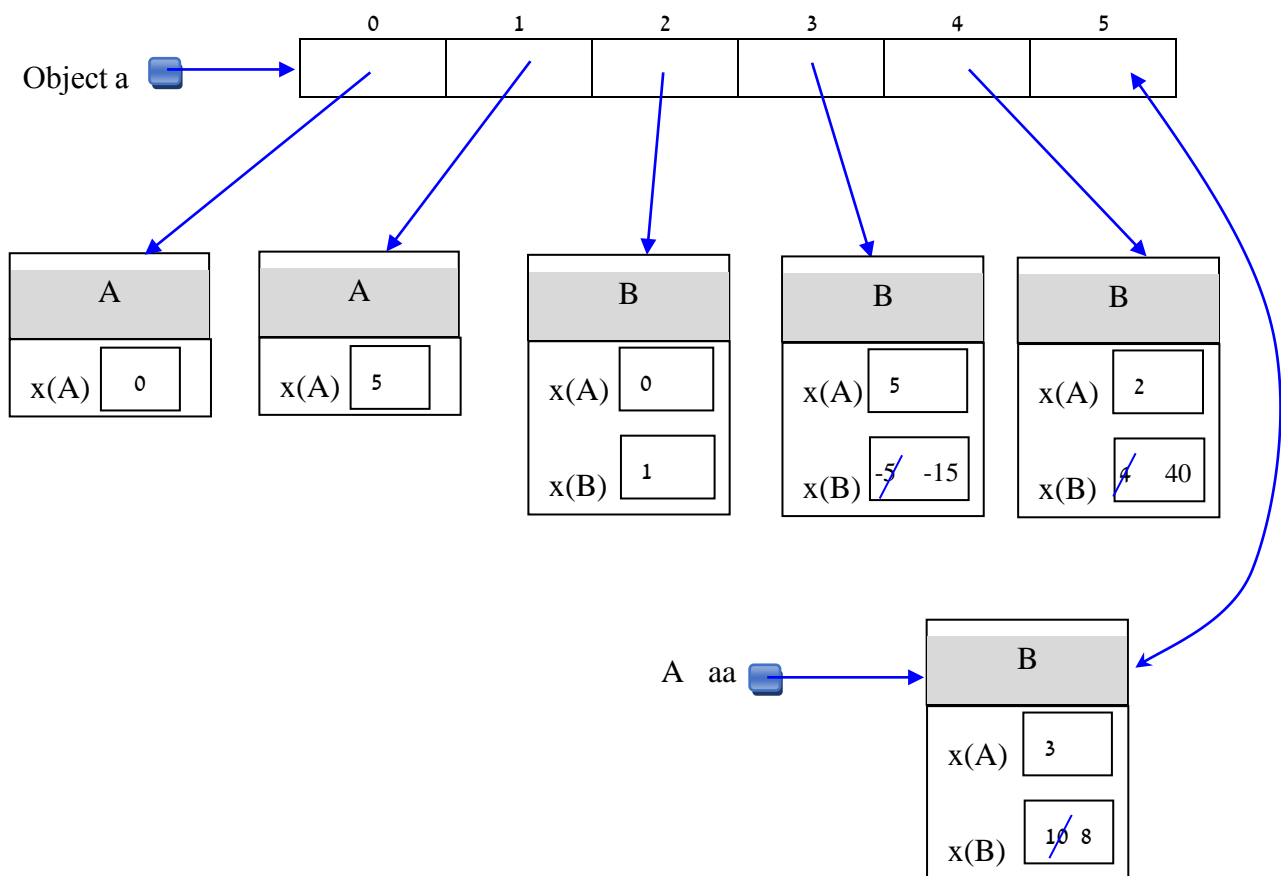
```

A a1 = new B (1, 20);
Object obj = a1;
B b1 = (B)a1;    // תיקון: המרה ל-B
A a2 = a1;

```

א.

ב.



```
//--- (i) ---
a2.doubleX ();
System.out.println(a2);
```

(1) ג.

```
// תקין
// xA = 198 xB = 99; : הפלט
```

(2)

```
//--- (ii) ---
a2.tenTimesX(); // שגיאת קומפילציה. פעולה לא מוגדרת עבור הפניה מסוג זה
System.out.println(a2.tenTimesX()); // שגיאה: void tenTimesX() מחזירה
```

(3)

```
//--- (iii) ---
if (a2 instanceof B)
{
    a2.tenTimesX(); //--- שגיאת קומפילציה. יש להמיר לטיפוס העצם
    System.out.println(a2);
}
```

(4)

```
//--- (iv) ---
((B)a1).tenTimesX();
System.out.println(a1);
//--- לא תקין. העצם לא מהטיפוס הנכון
//--- שגיאת זמן ריצה. השגיאה מתגלה בעת ניסיון ההמרה
```

(5)

```
//--- (v) ---
a2.calc();
System.out.println(a2);
//--- תקין
// xA = 2 xB = 97; : הפלט
```

```
//--- (vi) ---
B bb = (B) a2;
System.out.println(bb.baseX());
```

(6)

```
//--- תקין
// 2 : הפלט
```

תכנות מונחה עצמים C#

הפתרון לפרק זה נכתב ע"י דיתה אוהב-ציון.

אלה 15

- א. 1 שגיאת הידור. העצם g מטיפוס המחלקה Object שלא מכילה ולא מכירה את הפעולה Foo().
(מחלקת אב לא מכירה פעולות שיש בבנים שלה)
- א. 2 שגיאת הידור. העצם g מטיפוס המחלקה B שלא מכילה ולא מכירה את הפעולה Foo().
(מחלקת אב לא מכירה פעולות שיש בבנים שלה)
- א. 3 תקין.
- א. 4 שגיאת הידור. העצם g מטיפוס המחלקה Object שלא מכילה ולא מכירה את הפעולה Foo().
(מחלקת אב לא מכירה פעולות שיש בבנים שלה)

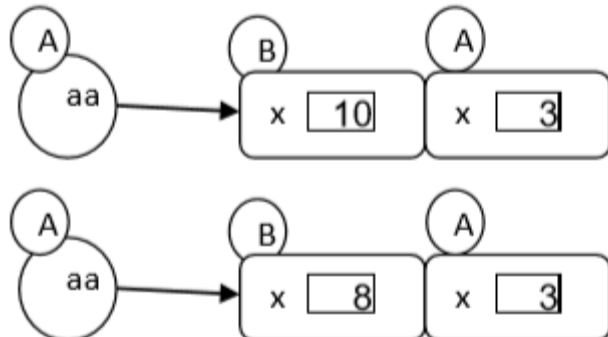
ב. המחלקה Artist

| | |
|--|---|
| <pre>public class Artist { protected string name; protected double sal; protected static int num; public Artist(double sal, string name) { } public Artist(double sal, string name, string name1){} public double Price() { } public virtual double Calc(double d) { } public virtual void Print() { } public int Train() { } }</pre> | <p>משתנה סטטי על פי הקריאה Artist.num</p> <p>בנאי המתאים לקריאה עם שתי מחרוזות</p> <p>הפעולה Value() מפעילה אותו</p> <p>על פי ההקריאה base.Calc() מאחר וב- Singer היא override, הרי באב היא וירטואלית</p> <p>על פי ההקריאה base.Print() מאחר וב- Singer היא override, הרי באב היא וירטואלית</p> <p>מאחר ואין הגדרה לפעולה כמימוש מהממשק ב- Singer הרי היא מוגדרת באב.</p> |
|--|---|

שאלה 16

א. ההוראה השלישית $B\ b1 = a1$; שגויה. עצם מטיפוס מחלקה יורשת לא יכול להחזיק עצם ממחלקת האב.
יש לבצע המרה $B\ b1 = (B)a1$;

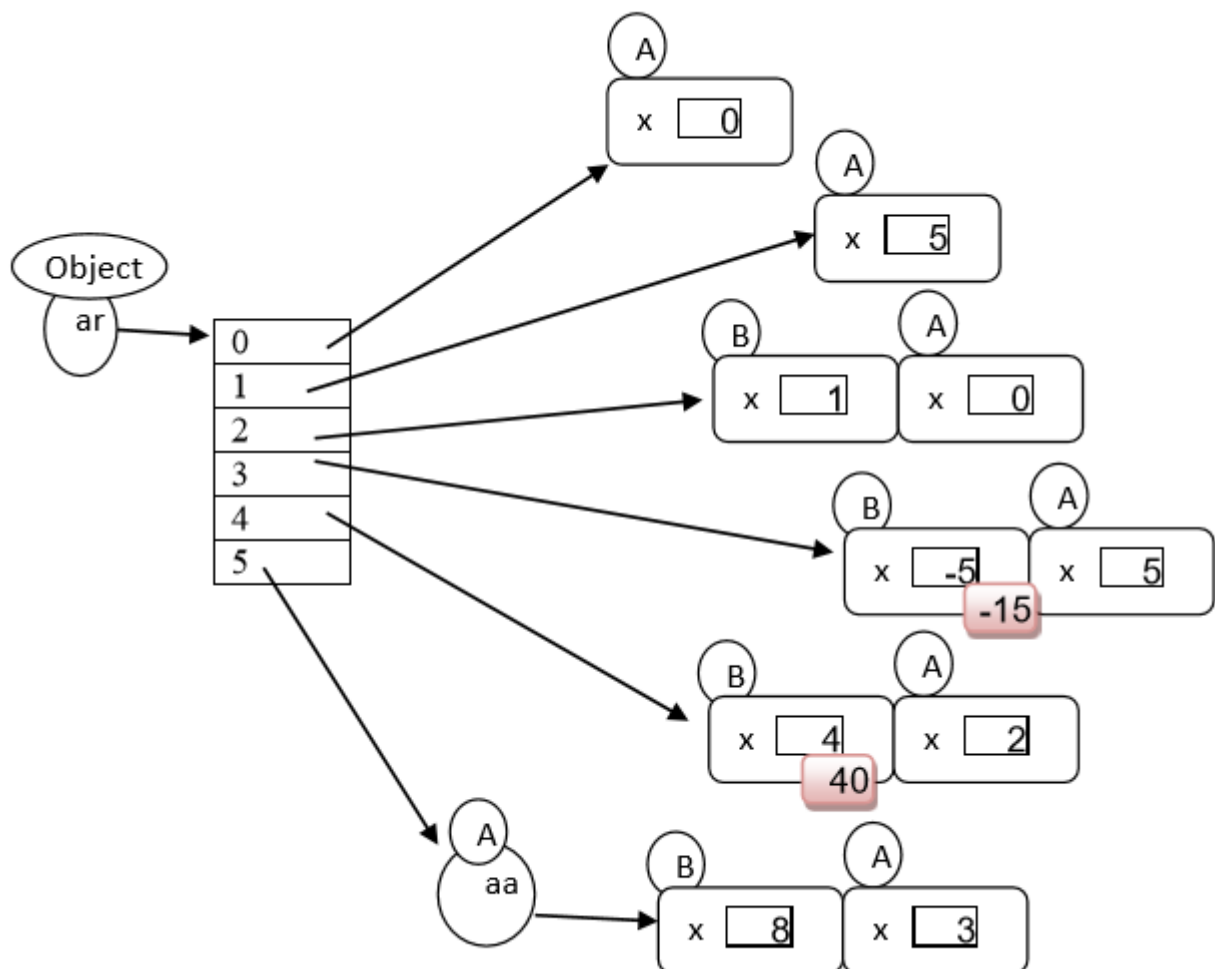
ב. $A\ aa = \text{new } B(3, 10)$;



הפעולה $aa.Sub()$ תפעיל את הפעולה מ-B הדורסת לכן הערך ישתנה ל-8

השורה $((A)ar[3]).TripleX()$; תשנה את הערך $B.x = -15$
(הפעולות המופעלות הן הפעולות הדורסות של B).

השורה $((B)ar[4]).TenTimesX()$; תשנה את הערך $B.x = 40$



ג.

1. תקין . הפלט : $xA=198$ $xB=99$;2. `a2.TenTimesX();`

לא תקין - שגיאת הידור.

הפניה מטיפוס האב לא מכירה פעולות ממחלקות יורשות. תיקון אפשרי : `((B)a2).TenTimesX();`3. אותה שגיאה . ה- `if` לא משנה.4. `((B)a1).TenTimesX();`

שגיאת זמן ריצה. הפניה מטיפוס האב המחזיקה עצם מטיפוס אב לא מכירה את היורשים ממנה.

5. תקין. הפלט : $xA=2$ $xB=97$;

6. תקין. הפלט : 2.