

א. בגרות לבתי ספר על-יסודיים
ב. בגרות לנבחנים חיצוניים
מועד הבחינה: קיץ תשע"ו, 2016
מספר השאלון: 602,899222

מדעי המחשב

2 יחידות לימוד

הוראות לנבחן

- א. משך הבחינה: שלוש שעות.
- ב. מבנה השאלון ומפתח ההערכה:
בשאלון זה שלושה פרקים.
פרק ראשון – יש לענות על חמש השאלות 1-5,
לכל שאלה – 10 נקודות. (10×5) – 50 נקודות
פרק שני – יש לענות על שתיים מן השאלות 6-8,
לכל שאלה – 15 נקודות. (15×2) – 30 נקודות
פרק שלישי – יש לענות על אחת מן השאלות 9-10,
לשאלה – 20 נקודות. (20×1) – 20 נקודות
סה"כ – 100 נקודות
- ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ ממחשב הניתן לתכנות.
- ד. הוראות מיוחדות: (1) כתוב בשפה אחת בלבד את כל התכניות שאתה נדרש לכתוב.
(2) רשום על הכריכה החיצונית של המחברת את השפה שבה אתה

כותב – Java או C#.

הערה: בתכניות שאתה כותב לא יורדו לך נקודות אם תכתוב אות גדולה במקום אות קטנה או להפך.

כתוב במחברת הבחינה בלבד, בעמודים נפרדים, כל מה שבדצונך לכתוב כטיוטה (ראשי פרקים, חישובים וכדומה).
רשום "טיוטה" בראש כל עמוד טיוטה. רישום טיוטות כלשהן על דפים שמחוץ למחברת הבחינה עלול לגרום לפסילת הבחינה!

ההנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות ולנבחנים כאחד.

בהצלחה!

/המשך מעבר לדף/

השאלות

שים לב: עליך לכתוב בשפה אחת בלבד את כל התכניות שאתה נדרש לכתוב.

רשום על הכריכה החיצונית של המחברת את השפה שאתה כותב בה – Java או C#.

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

לפותרים ב-Java: בכל שאלה שנדרשת בה קליטה, הנח שבתכנית כתובה ההוראה:

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

פרק ראשון (50 נקודות)

ענה על חמש השאלות 1-5 (לכל שאלה – 10 נקודות).

1. לפניך מחלקה **Student** המייצגת תלמיד. למחלקה שתי תכונות: שם – name,

ושנת לידה – year.

המחלקה כתובה ב-Java וב-C#.

במחלקה חסר מימוש הפעולה הבונה.

Java

```
public class Student
{
    private String name;
    private int year;
    public Student(String name, int year) {...}
}
```

C#

```
public class Student
{
    private string name;
    private int year;
    public Student(string name, int year) {...}
}
```

א. השלם ב-Java או ב-C# את מימוש הפעולה הבונה של המחלקה **Student**.

ב. הנח שנתונה פעולה ראשית במחלקה Program. כתוב ב-Java או ב-C# בפעולה הראשית

הוראות אשר ייצרו שני עצמים בשם st1 ו-st2 מטיפוס **Student**,

שלשניהם שמות שונים זה מזה, ולשניהם אותה שנת לידה.

2. לפניך קטע מפעולה ראשית במחלקה Program, הכתוב ב-Java וב-C#. בפעולה הראשית הוגדר מערך חד-ממדי arr מטיפוס שלם. המערך מכיל מספרים גדולים מ-0.

```

Java
for (int i = 0; i < arr.length; i = i + 2)
{
    if ((arr[i] > 9) && (arr[i] < 100))
    {
        arr[i] = 0;
    }
}
    
```

```

C#
for (int i = 0; i < arr.Length; i = i + 2)
{
    if ((arr[i] > 9) && (arr[i] < 100))
    {
        arr[i] = 0;
    }
}
    
```

א. לפניך מערך arr.

	0	1	2	3	4	5	6	7
arr	12	2	324	33	67	888	9	5

עקוב בעזרת טבלת מעקב אחר ביצוע קטע הפעולה.

בטבלת המעקב יש לכלול:

עמודה ל-*i*, עמודה ל-*arr[i]*, ועמודה שבה יצוין אם התנאי בפקודת if מתקיים או אינו מתקיים.

הצג את המערך לאחר ביצוע הקטע.

ב. תן דוגמה מייצגת למערך בגודל 8 שבעברו, לאחר ביצוע קטע הפעולה, כל ערכי המערך יהיו שונים מ-0.

3. כתוב ב-Java או ב-C# פעולה חיצונית שתקבל משתנה בוליאני, ומספר שלם תלת ספרתי וגדול מ-0.

אם ערכו של המשתנה הבוליאני הוא true – הפעולה תחזיר את סכום הספרות של המספר. אחרת – הפעולה תחזיר את מכפלת הספרות של המספר.

/המשך בעמוד 4/

4. נתונה מחלקה **Worker** המייצגת עובד. למחלקה שתי תכונות: שם — name, וגובה המשכורת — salary.

במחלקה הוגדרה פעולה בונה שהכותרת שלה:

ב־Java: `public Worker(String name, int salary)`

ב־C#: `public Worker(string name, int salary)`

כמו כן הוגדרה במחלקה פעולה שהכותרת שלה:

ב־Java: `public boolean isBig()`

ב־C#: `public bool IsBig()`

הפעולה מחזירה true — אם המשכורת גדולה מ־9000 שקלים,

אחרת — הפעולה מחזירה false.

לכל תכונה הוגדרו ב־Java פעולות get ו־set וב־C# פעולות Get ו־Set.

לפניך קטע מפעולה ראשית במחלקה Program, הכתוב ב־Java וב־C#.

Java

```
Worker w1 = new Worker("Moshe", 10800);
```

```
Worker w2 = new Worker("Haim", 4800);
```

```
if (w1.isBig() && w2.isBig())
```

```
    System.out.println("Happy");
```

```
else
```

```
{
```

```
    System.out.println("Unhappy");
```

```
    if (!w1.isBig())
```

```
        System.out.println(w1.getName());
```

```
    if (!w2.isBig())
```

```
        System.out.println(w2.getName());
```

```
}
```

/המשך בעמוד 5/

C#

```

Worker w1 = new Worker("Moshe", 10800);
Worker w2 = new Worker("Haim", 4800);
if (w1.IsBig() && w2.IsBig())
    Console.WriteLine("Happy");
else
{
    Console.WriteLine("Unhappy");
    if (!w1.IsBig())
        Console.WriteLine(w1.GetName());
    if (!w2.IsBig())
        Console.WriteLine(w2.GetName());
}

```

עקוב אחר קטע הפעולה ורשום מה יהיה הפלט. במעקב הצג את העצמים ואת ערכי תכונותיהם.

ליד כל עצם כתוב מה תחזיר בעבורו הפעולה isBig ב־ Java או IsBig ב־ C#.

5.

בעיר מסוימת נקבע שבכל בניין שיש בו יותר מ־ 4 דירות חייב להיות ועד בית.

הוגדרה המחלקה בניין — **Building** שאחת התכונות שלה היא מספר הדירות בבניין:

```
private int numAps;
```

א. כתוב ב־ Java או ב־ C# פעולה במחלקה **Building** שתחזיר true — אם חייב להיות

ועד בית בבניין, אחרת — הפעולה תחזיר false.

ב. במחלקה Test בפעולה הראשית הוגדר מערך חד־ממדי bdng מטיפוס **Building**,

שכל אחד מאיבריו מייצג בניין בשכונה מסוימת בעיר.

כתוב ב־ Java או ב־ C#, בפעולה הראשית, קטע תכנית שימנה ויציג כפלט את

מספר הבניינים בשכונה שחייב להיות בהם ועד בית:

עליך להשתמש בפעולה שכתבת בסעיף א.

פרק שני (30 נקודות)

ענה על שתיים מהשאלות 6-8 (לכל שאלה – 15 נקודות).

6. הוחלט לארגן סקר ב-248 יישובים, כדי לבדוק את רמת שביעות הרצון של התושבים מן ההתנהלות של בתי הספר בתחומם. יישוב משתתף בסקר אם הוא מתאים לסקר לפי כמה תנאים. התנאים נקבעים בכל שנה מחדש.

נתונה המחלקה יישוב – City הכתובה ב-Java וב-C#.

Java

```
public class City
{
    private String name; // שם יישוב
    private int popul; // מספר תושבים ביישוב
    private int numOfSchools; // מספר בתי הספר ביישוב
    public City(String name, int popul, int numOfSchools) {...} // פעולה בונה
    public boolean isFit(){...} // פעולה שמחזירה true אם היישוב מתאים לסקר, ו-false אחרת
```

C#

```
public class City
{
    private string name; // שם יישוב
    private int popul; // מספר תושבים ביישוב
    private int numOfSchools; // מספר בתי הספר ביישוב
    public City(string name, int popul, int numOfSchools) {...} // פעולה בונה
    public bool IsFit(){...} // פעולה שמחזירה true אם היישוב מתאים לסקר, ו-false אחרת
```

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

א. כתוב ב-Java או ב-C# במחלקה Program פעולה ראשית שתקלוט, בעבור כל אחד מ-248 היישובים, את הפרטים האלה:

- שם היישוב
- מספר התושבים ביישוב
- מספר בתי הספר ביישוב

הפעולה הראשית תיצור לכל יישוב עצם מטיפוס City.
הפעולה הראשית תציג כפלט:

- לכל יישוב: את שם היישוב, ואם הוא מתאים לסקר או אינו מתאים לסקר.
- את מספר היישובים שאינם מתאימים לסקר.

ב. בשנת 2015 נקבעו התנאים האלה:

בסקר ישתתפו רק יישובים שיש בהם יותר מ-3 בתי ספר, ויותר מ-5,000 תושבים.
ממש ב-Java את הפעולה isFit או ב-C# את הפעולה IsFit, בהתאם לתנאים שנקבעו
בשנת 2015.

/המשך בעמוד 8/

7. לפניך חלק מממשק המחלקה עט — Pen. למחלקה ארבע תכונות:
צבע העט — color מטיפוס מחרוזת; שם היצרן — made מטיפוס מחרוזת;
מחיר העט — price מטיפוס ממשי; משקל העט — weight מטיפוס ממשי.

תיאור הפעולה	כותרת הפעולה ב־ Java
פעולה הבונה עט בצבע "red", של יצרן made, במחיר 10.0 שקלים, ובמשקל 25.0 גרם.	public Pen(String made)
פעולה הבונה עט בצבע color, של יצרן made, במחיר price, ובמשקל weight.	public Pen(String color, String made, double price, double weight)
פעולה המחזירה את שם היצרן של העט.	public String getMade()
פעולה המחזירה את מחיר העט.	public double getPrice()
פעולה המעדכנת את מחיר העט להיות x.	public void setPrice(double x)
פעולה המחזירה true אם שם היצרן של העט other זהה לשם היצרן של העט הנוכחי, אחרת — מחזירה false.	public boolean isSameMade(Pen other)
פעולה המחזירה true אם מחיר העט other זהה למחיר העט הנוכחי, אחרת — מחזירה false.	public boolean isSamePrice(Pen other)

תיאור הפעולה	כותרת הפעולה ב־ C#
פעולה הבונה עט בצבע "red", של יצרן made, במחיר 10.0 שקלים, ובמשקל 25.0 גרם.	public Pen(string made)
פעולה הבונה עט בצבע color, של יצרן made, במחיר price, ובמשקל weight.	public Pen(string color, string made, double price, double weight)
פעולה המחזירה את שם היצרן של העט.	public string GetMade()
פעולה המחזירה את מחיר העט.	public double GetPrice()
הפעולה המעדכנת את מחיר העט להיות x.	public void SetPrice(double x)
פעולה המחזירה true אם שם היצרן של העט other זהה לשם היצרן של העט הנוכחי, אחרת — מחזירה false.	public bool IsSameMade(Pen other)
פעולה המחזירה true אם מחיר העט other זהה למחיר העט הנוכחי, אחרת — מחזירה false.	public bool IsSamePrice(Pen other)

א. ממש במחלקה **Pen** :

ב־ Java את הפעולה הבונה `Pen(String made)`

או ב־ C# את הפעולה הבונה `Pen(string made)`

ב. ממש במחלקה **Pen**, ב־ Java את הפעולה `setPrice` או ב־ C# את הפעולה `SetPrice`.

ג. ממש במחלקה **Pen**, ב־ Java את הפעולה `isSamePrice` או ב־ C# את הפעולה

`IsSamePrice`.

ד. לפניך קטע מפעולה ראשית במחלקה `TestProg`, הכתוב ב־ Java וב־ C#.

עקוב אחר ביצוע קטע הפעולה ורשום את הפלט שיתקבל.

במעקב הצג את העצמים ואת ערכי תכונותיהם.

ליד כל עצם רשום את שמות הפעולות שהופעלו בו, ובעבור כל פעולה – את הערך שהוחזר.

Java

```
Pen a1 = new Pen("red" , "aaa" , 10.0 , 5.5);
Pen a2 = new Pen("bbb");
Pen a3 = new Pen("black" , "ccc" , 8.0 , 12.5);
a1.setPrice(15.0);
if(!(a1.isSamePrice(a3)))
    if(a2.isSameMade(a1))
        System.out.println("YES");
    else
        System.out.println("NO");
else
    System.out.println("OK");
```

C#

```
Pen a1 = new Pen("red" , "aaa" , 10.0 , 5.5);
Pen a2 = new Pen("bbb");
Pen a3 = new Pen("black" , "ccc" , 8.0 , 12.5);
a1.SetPrice(15.0);
if(!(a1.IsSamePrice(a3)))
    if(a2.IsSameMade(a1))
        Console.WriteLine("YES");
    else
        Console.WriteLine("NO");
else
    Console.WriteLine("OK");
```

/המשך בעמוד 10/

8.

חברת הטיולים "טייל בהנאה" מארגנת טיול משפחות. משפחות המשתתפות בטיול נדרשות לשלם בעבור ההדרכה ובעבור ארוחה הניתנת במהלך הטיול.

מחיר ההדרכה הוא קבוע – 100 שקלים לכל משפחה.
מחיר הארוחה בטיול הוא:

30 שקלים למשתתף – למשפחה בת 3 משתתפים או פחות.

28 שקלים למשתתף – למשפחה בת 4-5 משתתפים.

26 שקלים למשתתף – למשפחה שבה יותר מ-5 משתתפים.

לפניך המחלקה משפחה – **Family** הכתובה ב-Java וב-C#.

Java

```
public class Family
{
    private int num; // מספר המשתתפים מהמשפחה
    private String familyName; // שם המשפחה
    public Family(int num , String familyName) {...}
}
```

C#

```
public class Family
{
    private int num; // מספר המשתתפים מהמשפחה
    private string familyName; // שם המשפחה
    public Family(int num , string familyName) {...}
}
```

לכל תכונה הוגדרו ב-Java פעולות get ו-set וב-C# פעולות Get ו-Set.

א. כתוב במחלקה **Family**, ב-Java פעולה calcPrice או ב-C# פעולה CalcPrice,

שתחזיר את המחיר שהמשפחה נדרשת לשלם בעבור הטיול.

ב. כתוב ב-Java או ב-C#, במחלקה Program, פעולה ראשית שתקלוט בעבור כל משפחה

המשתתפת בטיול את מספר המשתתפים בה ואת שם המשפחה. הקלטיסטיים כאשר ייקלט 0 בעבור מספר המשתתפים מהמשפחה.

הפעולה הראשית תיצור לכל משפחה עצם מטיפוס **Family**.

הפעולה הראשית תציג כפלט:

— לכל משפחה: את שמה ואת המחיר שתשלם בעבור הטיול.

— את המספר של סך כל המשתתפים בטיול.

— את המחיר הגבוה ביותר שתשלם משפחה כלשהי בעבור הטיול.

עליך להשתמש בפעולה שכתבת בסעיף א.

שים לב: המשך הבחינה בעמוד הבא.

פרק שלישי (20 נקודות)

ענה על אחת מהשאלות 9-10.

9. נתונה מחלקה Card המייצגת קלף. למחלקה שתי תכונות: ערך value, בין 1 ל-9, מטיפוס שלם; צבע color מטיפוס שלם: 1 מייצג אדום, 2 מייצג צהוב, 3 מייצג ירוק, 4 מייצג כחול.

כותרת הפעולה ב־ Java	תיאור הפעולה
public Card(int value, int color)	פעולה הבונה קלף שערכו value וצבעו color.
public boolean isSame(Card other)	פעולה המחזירה true אם הערך של הקלף other זהה לערך של הקלף הנוכחי וגם צבע הקלף other זהה לצבע הקלף הנוכחי. אחרת – הפעולה מחזירה false.
public int compareVal(Card other)	פעולה המחזירה: 1 – אם הערך של הקלף הנוכחי גדול מהערך של הקלף other, 2 – אם הערך של הקלף הנוכחי קטן מהערך של הקלף other, 0 – אם הערכים של הקלף הנוכחי ושל הקלף other זהים.

כותרת הפעולה ב־ C#	תיאור הפעולה
public Card(int value, int color)	פעולה הבונה קלף שערכו value וצבעו color.
public bool IsSame(Card other)	פעולה המחזירה true אם הערך של הקלף other זהה לערך של הקלף הנוכחי וגם צבע הקלף other זהה לצבע הקלף הנוכחי. אחרת – הפעולה מחזירה false.
public int CompareVal(Card other)	פעולה המחזירה: 1 – אם הערך של הקלף הנוכחי גדול מהערך של הקלף other, 2 – אם הערך של הקלף הנוכחי קטן מהערך של הקלף other, 0 – אם הערכים של הקלף הנוכחי ושל הקלף other זהים.

נתונה המחלקה חפיסת קלפים – **Deck** שיש לה תכונה אחת – מערך חד-מדמי **allCards** מטיפוס **Card**.

תיאור הפעולה	כותרת הפעולה ב-Java
פעולה הבונה חפיסת קלפים: הפעולה מגדירה מערך allCards בגודל 36 ובו 36 קלפים שונים זה מזה.	<code>public Deck()</code>
פעולה המחזירה את הקלף הנמצא במקום i במערך allCards .	<code>public Card seeCard(int i)</code>
תיאור הפעולה	כותרת הפעולה ב-C#
פעולה הבונה חפיסת קלפים: הפעולה מגדירה מערך allCards בגודל 36 ובו 36 קלפים שונים זה מזה.	<code>public Deck()</code>
פעולה המחזירה את הקלף הנמצא במקום i במערך allCards .	<code>public Card SeeCard(int i)</code>

א. ממש ב-Java את הפעולה `seeCard` או ב-C# את הפעולה `SeeCard` שבמחלקה **Deck**.

לפניך תיאור של משחק קלפים לשני שחקנים.

בתחילת המשחק לכל אחד מן השחקנים יש חפיסת קלפים.

בכל תור:

כל אחד מן השחקנים מגריל מספר אקראי **i** בין 0 ל-35 (כולל) ומציג את הקלף שנמצא במקום **i** במערך **allCards** מהחפיסה שלו.

– אם שני הקלפים זהים (בערך ובצבע), כל אחד מן השחקנים מקבל נקודה אחת.

– שחקן שערך הקלף שלו גדול מערך הקלף של השחקן האחר, מקבל 3 נקודות נוסף על כך.

– כל שחקן שצבע הקלף שלו ירוק, מקבל 2 נקודות.

המשחק מסתיים כאשר אחד השחקנים צבר לפחות 28 נקודות.

ב. כתוב ב-Java או ב-C# קטע מפעולה ראשית במחלקה **Program** שידמה את משחק הקלפים.

קטע התכנית ידפיס את מספר הנקודות שצבר כל אחד מן השחקנים בסיום המשחק.

עליך להשתמש בפעולות המחלקות **Card** ו-**Deck**, ואינך חייב לממש אותן.

הנה שבמחלקות **Card** ו-**Deck** הוגדרו ב-Java פעולות `get` ו-`set`

ב-C# פעולות `Get` ו-`Set` לכל תכונה.

10. מטריצה דלילה – sparse matrix – היא מערך דר־ממדי מטיפוס שלם שהערך של רוב האיברים בו הוא 0.

לדוגמה המערך הד־ממדי בגודל 4x5 שלפניך.

	0	1	2	3	4
0	0	1	0	0	0
1	0	0	0	0	9
2	0	0	0	0	0
3	7	0	0	0	0

כדי לחסוך מקום בזיכרון של המחשב, אפשר לשמור רק את האיברים של המערך הד־ממדי השונים מ־0. בעבור כל איבר נשמרים ערכו ומקומו במערך הד־ממדי: מספר השורה ומספר העמודה.

בעבור המערך הד־ממדי שבדוגמה, 3 האיברים השונים מ־0 יישמרו באופן הזה:

value = 1 row = 0 col = 1	value = 9 row = 1 col = 4	value = 7 row = 3 col = 0
---------------------------------	---------------------------------	---------------------------------

הוגדרה המחלקה **Item** המייצגת איבר שונה מ־0 במערך הד־ממדי. תכונותיה הן: הערך של האיבר במערך הד־ממדי, ומקומו: מספר השורה ומספר העמודה.

א. (1) כתוב ב־Java או ב־C# את הכותרת ואת התכונות של המחלקה **Item**.

(2) כתוב ב־Java או ב־C# פעולה בונה במחלקה **Item**, המקבלת כפרמטרים ערכים

לכל תכונה.

הוגדרה מחלקה נוספת – **Sparse**, שתכונותיה הן:

מערך חד-ממדי `itemAr` מטיפוס `Item`,

`rows` – מטיפוס שלם, מספר השורות במטריצה דלילה,

`cols` – מטיפוס שלם, מספר העמודות במטריצה דלילה.

ב. כתוב ב-Java או ב-C# את כותרת המחלקה **Sparse** ואת התכונות שלה.

במחלקה **Sparse** נתונה פעולה שהכותרת שלה:

ב-Java: `public int countNoZero(int[][] mat)`

ב-C#: `public int CountNoZero(int[,] mat)`

הפעולה מקבלת מערך דר-ממדי ומחזירה את מספר האיברים בו השונים מ-0.

ג. כתוב ב-Java או ב-C# פעולה בונה במחלקה **Sparse**, המקבלת כפרמטר מערך דר-ממדי

מטיפוס שלם שהוא מטריצה דלילה.

עליך להשתמש בפעולה הנתונה.

בהצלחה!