

מדעי המחשב

פרק ראשון

שאלה 1

Java

```
//---          נכתב ע"י רמי דייך          ---  
//---          פעולה המחזירה את מספר התא הראשון במערך ---  
//---          שסכום המספרים עד אליו גדול מהמספר שהתקבל ---  
public static int above(int[] arr, int num)  
{  
    int sum = 0;  
    for (int i = 0; i < arr.length; i++)  
    {  
        sum += arr[i];  
        if(sum > num)  
            return i;  
    }  
    return -1;  
}
```

C#

הפתרון לשאלה זו נכתב ע"י ראמי ג'בלי

```
//---          פעולה המחזירה את מספר התא הראשון במערך ---  
//---          שסכום המספרים עד אליו גדול מהמספר שהתקבל ---  
public static int Above(int[] arr, int num)  
{  
    int sum = 0, index=-1;  
    for (int i = 0; i < arr.Length; i++)  
    {  
        sum = sum + arr[i];  
        if (sum > num)  
        {  
            index = i;  
            return index;  
        }  
    }  
    return index;  
}
```

Java

```
//--- נכתב ע"י רמי דייגן ---
//--- א. בנאי המקבל את המשקל בק"ג ובגורם ---
public Weight(int kilo, int gram)
{
    this.kilo = kilo;
    this.gram = gram;
}

//--- א. בנאי המקבל את המשקל בק"ג ---
public Weight(int totalGram)
{
    this.kilo = totalGram/1000;
    this.gram = totalGram % 1000;
}

//--- ב. הוספת משקל ---
public void add(Weight other)
{
    int gr = this.gram + other.gram;
    this.gram += gr % 1000;
    this.kilo += other.kilo + gr % 1000;
}

//--- נכתב ע"י רמי דייגן ---
//--- ב. הוספת משקל ---
public void add(Weight other)
{
    if (this.gram + other.gram > 1000)
    {
        this.kilo += other.kilo + 1;
        this.gram = (this.gram + other.gram) % 1000;
    }
    else
    {
        this.kilo += other.kilo;
        this.gram += other.gram;
    }
}
}
```

```

//---          נכתב ע"י רמי דיון          ---
//---          ב. פעולה המחזירה אמת אם המשקל הנוכחי ---
//---          קטן מהמשקל האחר ושקר אחרת ---
public boolean less(Weight other)
{
    if(this.kilo < other.kilo)
        return true;
    if(this.kilo == other.kilo && this.gram < other.gram)
        return true;
    return false;
}

//---          סעיף ג' - ירד במיקוד          ---
//---          הפתרון נכתב ע"י רמי דיון          ---
//---          פעולה המחזירה עצם חדש המכיל את סכום איברי המערך ---
public Weight sum()
{
    Weight w = new Weight(); // יצירת העצם המוחזר

    //---          עדכון משקל העצם בסכום המשקלים ---
    for (int i = 0; i < this.arr.length; i++)
    {
        w.add(this.arr[i]);
    }
    return w;
}

```

C#

הפתרון לשאלה זו נכתב ע"י ראמי ג'בלי

```
// א. בנאי המקבל קילו וגרם
public Weight(int kilo, int gram)
{
    this.kilo = kilo;
    this.gram = gram;
}

// א. בנאי מקבל מספר הגרמים
public Weight(int totalGram)
{
    this.kilo = totalGram / 1000;
    this.gram = totalGram % 1000;
}

// ב. פעולה מקבלת עצם Weight ומוסיפה אותו את ערכיו חמשקל של העצם הנוכחי
public void Add(Weight other)
{
    int x = this.gram + other.gram;
    this.gram = x % 1000;
    this.kilo = other.kilo + x % 1000;
}

// ב. פעולה מקבלת עצם Weight ומחזירה אמת אם משקל העצם הנוכחי
// קטן מן משקל האחר אחרת מחזירה שקר
public bool Less(Weight other)
{
    if (this.kilo < other.kilo)
        return true;
    if (this.kilo == other.kilo && this.gram < other.gram)
        return true;
    return false;
}
```

```
// ג. פעולה מחזירה סכום המשקלים  
// לא נדרש בבחינה  
public Weight Sum()  
{  
    int s1 = 0;  
    int s2 = 0;  
    for (int i = 0; i < arr.Length; i++)  
    {  
        s1 = s1 + arr[i].GetKilo();  
        s2 = s2 + arr[i].GetGram();  
    }  
  
    int x1 = s2 / 1000;  
    int x2 = s2 % 1000;  
  
    return new Weight(s1+x1, s2+x2);  
}
```

Java

```
//---          הפתרון נכתב ע"י רמי דייב          ---
//--- א. פעולה המחזירה את ההכנסה שהתקבלה מהחדר (נכתב במחלקה Room) ---
public int income()
{
    int price = 0;
    if(this.roomType == 1)
        price = 50;
    else
        price = 100;
    return this.nightsReserved * price;
}

//---          הפתרון נכתב ע"י רמי דייב          ---
//--- ב. (1) פעולה המוצאת את החדר הראשון העונה לתנאים ומעדכנת את פרטיו ---
//--- אם נמצא החדר יוחזר מספרו ואין כזה יוחזר -1 (נכתב במחלקה Hostel) ---
public int orderRoom(int type, int nights)
{
    for (int i = 0; i < this.allRooms.length; i++)
    {
        if(this.allRooms[i].getRoomType() == type && this.allRooms[i].getNightsReserved() > 0)
        {
            this.allRooms[i].setNightsReserved(nights);
            return this.allRooms[i].getRoomNum();
        }
    }
    return -1;
}

//--- ב. (2) פעולה המחזירה מערך ובו סך ההכנסות מהחדרים בכל קומה (נכתב במחלקה Hostel) ---
public int[] floorIncome()
{
    int[] arr = new int[3];
    for (int i = 0; i < this.allRooms.length; i++)
    {
        if(this.allRooms[i].getRoomNum()/100 == 1)
            arr[0] += this.allRooms[i].income();
        else{
            if(this.allRooms[i].getRoomNum()/100 == 2)
                arr[1] += this.allRooms[i].income();
            else
                arr[2] += this.allRooms[i].income();
        }
    }
    return arr;
}
..
```

C#

הפתרון לשאלה זו נכתב ע"י ראמי ג'בלי

```
// א. נכתב במחלקה Room
// פעולה מחזירה את ההכנסה
public int Income()
{
    if (roomType == 1)
        return 50 * nightsReserved;
    else return 100 * nightsReserved;
}

// ב. נכתב במחלקה Hostel
// פעולה מוצאת חדר הראשון הפנוי וגם מסוג type ומעדכנת הנתונים
public int OrderRoom(int type, int night)
{
    int place=-1;
    for (int i = 0; i < arr.Length; i++)
        if (arr[i].GetRoomType() == type && arr[i].GetNightReserved() == 0)
        {
            place = i;
            arr[place].SetNightReserved(night);
            return arr[place].GetRoom();
        }
    return place;
}

// פעולה מחזירה סכום ההכנסות מכל החדרים לתוך מערך
public int[] FloorINcome()
{
    int[] arrIncome = new int[3];
    for (int i = 0; i < arr.Length; i++)
    {
        if (arr[i].GetRoom() / 100 == 1)
            arrIncome[0] = arrIncome[0] + arr[i].Income();
        else if(arr[i].GetRoom() / 100 == 2)
            arrIncome[1] = arrIncome[1] + arr[i].Income();
        else arrIncome[2] = arrIncome[2] + arr[i].Income();
    }
    return arrIncome;
}
```

פרק שני

שאלה 4

Java

```
//--- א. הפעולה מחזירה אמת אם הספרה קיימת כספרת אחדות ---
//--- באחד מאיברי המחסנית ושקר אחרת ---
public static boolean isExist (Stack<Integer> stk, int num)
{
    boolean found = false;
    Stack<Integer> sTemp = new Stack<Integer>();
    while (! stk.isEmpty() && !found)
    {
        if (stk.top() % 10 == num)
            found = true;
        sTemp.push(stk.pop());
    }
    while (! sTemp.isEmpty())
        stk.push (sTemp.pop());
    return found;
}

//--- ב. הפעולה מחזירה אמת האם כל הספרות המשמעותיות ---
//--- נמצאות באיברים שבמחסנית ושקר אחרת ---
public static boolean allExist (Stack<Integer>stk)
{
    Stack<Integer> stkC = clone (stk);
    int num;
    while (!stkC.isEmpty())
    {
        num = stkC.pop();
        if (!isExist(stk, msd(num)))
            return false;
    }
    return true;
}

//--- פעולה המחזירה את הספרה המשמעותית ביותר במספר ---
public static int msd (int num)
{
    while (num > 9)
        num = num / 10;
    return num;
}
```


C#

הפתרון לשאלה זו נכתב ע"י ראמי ג'בלי

```
// פעולה מחזירה אמת אם יש במחסנית מספר שספרת
// האחדות שלו שווה למספר num אחרת מחזירה שקר
public static bool IsEsxist(Stack<int> stk, int num)
{
    Stack<int> s2 = new Stack<int>();
    int dig,x,c=0;
    while (!stk.IsEmpty())
    {
        x=stk.Pop();
        dig = x % 10;

        if (num == dig)
            c++;
        s2.Push(x);
    }
    while (!s2.IsEmpty())
        stk.Push(s2.Pop());

    return c > 0;
}

// פעולה מחזירה אמת אם כל הספרות המשמעותיות במחסנית
// מופיעות בספרת האחדות במספרים כל שהם אחרת מחזירה שקר
public static bool AllExist(Stack<int> s)
{
    Stack<int> s2=new Stack<int>();
    Stack<int> s3 = Clone(s);
    int c=0,x;

    while (!s.IsEmpty())
    {
        x = s.Pop();
        s2.Push(x);
        int last = LastDigit(x);
        Console.WriteLine(s3);
        if (!IsEsxist(s3, last))
            c++;
    }
    while (!s2.IsEmpty())
        s.Push(s2.Pop());

    return c == 0;
}

// פעולת עזר מחזירה הספרה האחרונה משמאל
public static int LastDigit(int x)
{
    int lastdig = 0;
    while (x > 0)
    {
        lastdig = x % 10;
        x = x / 10;
    }
    return lastdig;
}
```

שאלה 5

Java

המחלקה Race תכיל רשימה של מתחרים, ממוינת לפי תוצאת הריצה כך שהאצן המהיר ביותר (זמן מינימלי) יהיה בתחילת הר שימה והאיטי ביותר בסופה. פעולה ההוספה לרשימה - הוסף בצורה ממוינת לרשימה

```
//--- שאלה 4 - המחלקה תחרות ---  
public class Race  
{  
    //--- תכונות המחלקה ---  
    private Node<Competitor> lst;  
  
    public Race()  
    {  
        this.lst = null;  
    }  
  
    //--- ב. פעולה המוסיפה את המתחרה לרשימת המתחרים ---  
    //--- בצורה ממוינת לפי זמן הריצה (הראשון - המהיר ביותר) ---  
    public void add (Competitor c)  
    {  
        Node<Competitor> pos = this.lst, prev = null;  
        while (pos != null && pos.getValue().isBefore(c))  
        {  
            prev = pos;  
            pos = pos.getNext();  
        }  
        if (prev == null)  
            this.lst = new Node<Competitor> (c, this.lst);  
        else  
            prev.setNext(new Node<Competitor> (c, pos));  
    }  
  
    //--- ב. פעולה המקבלת דירוג/מקום בתחרות ---  
    //--- ומחזירה את שם המתחרה שהגיע לדירוג זה ---  
    public String rank (int x)  
    {  
        Node<Competitor> pos = this.lst;  
        while (pos != null && x > 0)  
        {  
            pos = pos.getNext();  
            x --;  
        }  
        if (pos != null)  
            return pos.getValue().getName();  
        return "NOR FOUND";  
    }  
}
```

```
//--- נכתב במחלקה Competition ---  
//--- פעולה המחזירה אמת אם הזמן של ---  
//--- המתחרה הנוכחי קצר יותר מהזמן של המתחרה c ---  
public boolean isBefore (Competitor c)  
{  
    int myTime = this.minutes * 60 + this.second;  
    int otherTime = c.minutes * 60 + c.second ;  
    return myTime < otherTime;  
}
```

ההוראות המסומנות מיותרות לפי ההנחה שניתנה בשאלה

C#

הפתרון לשאלה זו נכתב ע"י ראמי ג'בלי

```
public class Race
{
    private Node<Competitor> list;

    public Race()
    {
        this.list = null;
    }

    //פעולה מוסיפה עצם מסוג Competitor לאוסף המחוייין-יעילות לינארית
    public void Add(Competitor c)
    {
        Node<Competitor> pos = list;
        Node<Competitor> prev = null;
        if (pos == null)
            list = new Node<Competitor>(c, list);
        else if (c.GetTimeSecond() < pos.GetValue().GetTimeSecond())
            list = new Node<Competitor>(c, list);
        else
        {
            while (pos != null && c.GetTimeSecond() > pos.GetValue().GetTimeSecond())
            {
                prev = pos;
                pos = pos.GetNext();
            }
            prev.SetNext(new Node<Competitor>(c, pos));
        }
    }

    public string Rank(int x)
    {
        Node<Competitor> pos = list;
        for (int i = 1; pos != null && i < x; i++)
            pos = pos.GetNext();
        return pos.GetValue().GetName();
    }
}
```

שאלה 6

Java

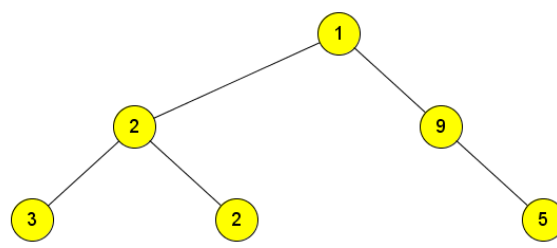
```

//--- פעולה עוטפת, שולחת את העץ עם ערך ראשוני ---
public static void numbers (BinNode<Integer> bt)
{
    numbers (bt, 0);
}

//--- פעולה המדפיסה את כל המספרים הנוצאים בכל מסלולי העץ ---
public static void numbers (BinNode<Integer> bt, int num)
{
    if (bt != null)
    {
        if (isLeaf(bt))
            System.out.println(bt.getValue()+ num*10);
        else
        {
            num = num * 10 + bt.getValue();
            if (bt.hasLeft())
                numbers(bt.getLeft(), num);
            if (bt.hasRight())
                numbers(bt.getRight(), num);
        }
    }
}

//--- פעולה המחזירה אמת אם עלה ושקר אחרת ---
//--- הנחה: bt != null ---
public static boolean isLeaf (BinNode<Integer> bt)
{
    return bt.getLeft() == bt.getRight();
}
    
```

123
122
195



```

--- הפתרון נכתב ע"י גרינוואלד - שימוש במחזורות ---
public static void printAll(BinNode<Integer> bt)
{
    if (bt != null)
        printAll(bt, "");
}
public static void printAll(BinNode<Integer> bt, String str)
{
    if (bt == null)
        System.out.println(str);
    else
    {
        // bt != null
        if (!bt.hasLeft() && !bt.hasRight())
            System.out.println(str + bt.getValue());
        else
        {
            if (bt.hasLeft())
                printAll(bt.getLeft(), str + bt.getValue());
            if (bt.hasRight())
                printAll(bt.getRight(), str + bt.getValue());
        }
    }
}

```

פתרונות המשנים את ערכי הצמתים (ולאחר מכן מחזירים אותם למצבם התחילי)

--- נכתב ע"י מאיר ליכט ---

```

public static void printAll(BinNode<Integer> t)
{
    // question 6 bagrut 2020. no envelop required, no harm to the tree
    if (isLeaf(t)) System.out.println(t.getValue());
    if (t.hasRight())
    {
        int rightVal = t.getRight().getValue();
        // set right value to this value * 10 + rightVal;
        t.getRight().setValue(rightVal + t.getValue()*10);
        printAll(t.getRight());
        //restore right to its original value
        t.getRight().setValue(rightVal);
    }
    if (t.hasLeft())
    {
        int leftVal = t.getLeft().getValue();
        // set left value to this value * 10 + leftVal;
        t.getLeft().setValue(leftVal + t.getValue()*10);
        printAll(t.getLeft());
        //restore left to its original value
        t.getLeft().setValue(leftVal);
    }
}

```

--- נכתב ע"י אריק ברזילי ---

```

public static void printAll(BinNode<Integer> t) //6 שאלה
{
    if (t==null)
        return;
    int num = t.getValue();
    if (t.getLeft() == t.getRight()){ //t is a leaf
        System.out.println(num);
    } else {
        if (t.hasLeft()) {
            t.getLeft().setValue(num*10 + t.getLeft().getValue());
            printAll(t.getLeft());
        }
        if (t.hasRight()) {
            t.getRight().setValue(num*10 + t.getRight().getValue());
            printAll(t.getRight());
        }
    }
    // צומת מתקן את ערכו, לאחר שסיים לטפל בילדים, או להדפיס את עצמו
    t.setValue( t.getValue() % 10 ); //back to original value
}

```

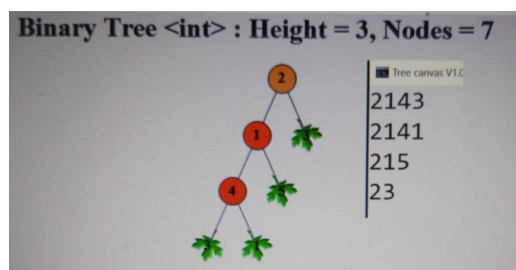
C#

הפתרון לשאלה זו נכתב ע"י ראמי ג'בלי

```
// פעולה בודקת אם העץ הוא עלה
public static bool IsLeaf(BinNode<int> t)
{
    return t.GetLeft() == null && t.GetRight() == null;
}

// פעולה מקבלת עץ ומספר num ומדפיסה המסלולים בעץ מייצג
public static void print(BinNode<int> t, int num)
{
    if (t != null)
    {
        num = num * 10 + t.GetValue();
        print(t.GetLeft(), num);
        print(t.GetRight(), num);
        if (IsLeaf(t))
            Console.WriteLine(num);
    }
}

// פעולה מדפיסה כל המספרים שהמסלולים בעץ מייצגים
public static void PrintAll(BinNode<int> t)
{
    print(t, 0);
}
```



C#

הפתרון לשאלה זו נכתב ע"י דיתה אוהב ציון

דרך א. הדפסת הערך כמספר שלם

```
// פעולת עזר האם עלה?
// מקבלת צומת ומחזירה אמת אם היא עלה, אחרת מחזירה שקר
public static bool IsLeaf(BinNode<int> t) //
{
    return !t.HasLeft() && !t.HasRight();
}

public static void Print(BinNode<int> tree, int num)
{
    if(tree!=null)
    {
        num = num * 10 + tree.GetValue(); // חישוב הערך עד מקום זה
        if (IsLeaf(tree)) Console.WriteLine(num); // אם הצומת עלה - הדפס
        else
        {
            Print(tree.GetLeft(), num);
            Print(tree.GetRight(), num);
        }
    }
}
```

דרך ב. הדפסת הערך כמחרוזת

```
public static void PrintyAll(BinNode<int> tree) // פעולת מעטפת
{
    string num = ""; // איתחול כמחרוזת ריקה
    if (tree != null)
        Print(tree, num);
}

public static void Print(BinNode<int> tree, string num)
{
    if(tree!=null)
    {
        num += tree.GetValue().ToString(); // חישוב הערך עד מקום זה
        if (IsLeaf(tree)) Console.WriteLine(num); // אם הצומת עלה - הדפס
        else
        {
            Print(tree.GetLeft(), num);
            Print(tree.GetRight(), num);
        }
    }
}
```

פרק פיוסי

מערכות מחשב ואסמבלי

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

שאלה 7

.א

לפניך קטע תוכנית באסמבלי

```
MOV BX,50h
MOV CL,8
L1: MOV AX,[BX]
    ROL AX,CL
    MOV [BX],AX
    ADD BX,2
    CMP BX,57h
    JBE L1
```

(1)

כתובת	50h	51h	52h	53h	54h	55h	56h	57h
תוכן	11h	22h	33h	44h	55h	66h	77h	88h
	22h	11h	44h	33h	66h	55h	88h	77h

AX		BX		CX	
AH	AL	BH	BL	CH	CL
		00h	50h		08h
22h	11h				
11h	22h				
		00h	52h		
44h	33h				
33h	44h				
		00h	54h		
66h	55h				
55h	66h				
		00h	56h		
88h	77h				
77h	88h				
		00h	58h		

(2) הקטע עטבר על 8 מילים בזיכרון החל מכתובת 50h ומחליף את ערך הבית הנמוך בזיכרון עם ערך הבית הגבוה בזיכרון.

ב.

לפניך הקטע

```
While (a>0 || b<=c)
{
  a = b+c;
  c--;
}
```

הנח שמשתנים a,b,c מכוונים, ומאוחסנים באוגרים AX,BX,CX בהתאם.

הנח שמשתנים a,b,c מכוונים, ומאוחסנים באוגרים AX,BX,CX בהתאם.

AGAIN:

```
CMP AX, 0
JG  CONT
CMP BX, CX
JG  STOP
```

CONT:

```
MOV AX, BX
ADD AX, CX
DEC CX
JMP AGAIN
```

STOP: NOP

ג. (בוטל בבחינה)

(1) בהנחה שסיבית D3 הסיבית הרביעית מצד ימין של אוגר AX מאוחסן הערך 0, שתי הפקודות הבעות מבצעות אותה פעולה. כלומר הערך של אוגר AH יהיה זהה לאחר הרצת שתי הפקודות

?	?	?	?	0	?	?	?
---	---	---	---	---	---	---	---

AND AH,11110111b פקודה 1

SUB AH,8 פקודה 2

תשובה - לא נכון!

בפקודה 1 - AH ישאר ללא שינוי

בפקודה 2 - AH ישתנה בהתאם לערך שיש בו

דוגמא: AH= 0C5h

AND AH,11110111b

אחרי הפקודה AH= 0C5h יישאר ללא שינוי

SUB AH,8

אחרי הפקודה AH= 0BDh ישאר ללא שינוי

(2) בהנחה שסיבית D3 הסיבית הרביעית מצד ימין של אוגר AX מאוחסן הערך 1, שתי הפקודות הבעות מבצעות אותה פעולה. כלומר הערך של אוגר AH יהיה זהה לאחר הרצת שתי הפקודות

?	?	?	?	1	?	?	?
---	---	---	---	---	---	---	---

תשובה - נכון!

בפקודה 1 - AH

בפקודה 2 - AH

בשני המצבים סיבית D3 תתאפס.

שאלה 8

שאלה 8

א. לפניך קטע תוכנית באסמבלי (בוטל בבחינה)

MOV AX,620h

SUB AH,76h

AX		ZF	CF	SF
AH	AL			
06h	20h			
90h			1	1

	06h+	- 06h
	<u>8Ah</u>	<u>76h</u>
	90h	

חיבור של הנגדי כדי לראות את
אוגר הדגלים

ב.

```

TEST PROC

    MOV BP, SP
    MOV DX, [BP+4] ; VALUE
    MOV CX, [BP+2] ; INDEX

    MOV AL, 1 ; FOUND
    MOV SI, 0
    CMP CX, 0
    JE NOTOK

L1:
    CMP ARR[SI], DL
    JE OK
    INC SI
    LOOP L1

NOTOK:
    MOV AL, 0

OK:
    RET 4

ENDP
    
```

חקר ביצועים

הפתרון לפרק זה נכתב ע"י: ???

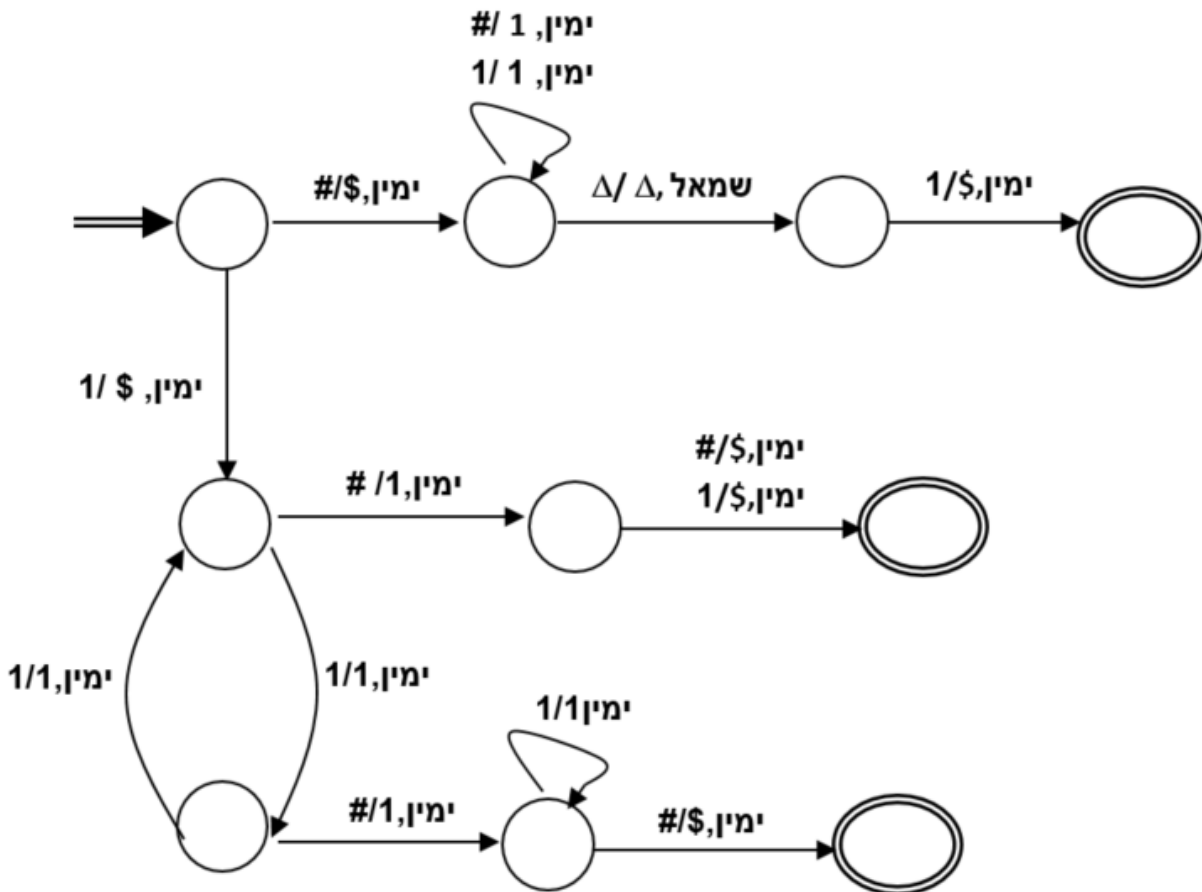
שאלה 9

שאלה 10

מודלים חישוביים

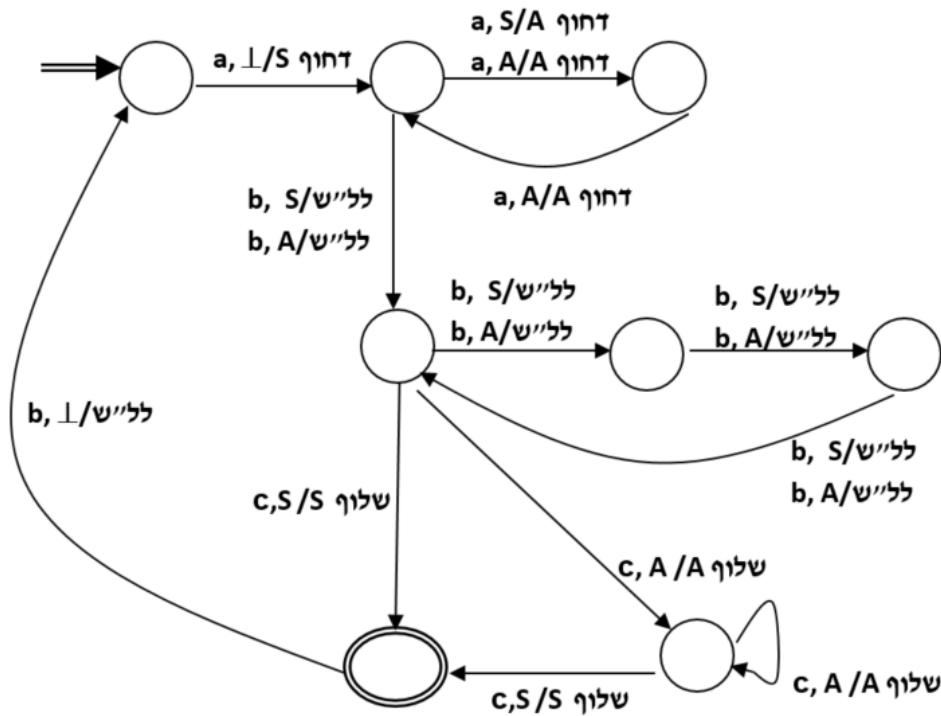
הפתרון לפרק זה נכתב ע"י: רחל לוזמר

שאלה 11



שאלה 12

.א



ב. סעיף שלא נדרש לפתור.

בשפת החיתוך המילים האפשריות מתקבלות כאשר:

$k=1$ או $k=3$ (מספר אי זוגי בתחום $[0,4]$), $m=1$ או $m=4$ (שארית חלוקה ב 3 היא 1 בתחום $[0,4]$).

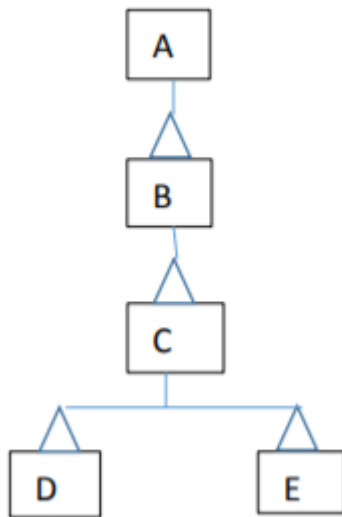
לכן, המילים המתקבלות הן: $L_3 = \{abc, ab^4c, a^3bc^3, a^3b^4c^3\}$

תכנות מונחה עצמים - Java

הפתרון לפרק זה נכתב ע"י: EVI גרינוואלד

שאלה 13

השאלה בודקת: ירושה ופולימורפיזם,
 א-היררכיה המחלקות:



ב- לפני שנענה על השאלה נבדוק מה מחזירות הפעולות foo ו bar ובאיזה מחלקות הן קיימות.

class	foo	bar	getX	foo-x	Type value
A	x	-	x	0	1
B	x+1	-	x	1	2
C	x+2	x	x	2	3
D	x-1	x	x	-1	4
E	x+2	x+1	x	2	5

פתרון 2	פתרון 1
<pre>public static int getType(Object m) { A a = (A)m; int k = a.foo() - a.getX() + 1; if (k == 3) { // type of C C c = (C)a; int y = c.bar() - c.getX(); if (y == 1) // type E k = 5; } else if (k == 0) // type D k = 4; return k; }</pre>	<pre>public static int getType(Object m) { A a = (A)m; int k = a.foo() - a.getX() + 1; if (k != 1 && k != 2) { // type is not A or B type is C C c = (C)a; if (k == 0) k = 4; else if (c.bar() - c.getX() == 1) k = k + 2; } return k; }</pre>

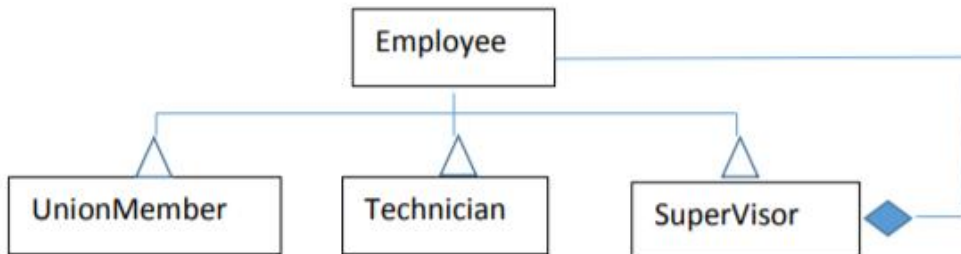
ג פלט הפעולה הראשית:

הסבר:
 שורה 1 מפעילה בנאי של A, מדפיסה A:x:9
 שורה 2 מפעילה בנאי של B שמפעיל בנאי של A, מדפיסה A:x:9
 שורה 3 מפעילה בנאי ריק של D שמפעיל בנאי ריק של C שמפעיל בנאי של B שמפעיל בנאי של A, פלט A:x:9 ואז מציגה את הפלט של הבנאי של D הריק אשר מוסיף 1 ל x ומציג: D:x:10
 שורה 4 מפעילה בנאי עם פרמטר אחד של D שמפעיל בנאי עם פרמטר אחד של C שמפעיל בנאי עם פרמטר אחד של B שמפעיל בנאי עם פרמטר אחד של A ומציג A:x:5 ולאחריו D:x:5
 שורה 5 מפעילה בנאי של D עם 2 פרמטרים, שמפעיל בנאי ריק של C שמפעיל בנאי ריק של B שמפעיל בנאי ריק של A ומדפיס A:x:9, לאחריו מעדכן את ערך תוכנת x להיות 19 ומציג D:x:19 $9+3+7 = 19$

```
A: x = 9
A: x = 9
A: x = 9
D: x = 10
A: x = 5
D: x = 5
A: x = 9
D: x = 19
```


שאלה 14

א- היררכית המחלקות



ב. כותרת המחלקות והתכונות

```

public class Employee
{
    // עובד רגיל
    public static final int POINTA = 4; // ניקוד לעובד בחברה (לא נדרש)
    private int seniority; // מספר שנים שבחברה

    public int getScore()
    {
        return POINTA + this.seniority;
    }
}

public class UnionMember extends Employee
{
    // חבר ועד
    public static final int POINTB = 2; // ניקוד עבור כל שנה בועד (לא נדרש)
    private int memberYears; // מספר שנים שחבר בועד

    public int getScore()
    {
        int score = super.getScore() * 2;
        score += 2 * this.memberYears;
        return score;
    }
}

public class Technician extends Employee
{
    // טכנאי
    private int numComputers; // מספר מחשבים לטיפול

    public int getScore()
    {
        return super.getScore() + this.numComputers;
    }
}

public class SuperVisor extends Employee
{
    // אחרי פרויקט
    public static final int NUM_EMPLOYEES = 15; // מספר עובדים תחתיו (לא נדרש)
    private Employee[] employees; // העובדים שבאחריותו

    public int getScore()
    {
        int score = super.getScore();
        for (int emp = 0; emp < NUM_EMPLOYEES; emp++)
        {
            if (this.employees[emp] != null)
                score += this.employees[emp].getScore();
        }
        return score;
    }
}
    
```

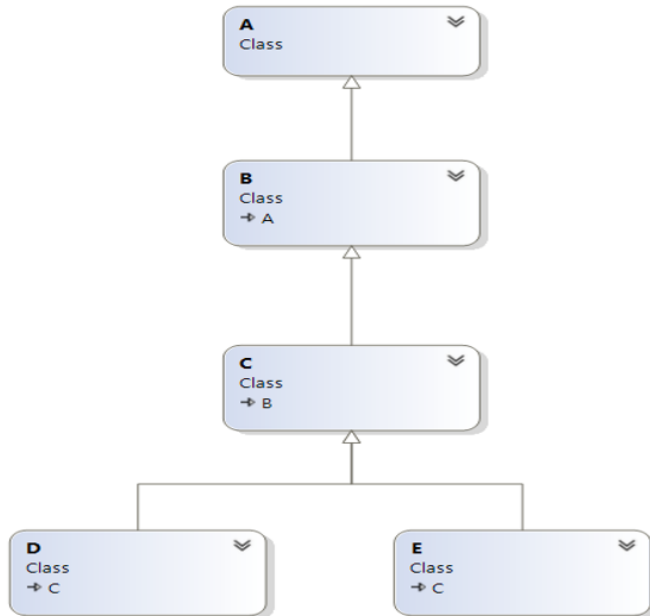
ג- פעולה המקבלת מערך עצמים ומחזירה 'אמת' אם ההצעה התקבלה, 'שקר' - אחרת

```
public static boolean isAccepted(Object[] arr)
{
    int scoreNo = 0;
    int scoreYes = 0;
    for (int k=0; k < arr.length; k++)
    {
        if (arr[k] != null && arr[k] instanceof Employee)
        {
            Employee emp = (Employee)arr[k];
            if (arr[k] instanceof Supervisor ||
                arr[k] instanceof UnionMember )
                scoreNo += emp.getScore();
            else // Technician or Employee not Supervisor and not UnionMember
                scoreYes += emp.getScore();
        }
    }
    // end for
    return scoreYes > scoreNo;
}
```

תכנות מונחה עצמים - C#
הפתרון לפרק זה נכתב ע"י : דיתה אוהב ציון

שאלה 15

א. תרשים המחלקות



ב. כדי לתכנן את הפעולה יש לראות את ערכי ברירת המחדל של העצמים

<pre>public static int GetType(object m) { A a = (A)m; if (a.GetX() == a.Foo()) return 1; // A if (a.Foo() - a.GetX() == 1) return 2; // B C c = (C)a; if (c.Foo() - c.GetX() == 2) return c.Bar() == c.GetX() ? 3 : 5; //C E return 4; //D }</pre>	<p>ערכי ברירת מחדל של עצמים</p> <p>A A.x=9 Foo=9 x=9</p> <p>B A.x=9 Foo=10 x=9</p> <p>C A.x=9 Foo=11 x=9 Bar=9</p> <p>D A.x=9 D.x=10 Foo=9 x=10 Bar=10</p> <p>E A.x=9 Foo=11 x=9 Bar=10</p>
---	---

הסבר

טיפוס	Object	ערכים
A		Foo=9 x=9
B		מופעל foo של B Foo=10 x=9
C	צריך המרה חדשה	Foo=11 x=9 ערכים x=9 Bar=9
E		x=9 Bar=10

ג. הפלט (שם העצם רק להבנה מה נבנה לכל אחד).

a1
A.x=9

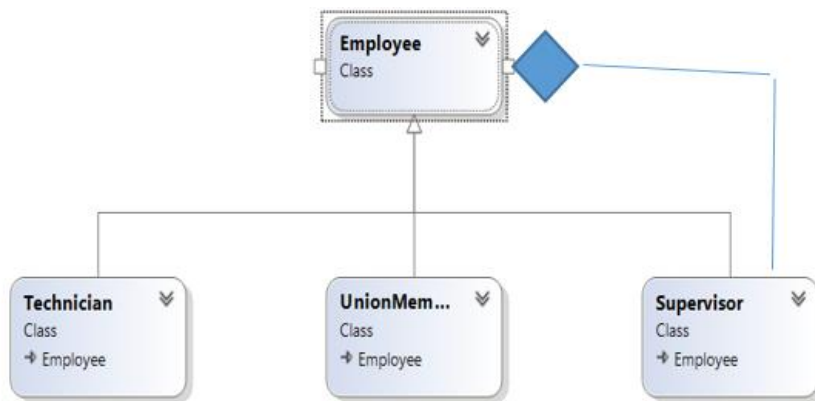
a2
A.x=9

a3
A.x=9
D.x=10

a4
A.x=5
D.x=5

a5
A.x=9
D.x=19

שאלה 16



א. תרשים מחלקות

ב. המחלקות

<pre> public class Employee { private int seniority; public virtual int GetScore() => 4 + seniority; } </pre>	עובד רגיל
<pre> public class UnionMember: Employee { private int unionSeniority; public override int GetScore() => base.GetScore()*2 + (unionSeniority*2); } </pre>	חבר ועד
<pre> public class Technician:Employee { private int compNum; public override int GetScore() => base.GetScore() + compNum; } </pre>	טכנאי
<pre> public class Supervisor:Employee { private List<Employee> employees; public override int GetScore() { </pre>	אחראי פרויקטים
<pre> int s = base.GetScore(); foreach (Employee e in employees) s += e.GetScore(); return s; } } </pre>	

ג. ההצבעה – נאמר בשאלה שהפעולה מקבלת מערך של כל העובדים.

```
public static bool IsAccepted(object[] arr)
{
    int sum = arr.Length;           // ניקוד מקסימלי - מספר העובדים בחברה
    foreach (object obj in arr)
    {
        Employee e = (Employee)obj;
        if (e is Supervisor || e is UnionMember)
            sum -= 1;               // הורד את המתנגדים
    }
    return sum >= 0;               // מחזיר אמת אם הסכום גדול מ 0, אחרת מחזיר שקר.
}
```

דרך נוספת:

```
public static bool IsAccepted(object[] arr)
{
    int sum = 0;                   // מונה הצבעות
    foreach (object obj in arr)
    {
        Employee e = (Employee)obj;
        if (e is Supervisor || e is UnionMember)
            sum -= 1;               // הורד את המתנגדים
        else
            sum += 1;               // הוסף את התומכים
    }
    return sum >= 0;               // מחזיר אמת אם הסכום גדול מ 0, אחרת מחזיר שקר.
}
```