



מדעי המחשב – 2 יחידות לימוד
פתרון בחינת הבגרות

פרק א

שאלה 1

```
//--- סעיף א' - פעולה בונה למחלקה Card ---
public Card(String color, int num)
{
    this.color = color;
    this.num = num;
}
//----- סעיף ב' - יצירת שני עצמים בעלי צבע וערך שונים
public static void main(String[] args)
{
    Card card1 = new Card ("Red", 3);
    Card card2 = new Card ("Black", 5);
}
}
```

שאלה 2

| b1 | b2 | b1.isInRange ? | b2.isInRange ? | פלט |
|---|---|----------------|----------------|-----|
|  |  | true - כן | | *** |
| | | | false - לא | |

| Student | | Student | |
|---------|------|---------|------------|
| num | mark | num | mark |
| 1111 | 92 | 3333 | 200 999 |

הערות למתכנתים בשפת java:

- בכל מקום שבו נדרש קטע תכנית, אין צורך להגדיר את אובייקט הקלט Scanner
- פתרון הבחינה מניח את קיומו של אובייקט הקלט הבא במחלקה:

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

שאלה 3

א. טבלת מעקב:

| a | b | x | a>b | פלט |
|---|---|----|-----|----------|
| 8 | 2 | 0 | כן | ***16*** |
| | | 16 | | 16 |
| | | | | |

ב. (כל שני מספרים כך ש-a אינו גדול מ-b. התנאי a>b לא יתקיים ויודפס רק ערך x בסיום)

א. a = 3, b = 5

```
System.out.print(" מספר ראשון --> ");
int a = input.nextInt();
System.out.print(" מספר שני --> ");
int b = input.nextInt();

int sum = a+b, sum1;

for (int i = 1 ; i < 218 ; i++)
{
    System.out.print(" מספר ראשון --> ");
    a = input.nextInt();
    System.out.print(" מספר שני --> ");
    b = input.nextInt();

    sum1 = a + b;

    if (sum1 < sum)
        System.out.println("smaller");
    sum1 = sum;
}
```

שאלה 4

שאלה 5

| s | x | i | i < 4 | s > 10 | פלט |
|----|---|---|-------|--------|-----|
| 0 | | | | | |
| 2 | 2 | 1 | כן | לא | --- |
| 10 | 8 | 2 | כן | לא | --- |
| 16 | 6 | 3 | כן | כן | *** |
| | | 4 | לא | | |

א.

ב. (המספר הראשון שייקלט יהיה גדול מ-10, האחרים כבר לא משנה)

א. 1, 2, 11

פרק ב

שאלה 6

```

A a = new A();
a.setNum(3);
a.setWord("am...");
System.out.println(a);
A a1 = new A(a.getNum(), "la");
System.out.println(a1);
    
```

.א

```

public class A
{
    //--- תכונות ---
    private int num;
    private String word;

    //--- פעולות בונות ---
    public A ()
    {
        this.num = 0;
        this.word = ""; // מחרוזת ריקה
    }
    public A (int num, String word)
    {
        this.num = num;
        this.word = word;
    }

    //--- נדרשת רק תכונה get למספר ---
    public int getNum () { return this.num; }

    //--- פעולות set לכל תכונה ---
    public void setNum(int num) { this.num = num; }
    public void setWord(String word) { this.word = word; }

    //--- toString ---
    public String toString()
    {
        String str = "";
        for (int i = 0 ; i < this.num ; i++)
            str += this.word;
        return str;
    }
}
    
```

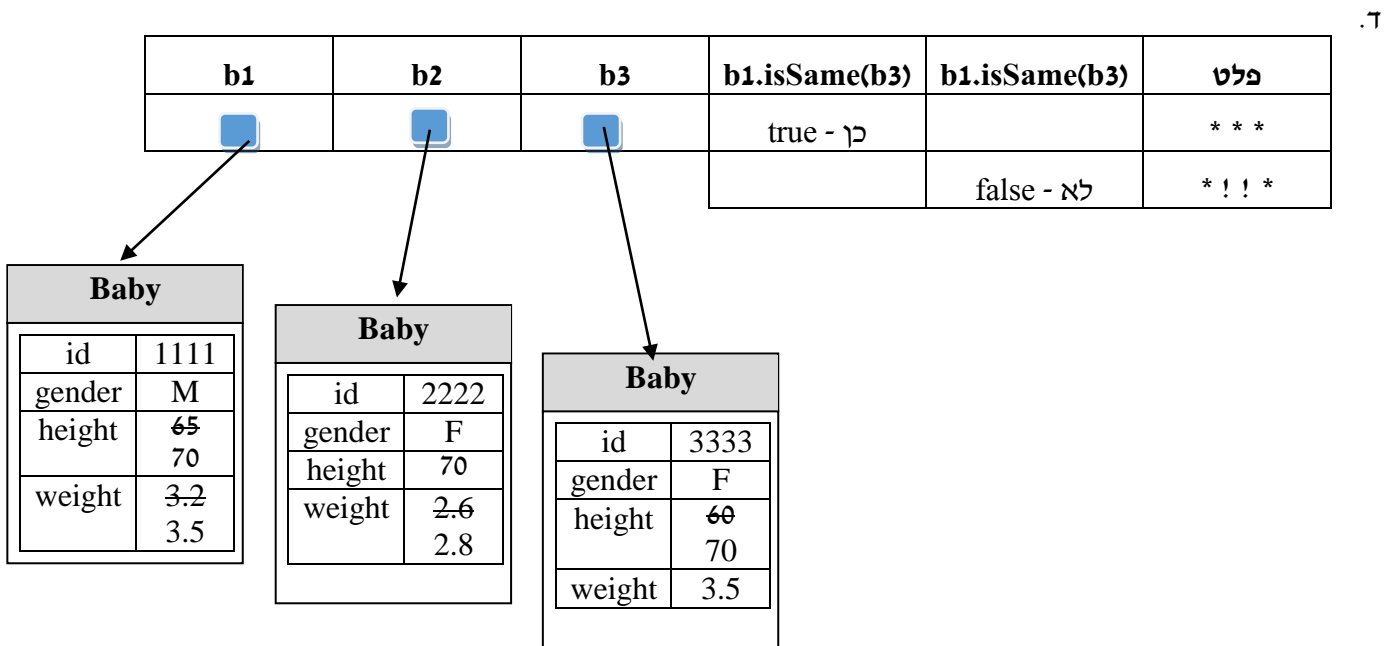
ב. הוראת הפלט השניה תדפיס: lalala

עלון 7

```
//--- סעיף א' - פעולות בונות ---
public Baby(String id, String gender, int height, double weight)
{
    this.id = id;
    this.gender = gender;
    this.height = height;
    this.weight = weight;
}
public Baby(String id, int height, double weight)
{
    this.id = id;
    this.gender = "F";
    this.height = height;
    this.weight = weight;
}

//--- סעיף ב' - הוספה למשקל ---
public void updateWeight(double weight)
{
    this.weight += weight;
}

//--- סעיף ג' - השוואת גובה ומשקל ---
public boolean isSame (Baby other)
{
    return this.height == other.height &&
           this.weight == other.weight;
}
```



```
//--- סעיף א' - בנאי במחלקה לקוח ---
public Customer(String name, String telNum)
{
    this.name = name;
    this.telNum = telNum;
}

public class Store
{
    //--- סעיף ב'1 - תכונות ---
    private Customer [] arrCust; // מערך הלקוחות
    private int current; // מספר הלקוחות במערך

    //--- סעיף ב'2 - הוספת לקוח למערך הלקוחות ---
    public void add (Customer cust)
    {
        this.arrCust[this.current] = cust;
        this.current ++;
    }
}
```

עלה 8

בשאלה הוגדר:

ב. נתונה המחלקה חנות – Store שיש לה שתי תכונות:

מערך חד־ממדי – arrCust מטיפוס Customer

מספר הלקוחות הנוכחי – current מטיפוס שלם, שערכו קטן מגודל המערך.

הנח שלכל תכונה הוגדרו ב־Java פעולות get ו־set, וב־C# פעולות Get ו־Set.

הנחה זו בעייתית.

לא יתכן שהמחלקה תחשוף את מבנה הנתונים שלה (לא מאפשר שדרוג למבנה נתונים אחר)

אבל ללא אחזור המערך בתכנית לא נוכל לבצע את המשימה הבאה:

(פעולה מסוג זה חייבת להיות פעולה פנימית במחלקה Store)

iii כתוב ב־Java או ב־C# במחלקה Program, פעולה חיצונית שתקבל s מטיפוס Store, ומספר שלם num בין 0 ל־99. הפעולה תחזיר את מספר הטלפון של הלקוח – Customer, ש־num הוא המציין של מקומו במערך arrCust. אם אין לקוח כזה – תחזיר הפעולה "no".

```
//--- סעיף ב'3 - מספר הטלפון של הלקוח במקום הנתון ---
public static String getPhone (Store s, int num)
{
    Customer [] arr = s.getArrCust();
    String phone = "no";
    if (arr[num] != null)
        arr[num].getTelNum();
    return phone;
}
```

פתרון שאינו מפר את עקרונות תמ"ע:

פעולה פנימית במחלקה Store:

```
//--- פעולה המקבלת מספר שלם בין 0 ו-99 ---  
//--- ומחזירה את מספר הטלפון של הלקוח במקום זה ---  
//--- אם אין לקוח במקום זה יוחזר סמ ---  
public String getPhone (int num)  
{  
    if (num < this.current)  
        return this.arrCust[num].getTelNum();  
    return "no";  
}
```

פעולה חיצונית בתכנית הראשית:

```
//--- סעיף ב'3 - מספר הטלפון של הלקוח במקום הנתון ---  
public static String getPhone (Store s, int num)  
{  
    return s.getPhone(num);  
}
```

פרק ג

שאלה 9

```
//--- סעיף א' - מוחזר אמת אם תת המערך הוא תת מערך מספרי זוגי ---  
public boolean isEvenSumMatrix (int k, int j)  
{  
    int row = this.matrix.length;        // מספר שורות  
    int col = this.matrix[0].length;     // מספר עמודות  
    for (int i = k ; i < row ; i++)  
    {  
        for (int n = j ; n < col ; n++)  
            if (this.matrix[i][n] % 2 != 0)  
                return false;  
    }  
    return true;  
}  
  
public static Scanner input = new Scanner(System.in);  
  
//--- סעיף ב' - פעולה ראשית ---  
public static void main(String[] args)  
{  
    System.out.print("מספר השורות --> ");  
    int row = input.nextInt();  
    System.out.print("מספר העמודות --> ");  
    int col = input.nextInt();  
  
    Stam s = new Stam (row, col);  
    if (! s.isDistrict())  
        System.out.println("לא כל האיברים שונים זה מזה");  
    else  
    {  
        int min = s.min();                // ערך המינימום  
        int rw = s.rowNum(min);           // שורת המינימום  
        int cl = s.colNum(min);           // עמודת המינימום  
        if (s.isEvenSumMatrix(rw, cl))  
            System.out.println(min + " - יש תת מערך מספרי זוגי החל מ-");  
        else  
            System.out.println(min + " - אין תת מערך מספרי זוגי החל מ-");  
    }  
}
```

אלף 10

```
//----- סעיף א' -----
//--- פעולה המחזירה את ההנחה בתשלום המיסים שהמשפחה זכאית לה ---
//--- אם אינה זכאית להנחה, הפעולה תחזיר 0 ---
public int reduction ()
{
    int redNum = 0;
    if (this.num > 6)
        redNum = 100;
    int redStud = this.numStudents * 40;
    return Math.max (redNum, redStud);
}
```

הערה לסעיף ב':

הפתרון המוצע אינו ממלא את דרישות השאלה במלואן. לפי נתוני השאלה, אין דרך לגשת למערך המשפחות שבמחלקה City מתוך המחלקה Program, וגם לא צריכה להיות גישה.

המחלקה City אמורה לספק את המידע על מספר המשפחות שלא קיבלו הנחה, ולכן, מקומה של הפעולה הוא במחלקה City, ומתוך Program רק מזמנים אותה.

במחלקה City שם נמצא מערך המשפחות, נכתוב את הפעולה הסורקת את המערך ומחזירה את מספר הנפשות שלא קיבלו הנחה תוך שימוש בפעולה המחשבת את גובה ההנחה למשפחה שמומשה במחלקה Family

```
//----- סעיף ב' -----
//--- פעול המחזירה את מספר המשפחות שלא קיבלו הנחה ---
public int noReduction ()
{
    int count = 0;
    for (int i = 0 ; i < this.numFamilies ; i++)
        if (this.arr[i].reduction() == 0)
            count ++ ;
    return count;
}
```

בפעולה הראשית (בתכנית) נזמן את הפעולה על אובייקט ct מסוג City:

```
System.out.println("no reduction : " + ct.noReduction());
```