

סוג הבדיקה: א. בגרות לבתי ספר על-יסודיים

ב. בגרות לנבחנים חיצוניים

מועד הבדיקה: קיץ תשע"ד, 2014

מספר השאלה: 603,899205

מַדְעֵי הַמִּחְשָׁב ב'

2 ייחדות לימוד (השלמה ל-5 י"ל)

הוראות לנבחן

א. משך הבדיקה: שלוש שעות.ב. מבנה השאלה ופתחה הערכה: בשאלון זה שני פרקים.

פרק ראשון — בפרק זה ארבע שאלות,
ומהן יש לענות על שתיים.
 $(25 \times 2) = 50$ נקודות

פרק שני — בפרק זה שאלות באربعة מסלולים שונים.
ענה על שאלות בק במסלול שלמדת,
לפי ההוראות בקבוצת השאלות במסלול זה.
 $(25 \times 2) = 50$ נקודות

סה"כ — 100 נקודות

ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ מחשב הנitin לתכניות.ד. הוראות מיוחדות:

(1) את כל התכניות שאתה נדרש לכתוב בשפת מחשב בפרק הראשון
כתב בשפה אחת בלבד — Java או C#.

(2) רשום על הכריכה החיצונית של המחברת באיזו שפה אתה כותב — Java או C#.

(3) רשום על הכריכה החיצונית של המחברת את שם המסלול שלמדת.
המסלול הוא אחד מארבעת המסלולים האלה:

מערכות מחשב ואסטREL, מבוא לחקור ביצועים, מודלים חישוביים, תכניות מונחה עצמים.

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

כתב במחברת הבדיקה בלבד, בעמודים נפרדים, כל מה שברצונך לכתוב בтипewriter (ראשי פרקים, חישובים וכדומה).
רשום "טיוטה" בראש כל עמוד טיוטה. רישום טיותות כלשhn על דפים שמוחוץ למחברת הבדיקה עלול לגרום לפסילת הבדיקה!

הנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות לנבחנים כאחד.

בהתלה!

/המשר מעבר לדף/

השאלות

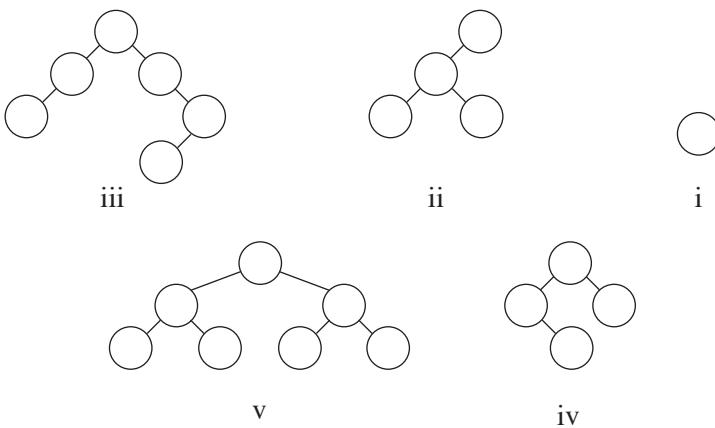
בשאלון זה שני פרקים: פרק ראשון ופרק שני.
עליך לענות על שאלות משני הפרקים, לפי ההוראות בכל פרק.

פרק ראשון (50 נקודות)

שים לב: בכל שאלה שנדרש בה שימוש אתה יכול להשתמש בעולות של המחלקות רשיימה, תור, מחסנית, עץ בינרי וחוליה, בלי למשם אותן. אם אתה משתמש בעולות נוספות, עליך למשם אותן.

ענה על שתים מהשאלות 1-4 (לכל שאלה — 25 נקודות).

1. **עץ בינארי מאוזן** הוא **עץ בינארי שבכל צומת שלו הערך המוחלט של הפרש הגבהים בין התת-עץ הימני לתת-עץ השמאלי הוא לכל היותר 1.**
- א. לפניך 5 עצים בינאריים – a. לכל אחד מן העצים, קבע אם הוא **עץ בינארי מאוזן** או אינו **עץ בינארי מאוזן**. אם העץ אינו עץ בינארי מאוזן, העתק אותו למחרתך וסמן x בצומת שmaps את האיזון.



- ב. כתוב ב- Java או ב- C# פעלת שתתקבל **עץ בינארי כלשהו**, ותחזיר true אם הוא **עץ בינארי מאוזן**, אחרת — הפעלה תחזיר false.
- אם הפרמטר שמוועבר לפעלת הוא null — הפעלה תחזיר true.

.2

שים לב: לשאלת זו שני נוסחים:
 נוסח אחד ב- Java (עמוד 3),
 ונוסח אחד ב- C# (עמוד 4).
 עבד על פי השפה שלמדת.

לפתרונות ב- Java

המחלקה **Collec** היא אוסף של מספרים שלמים גדולים מ- 0. לאוסף זה אפשר להוסיף איבר רק אם אין באוסף איבר אחר גדול ממנו.

- .א. כתוב ב- Java את כוורתה המחלקה **Collec** ואת התכונות שלה.
- .ב. כתוב ב- Java שתי פעולות בונות למחלקה **Collec** :
 - פעולה בונה, בלי פרמטרים, היוצרת אוסף ריק.
 - פעולה בונה שתתקבל מספר שלם n גדול מ- 0, ותיצור אוסף שיש בו איבר אחד שערכו n .
- .ג. ממש ב- Java. במחלקה **Collec** פעולה `add`, שתקבל מספר שלם וגודל מ- 0. ותוסיף אותו לאוסף, אם אפשר. אם המספר צורף לאוסף, הפעולה תחזיר `true`. אחרת — הפעולה תחזיר `false`.
- .ד. ממש ב- Java במחלקה **Collec** פעולה `small`, שתחזיר את המספר הקטן ביותר באוסף. אם האוסף ריק, הפעולה תחזיר -1 .
- .ה. ממש ב- Java במחלקה **Collec** פעולה בשם `smallest`, שתקבל עצם `c` מטיפוס **Collec**, ותחזיר את המספר הקטן ביותר מבין שני האוספים — האוסף הנוכחי ו- `c`. הנח כי שני האוספים אינם ריקים.

לפתרונות ב- C#

המחלקה **Collec** היא אוסף של מספרים שלמים וגדולים מ- 0. לאוסף זה אפשר להוסיף איבר רק אם אין בו איבר אחר גדול ממנו.

- א. כתוב ב- C# את כוורתה המחלקה **Collec** ואת התכונות שלה.
- ב. כתוב ב- C# שתי פועלות בונות למחלקה **Collec** :
 - פעללה בונה, בלי פרמטרים, היוצרת אוסף ריק.
 - פעללה בונה שתתקבל מספר שלם n גדול מ- 0, ותיצור אוסף שיש בו איבר אחד שערךו n .
- ג. ממש ב- C# במחלקה **Collec** פעללה **Add**, שתתקבל מספר שלם וגדול מ- 0 ותוסיף אותו לאוסף, אם אפשר. אם המספר צורף לאוסף, הפעולה תחזיר `true`. אחרת — הפעולה תחזיר `false`.
- ד. ממש ב- C# במחלקה **Collec** פעללה **Small**, שתחזיר את המספר הקטן ביותר באוסף.
- ה. ממש ב- C# במחלקה **Collec** פעללה בשם **Smallest**, שתתקבל עצם `c` מטיפוס **Collec**, ותחזיר את המספר הקטן ביותר מבין שני האוספים — האוסף הנוכחי ו- `c`. הנה כי שני האוספים אינם ריקים.

שים לב: לשאלה זו שני נוסחים:

נוסח אחד לפתרים על **פי התכנית הנוכחית** (ב- Java ו- C# – עמוד 5),

נוסח אחד לפתרים על **פי התכנית החדשה** (ב- Java ו- C# – עמוד 6).

עובד על **פי התכנית** שלמדת.

לפתרונות על פי התכנית הנוכחית

לפניר שלוש מחלקות:

המחלקה **Birth** – תאריך לידה, שתכונותיה הן: יום בחודש – day , חודש – month –

ושנה – year . כל אחת מן התכונות היא מספר שלם.

המחלקה **Student** – תלמיד, שתכונותיה הן: שם התלמיד – name מטיפוס מחוوظ,

ותאריך הלידה של התלמיד – birthDay מטיפוס Birth .

המחלקה **School** – בית ספר, שתכונתה היא: מערך חד-ממדי ar בגודל 6 , מטיפוס

List<Student> . כל תא במערך ar מייצג שכבת גיל בבית הספר: תא 0 מייצג את שכבה ז',

תא 1 מייצג את שכבה ח' וכך הלאה, עד תא 5 שמייצג את שכבה י"ב. כל תא מכיל רשימה של

כל תלמידי השכבה.

הנח שככל מחלוקת יש: פעולה בונה בררת מחדל ופעולה בונה שמקבלת פרמטר לכל תוכנה,

ב- Java פועלות get ו- set לכל תוכנה, וב- C# Get ו- Set לכל תוכנה.

כתוב ב- Java או ב- C# פעולה חיצונית, שתקבל עצם מטיפוס School ותחזיר מערך חד-ממדי,

בגודל 12 מטיפוס List<Student> . כל תא במערך מייצג חדש בשנה: תא 0 מייצג את ינואר,

תא 1 מייצג את פברואר וכך הלאה, עד תא 11 שמייצג את דצמבר. כל תא יכול רשימה של

כל התלמידים מכל השכבות שנולדו בחודש זה, בלי חשבות לסדר.

לפתרונות על פי התכנית החדשה

לפניך שלוש מחלקות:

המחלקה **Birth** — תאריך לידה, שתכונותיה הן: יום בחודש — day , חודש — month ו שנה — year . כל אחת מן התכונות היא מספר שלם.

המחלקה **Student** — תלמיד, שתכונותיה הן: שם התלמיד — name מטיפוס מחוوظת, ותאריך הלידה של התלמיד — birthDay מטיפוס Birth .

המחלקה **School** — בית ספר שתכונתה היא: מערך חד-ממדי `zr` בגודל 6 , מטיפוס `Node<Student>` תא 0 מייצג את שכבה ז', תא 1 מייצג את שכבה ח' וכן הלאה, עד תא 5 שמייצג את שכבה י"ב. כל תא מכיל רשימה של כל תלמידי השכבה.

הנח שכל מחלקה יש: פעולה בונה בררת מחדל ופעולה בונה שמקבלת פרמטר לכל תוכנה, ב- Java פועלות `get` ו `set` לכל תוכנה, וב- C# `Get` ו `Set` לכל תוכנה.

כתוב ב- Java או ב- C# פעולה חיצונית, שתקבל עצם מטיפוס **School** ותחזיר מערך חד-ממדי בגודל 12 מטיפוס `Node<Student>`. כל תא במערך מייצג חדש בשנה: תא 0 מייצג את ינואר, תא 1 מייצג את פברואר וכן הלאה, עד תא 11 שמייצג את דצמבר. כל תא יכול רשימה של כל התלמידים מכל השכבות שנולדו בחודש זה, בלי חשיבות לסדר.

4. **שים לב:** לשאלת זו שני נוסחים:

נוסח אחד לפותרים על פי התכנית הנוכחית (ב- C# ובי- Java – עמודים 11-7),
נוסח אחד לפותרים על פי התכנית החדשנית (ב- Java ובי- C# – עמודים 12-16).
עובד על פי התכנית של마다.

לפתרונות על פי התכנית הנוכחית

בשאלה זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניהם.

א. נתונה פעליה המקבלת רשימה lst של מספרים שלמים:

C#	Java
<pre>public static int What(List<int> lst) { if (lst.IsEmpty()) return 0; int x = lst.GetFirst().GetInfo(); lst.Remove(lst.GetFirst()); if(x >= 0) return What(lst); //(*) return 1 + What(lst); }</pre>	<pre>public static int what(List<Integer> lst) { if (lst.isEmpty()) return 0; int x = lst.getFirst().getInfo(); lst.remove(lst.getFirst()); if(x >= 0) return what(lst); //(*) return 1 + what(lst); }</pre>

נתונה הרשימה lst: -2 → -9087 → 16 → -43 → 5 → |||

(1) עקב אחר הפעולה הנתונה בעבור הרשימה lst .

רשום את הערך שיווצר, וسرטט את הרשימה lst לאחר ביצוע הפעולה.

(2) מה הפעולה מבצעת?

(3) החליפו את ההוראה המסומנת ב- (*) להוראה return 0; מה תבצע הפעולה לאחר החלפה?

i. מה תבצע הפעולה לאחר החלפה?

ii. מה תכיל הרשימה lst לאחר ביצוע הפעולה?

(שים לב: סעיף ב של השאלה בעמוד הבא.)

(אין קשר לסעיף א.). ב.

שים לב: לסעיף זה שני נוסחים:

נוסח אחד ב- Java (עמודים 9-8),

נוסח אחד ב- C# (עמודים 11-10). עבד על פי השפה שלמדת.

לפתרונות ב- Java

לפניך הפוליה sum הכתובת בחלוקת ראשית.

```
public static int sum (Stack<Integer> s)
{
    if (s.isEmpty())
        return 0;
    int x = s.pop();
    if (x%6 == 0)
        return x + sum(s);
    return sum(s);
}
```

נתונה המחסנית s .

→ ראש המחסנית	12
	4
	33
	6
	30
	0

(**שים לב:** המשך סעיף ב בעמוד הבא.)

/המשך בעמוד 9/

(1) עקוב אחר הפעולה sum בעבור המחסנית s, ורשום את הערך שיותזר.

במשך הראה את המעבר על המחסנית s .

(2) לפניו הפעולה sod .

public static void sod(Queue<Stack<Integer>> qq, Queue<Integer> qm)

{

 if (!qq.isEmpty())

 {

 int x = sum(qq.remove());

 qm.insert(x);

 sod(qq , qm);

 }

}

לפניך קטע תכנית המשמש בפעולת sod .

סרטט במחברתך כל אחד משני התווים q ו qr , לפניו הקראיה לפעולת sod ואחריו

ביצוע הפעולה sod .

public static void main(String[] args)

{

 Stack<Integer> s1 = new Stack<Integer>();

 Stack<Integer> s2 = new Stack<Integer>();

 Queue<Stack<Integer>> q1 = new Queue<Stack<Integer>>();

 s1.push(0);

 s1.push(30);

 s1.push(6);

 s1.push(33);

 s1.push(4);

 s1.push(12);

 s2.push(23);

 s2.push(36);

 s2.push(1);

 q1.insert(s1);

 q1.insert(s2);

 Queue<Integer> qr = new Queue<Integer>();

 sod(q1 , qr);

}

/המשך בעמוד /

C# - לפוטרים ב-

לפניר הפעולה Sum הכתובת במחלקה ראשית.

```
public static int Sum (Stack<int> s)
{
    if(s.IsEmpty())
        return 0;
    int x = s.Pop();
    if(x%6 == 0)
        return x + Sum(s);
    return Sum(s);
}
```

נתונה המחסנית s .

→ ראש המחסנית	12
	4
	33
	6
	30
	0

(1) עקוב אחר הפעולה Sum בעבור המחסנית s , ורשות את הערך שיווצר.

במשך הראה את המעבר על המחסנית s .

(2) לפניר הפעולה Sod .

```
public static void Sod(Queue<Stack<int>> qq, Queue<int> qm)
```

```
{
    if (!qq.IsEmpty())
    {
        int x = Sum(qq.Remove());
        qm.Insert(x);
        Sod(qq , qm);
    }
}
```

/המשך בעמוד הבא(2) בעמוד הבא).

(שים לב: המשך תת-סעיף (2) בעמוד הבא.)

לפניך קטע תכנות המשמש בעולה .
Sod

סרטט במחברתך את כל אחד משני התווים q_1 ו- qr , לפני הקראיה לפעולה
.Sod ואחרי ביצוע הפעולה

```
public static void Main(string[] args)
{
    Stack<int> s1 = new Stack<int>();
    Stack<int> s2 = new Stack<int>();

    Queue<Stack<int>> q1 = new Queue<Stack<int>>();
    s1.Push(0);
    s1.Push(30);
    s1.Push(6);
    s1.Push(33);
    s1.Push(4);
    s1.Push(12);
    s2.Push(23);
    s2.Push(36);
    s2.Push(1);
    q1.Insert(s1);
    q1.Insert(s2);
    Queue<int> qr = new Queue<int>();
    Sod(q1 , qr);
}
```

לפתרונות על פי התכנית החדשה

בשאלה זו שני סעיפים א-ב, שאין קשר ביניהם. ענה על שנייהם.

א. נתונה פעולה המתקבלת רישימה lst של מספרים שלמים:

C#	Java
<pre>public static int What(Node<int> lst) { if (lst == null) return 0; int x = lst.GetValue(); lst = lst.GetNext(); if (x >= 0) return What(lst); //(*) return 1 + What(lst); }</pre>	<pre>public static int what(Node<Integer> lst) { if (lst == null) return 0; int x = lst.getValue(); lst = lst.getNext(); if (x >= 0) return what(lst); //(*) return 1 + what(lst); }</pre>

נתונה הרשימה $|1| \rightarrow -2 \rightarrow -9087 \rightarrow 16 \rightarrow -43 \rightarrow 5 \rightarrow$ נתונה הרשימה $|1|$

(1) עקוב אחר הפעולה הנתונה בעבור הרשימה lst.

רשום את הערך שיווצר, וסרטט את הרשימה lst לאחר ביצוע הפעולה.

(2) מה הפעולה מבצעת?

(3) החליפו את ההוראה המסומנת ב- (*) להוראה `return 0;`

i. מה תבצע הפעולה לאחר החלפה?

ii. מה תכיל הרשימה lst לאחר ביצוע הפעולה?

(שים לב: סעיף ב בעמוד הבא.)

(אין קשר לסעיף א.). ב.

שים לב: לסעיף זה שני נוסחים:

נוסח אחד ב- Java (עמודים 13-14),

נוסח אחד ב- C# (עמודים 15-16). עבור על פי השפה שלמדת.

לפתרונות ב- Java

לפניך הפעולה sum הכתובת במחלקה ראיית.

```
public static int sum (Stack<Integer> s)
{
    if (s.isEmpty())
        return 0;
    int x = s.pop();
    if (x%6 == 0)
        return x + sum(s);
    return sum(s);
}
```

נתונה המחסנית s .

רראש המחסנית	→	12
		4
		33
		6
		30
		0

(שים לב: המשך סעיף ב בעמוד הבא).

/המשך בעמוד 14/

- עקבות אחר הפעולה `sum` בעבור המחסנית `s`, ורשום את הערך שיווחזר.
 במקבב הראה את המעבר על המחסנית `s`.
 לפניו הפעולה `sod`.

```
public static void sod(Queue<Stack<Integer>> qq, Queue<Integer> qm)
```

```
{
```

```
    if (!qq.isEmpty())
    {
        int x = sum(qq.remove());
        qm.insert(x);
        sod(qq, qm);
    }
}
```

לפניך קטע תוכנית המשתמש בפעולת `sod`.

סרטט במחברתך כל אחד משני התווים `q` ו- `r`, לפני הקראיה לפעולת `sod` ואחרי
 ביצוע הפעולה `sod`.

```
public static void main(String[] args)
```

```
{
```

```
    Stack<Integer> s1 = new Stack<Integer>();
    Stack<Integer> s2 = new Stack<Integer>();
```

```
    Queue<Stack<Integer>> q1 = new Queue<Stack<Integer>>();
```

```
    s1.push(0);
```

```
    s1.push(30);
```

```
    s1.push(6);
```

```
    s1.push(33);
```

```
    s1.push(4);
```

```
    s1.push(12);
```

```
    s2.push(23);
```

```
    s2.push(36);
```

```
    s2.push(1);
```

```
    q1.insert(s1);
```

```
    q1.insert(s2);
```

```
    Queue<Integer> qr = new Queue<Integer>();
```

```
    sod(q1, qr);
```

```
}
```

C# – לפוטרים ב-

לפניך הפעולה Sum הכתובת במחילה ראשית.

```
public static int Sum (Stack<int> s)
{
    if(s.IsEmpty())
        return 0;
    int x = s.Pop();
    if(x%6 == 0)
        return x + Sum(s);
    return Sum(s);
}
```

נתונה המחסנית s .

ראש המחסנית →	12
	4
	33
	6
	30
	0

עקב אחר הפעולה Sum בעבור המחסנית s , ורשום את הערך שיזוזר. (1)

במשך הראה את המעבר על המחסנית s .

לפניך הפעולה Sod (2)

```
public static void Sod(Queue<Stack<int>> qq, Queue<int> qm)
{
    if (!qq.IsEmpty())
    {
        int x = Sum(qq.Remove());
        qm.Insert(x);
        Sod(qq , qm);
    }
}
```

/המשך בעמוד 16/

(שים לב: המשך תתי-סעיף (2) בעמוד הבא.)

לפניך קטע תכנות המשתמש בפעולה `Sod`.
 סרטט במחברתך את כל אחד משני התווים `q` ו- `r`, לפני הקריאה לפעולה `Sod`
 ואחריו ביצוע הפעולה `Sod`.

```
public static void Main(string[] args)
{
    Stack<int> s1 = new Stack<int>();
    Stack<int> s2 = new Stack<int>();

    Queue<Stack<int>> q1 = new Queue<Stack<int>>();
    s1.Push(0);
    s1.Push(30);
    s1.Push(6);
    s1.Push(33);
    s1.Push(4);
    s1.Push(12);
    s2.Push(23);
    s2.Push(36);
    s2.Push(1);
    q1.Insert(s1);
    q1.Insert(s2);
    Queue<int> qr = new Queue<int>();
    Sod(q1 , qr);
}
```

פרק שני (50 נקודות)

בפרק זה שאלות בארבעה מסלולים:

מערכות מחשב וasmblar, עמודים 17-23.

מבוא לחקיר ביצועים, עמודים 33-24.

מודלים חישוביים, עמודים 34-37.

תכנות מונחה עצמים ב- Java , עמ' 38-45 ; תכנות מונחה עצמים ב- C# , עמודים 46-54.

ענה רק על שאלות במסלול שלמדת.

מערכות מחשב וasmblar

אם למדת מסלול זה, ענה על שתיים מהשאלות 5-8 (לכל שאלה – 25 נקודות).

5. בשאלת זו שני סעיפים א-ב, שאון קשור ביניהם. ענה על שניהם.

א. לפניך קטע תכנית באסמלר.

. לפניך ביצוע הקטע, תוכן התא שכתובתו H 200H הוא 10100001B

MOV	SI , 200H
MOV	AL , [SI]
MOV	BL , 0
MOV	CL , 8
AA:	ROR AL , 1
	JNC BB
	ADD BL , 1
BB:	DEC CL
	JNZ AA
	MOV SI , 200H
	MOV [SI + 1] , BL

(1) עקב אחר ביצוע קטע התכנית בעזרת טבלת מעקב. בטבלת המעקב פורט בכל שלב את התוכן של AL , CL ו- BL .

רשום את תוכן התא שכתובתו H 201 לשאך ביצוע קטע התכנית.

(2) מה מבצע קטע התכנית?

(3) שינו את קטע התכנית באופן זהה:

במקום ההוראה הששית JNC BB , נ כתבה ההוראה JC BB

מה יהיה תוכן התא שכתובתו H 201 עם סיום הביצוע של

קטע התכנית לאחר השינוי?

(שים לב: סעיף ב של השאלה בעמוד הבא.)

.ב. (אין קשר לסעיף א.)

נתון קטע תוכנית באסמבולר.

לפני ביצוע קטע התוכנית, תוכנו של האוגר AX היה H 1A2BH .

MOV	CX , 8
ROR	AX , CL

(1) מה יכול האוגר AX לאחר הביצוע של קטע התוכנית? רשום את תשובתך
במספרים הקסדצימליים.

(2) רשום הוראה אחת שתבצע את מה שمبرצע קטע התוכנית הנתון.

6. בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שנייהם.
 א. לפני קטע תוכנית באSEMBLER.

לפני הביצוע של קטע התוכנית, התוכן של האוגר AX הוא 42H .

MOV	CL , 2
PUSH	AX
SHL	AX , CL
POP	BX
ADD	AX , BX

- (1) עקב בעורת טבלת מעקב אחר הביצוע של קטע התוכנית, ורשום את התוכן של האוגרים AX ו- BX לאחר הביצוע של קטע התוכנית.
 (2) רשום הוראה אחת שתבצע את מה שמבצע קטע התוכנית שלפניך. הנה שהתוכן של האוגר DL הוא 05H .

ב. (אין קשר לסעיף א.).

במקטע הנתונים הוגדר המשתנה

X DB ?

שבו מאוחסן מספר שערכו בין 1 ל- 18 (כולל).

רוצים לכתוב תוכנית שתאחסן באוגר AL את מספר המספרים הדוא-ספרתיים שסכומם הספרות של כל אחד מהם שווה לערך המאוחסן במשתנה X .

לדוגמה:

אם במשתנה X מאוחסן הערך 15 , באוגר AL יאוחסן הערך 4 , כי יש 4 מספרים דוא-ספרתיים שסכומם הספרות של כל אחד מהם הוא 15 . מספרים אלה הם:
 .96, 87, 78, 69

(שים לב: המשך השאלה בעמוד הבא.)

קטע התכנית שלפניך, הכתוב ב- Java ובי- C#, מבצע את הנדרש, בלי לפרק מספר דו-ספרתי לספרות.

```
int a = 0;
for (int i = 1 ; i <= 9 ; i++)
    for (int j = 0 ; j <= 9 ; j++)
        if (i + j == x)
            a++;
```

קטע התכנית תורגם לקטע תכנית באסמבול.

בקטע התכנית באסמבול הושמו 4 הוראות המסומנות **i** – **iv**.

רשום במחברתך את מספרי ההוראות החסרות, וליד כל אחד מן המספרים כתוב באסמבול את הוראה החסורה כך שקטע התכנית יבצע את הנדרש.

MOV AL ,0

i _____

MOV DH ,1

A1: MOV BX ,CX

ii _____

MOV DL ,0

A2: MOV AH ,DH

iii _____

CMP AH ,X

JNE A3

INC AL

A3: INC DL

LOOP A2

iv _____

INC DH

LOOP A1

בשאלה זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שנייהם.

- א. השגורה (פרוצדורה) CHECK בודקת אם מילה (WORD) המועברת כפרמטר דורך המחסנית, מהויה שני מספרים עוקבים כך שהבית הנמור של המילה קטן ב- 1 מהבית הגבואה של המילה. למשל: 1 , 2 , או 203 , 204 . אם כן, השגורה מאחסנת 1 באוגר AL , אחרת — היא מאחסנת בו 0 . לאחר מכן השגורה חוזרת למקום שמננו נקרה. לפניך ארבעה קטעי תוכנית, i-iv , הכתובים באסמבול. עקוב בעזרת טבלת מעקב אחר כל אחד מן הקטעים, וקבע אם הוא מבצע את הנדרש או אינו מבצע את הנדרש.

i CHECK: POP CX

```
POP BX
XOR AL,AL
ADD BH,1
CMP BH,BL
JZ A1
INC AL
A1: PUSH CX
RET
```

ii CHECK: POP BX

```
POP CX
MOV AL,0
DEC CL
SUB CH,CL
JNZ A1
INC AL
A1: PUSH BX
RET
```

(שים לב: המשך סעיף א' וסעיף ב' בעמוד הבא.)

iii CHECK: POP CX
 POP BX
 MOV AL ,0
 SUB CL ,1
 CMP BH ,BL
 JNE A1
 MOV AL ,1
 A1: PUSH CX
 RET

iv CHECK: POP BX
 POP CX
 MOV AL ,1
 ADD CL ,1
 CMP CH ,CL
 JE A1
 DEC AL
 A1: PUSH BX
 RET

ב. (אין קשר לסעיף א).

לפניך קטע תכנית הכתוב ב- Java וב- C# .

x ו- y הם משתנים מטיפוס שלם.

```
while ((x > 0) && (y > 0))
{
  if(x > 10)
    x--;
  else
    y--;
}
```

כתב באסמבילר קטע תכנית שיבצע את קטע התכנית הכתוב ב- Java וב- C# .

הנה שהמשתנה x מאוחסן באוגר BX, והמשתנה y מאוחסן באוגר CX .

/המשך בעמוד 23/

.8. מערך חד-ממדי ARR בגודל N של מספרים שלמים — כאשר N זוגי — ייקרא **מאוזן** אם: הסכום של הערך המאוחסן בתא הראשון של המערך ושל הערך המאוחסן בתא האחרון של המערך שווה לסכום של הערך המאוחסן בתא השני של המערך ושל הערך המאוחסן בתא שלפניו האחרון במערך, ושווה לסכום של הערך המאוחסן בתא השלישי של המערך ושל הערך המאוחסן בתא השלישי מסוף המערך, וכן הלאה. וסכום זה שווה לערך המאוחסן במשתנה x.

לדוגמה:

בעבור מערך ARR בגודל 6 ומשתנה x שהערך המאוחסן בו הוא 10,

ARR	3	1	4	6	9	7
	0	1	2	3	4	5

הערך ARR **מאוזן** כי מתקיים: $3 + 7 = 1 + 9 = 4 + 6 = 10$.

.א. כתוב באסמבילר שגורה (פרוצדורה) בשם CHECK, שתקבל כפרמטרים שלוש כתובות של תאים בזיכרון. השגורה תבדוק אם סכום הערכים המאוחסנים בהתאם לשתי הכתובות הראשונות שווה לערך המאוחסן בתא שבכתובת השלישי. אם כן — השגורה תיחסן 1 באוגר DL, אחרת — היא תיחסן בו 0.

.ב. במקטע הנתונים מוגדרים מערך ARR ומשתנה x:
 ARR DB 100 DUP (?)
 X DB ?

כתב באסמבילר קטע תכנית שיבדוק אם המערך ARR הוא **מאוזן**.

אם כן — קטע התכנית ייחסן 1 באוגר DH, אחרת — הוא ייחסן בו 0.
 הנח שסכום הערכים של שני תאים איןנו עולה על 255.
 עליך להשתמש בשגורה CHECK שכתובה בסעיף א.

מבוא לחקור ביצועים

אם למדת מסלול זה, ענה על שתיים מהשאלות 9-12 (לכל שאלה – 25 נקודות).

9. בשאלת זו שלושה סעיפים, א-ג, שאין קשר ביניהם. ענה על שלושת הסעיפים.

א. נתונה בעיית תכנון ליניארי, ונתונים האילוצים:

$$2x_1 - x_2 \leq 6$$

$$x_2 \leq 4$$

$$x_1 \geq 4$$

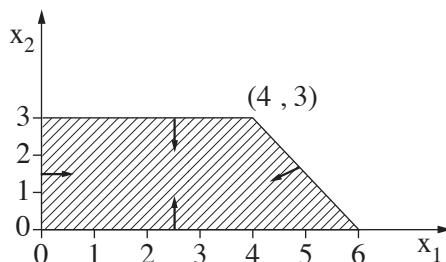
$$x_2 \geq 0$$

סרטט את תחום הפתרונות האפשריים הנקבע על סמך האילוצים הנתונים.

סמן את תחום הפתרונות האפשריים של האילוצים,

ורשום בגרף שמתקבל את הערכיהם של x_1 ו- x_2 בעבר כל אחד מן הקדקודים של תחום הפתרונות האפשריים.

ב. נתון סרטוט של תחום הפתרונות האפשריים של בעיית תכנון ליניארי מסוימת.



נתונה פונקציית המטרה: $(1) \text{ Max } \{Z = 3x_1 + 2x_2\}$.

אם לבעיה הנתונה יש אינסוף פתרונות בתחום הפתרונות האפשריים הנתון, רשות את הפתרון הכללי, ואת ערכה של פונקציית המטרה בפתרון האופטימלי.

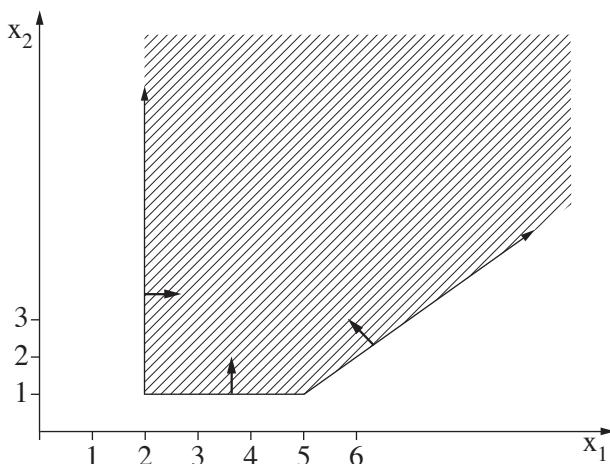
אחרת – רשות את הפתרון היחיד, ואת ערכה של פונקציית המטרה בפתרון זה.

נתונה פונקציית המטרה: $(2) \text{ Max } \{Z = kx_1 + x_2\} \text{ ו- } k > 0$.

קבע בעבר אילו ערכים של k יהיו לבעיה התכנון הליניארי הנתונה בתחום הפתרונות האפשריים הנתון אינסוף פתרונות אופטימליים. כתוב את הערך של פונקציית המטרה בעבר כל אחד מערכי $\{-k\}$ האלה. נמק את קביעותיך.

(שים לב: סעיף ג של השאלה בעמוד הבא.)

ג. נתון סרטווט של תחום הפתרונות האפשריים של בעיית תכנון לינארי, שאחד האילוצים שלו הוא: $x_1 - x_2 \leq 4$.



- (1) האם בתחום הפתרונות האפשריים הנתון יש פתרון אופטימלי לפונקציית המטרה $Z = x_1 - x_2$? נמק את תשובה.
- (2) האם בתחום הפתרונות האפשריים הנתון יש פתרון אופטימלי לפונקציית המטרה $Z = x_1 - x_2$? נמק את תשובה.

- . בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניהם.
- א. הוא גраф לא מכוון המוצג על ידי מטריצת הסמיכות שלפניך: $G = (V, E)$

	a	b	c	d	e
a	0	1	0	1	0
b	1	0	1	0	1
c	0	1	0	1	0
d	1	0	1	0	1
e	0	1	0	1	0

- (1) סרטט את הגרף G המוצג על ידי המטריצה הנתונה.
- (2) האם הגרף G הוא גרף שלם? נמק את תשובתך.
- (3) כמה רכיבי קשריות (Connected Components) יש בגרף הנתון, ומה הם?
- (4) מצא שני מסלולים קצרים ביותר מקדקוד a לקדקוד e.
- (5) תאor בלאחד מן המסלולים תיאור סכמטי.
תאor מעגל זה תיאור סכמטי.

(שים לב: סעיף ב של השאלה בעמוד הבא.)

.ב. (אין קשר לסעיף א).

$G = (V, E)$ הוא גראף מכוון המיוצג על ידי רשימה הסמיכויות שלפנינו:

a	$\rightarrow e \rightarrow d \rightarrow \parallel$
b	$\rightarrow a \rightarrow \parallel$
c	$\rightarrow d \rightarrow \parallel$
d	$\rightarrow c \rightarrow \parallel$
e	$\rightarrow b \rightarrow \parallel$

(1) סרטט את הגרף G המיוצג על ידי הרשימה הנתונה.

(2) כמה רכיבי קשריות חזקה (רק"חים — יש בגראף G

ומה הם?

(3).i. העתק למחברתך את הטבלה שלפנינו.

הפעל אלגוריתם סריקה לעומק (DFS) על הגרף G , החל מקדקוד a.

בפעולת האלגוריתם DFS חישב בעבר כל קדקוד u את $[u]d$ ואת $[u]f$.

$[u]d$ מציין את מועד הגילוי של הקדקוד u בועת הסריקה,

ור $[u]f$ מציין את המועד של סיום הטיפול בקדקוד u.

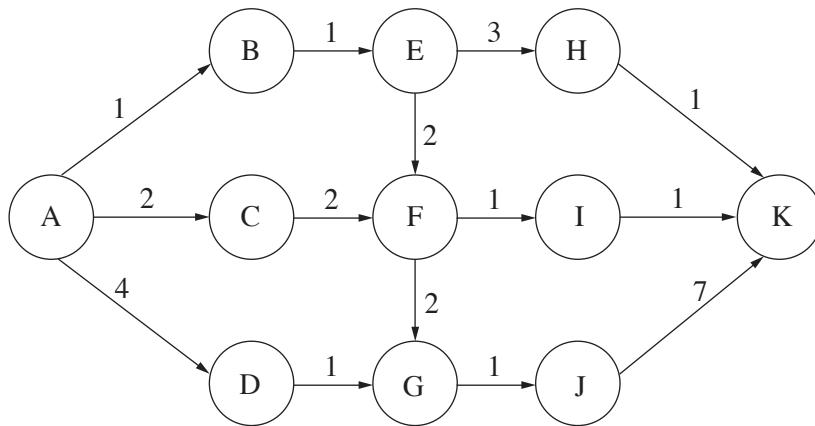
כתבו את תוצאות החישובים בטבלה שבמחברתך.

קדקוד u	a	b	c	d	e
$d[u]$					
$f[u]$					

.ii. סרטט במחברתך רק את העץ הפורש (DFS)/היער הפורש (DFS) שמתקיים.

התבוסס על הייצוג הנוכחי ברשימה הסמיכויות.

. א. לפניך רשת (V, E) .



מצא את כל המסלולים הקצרים ביותר מקדקוד A לקדקוד K בראשת הנתונה.
תאר כל אחד מן המסלולים תיאור סכמתי.

. ב. נוסף לרשת G , נתון עץ פורש מכוכן T שמחבר את כל קדקודיו הרשת ואינו מכיל מעגלים.

לפניך אלגוריתם:

צעד 1: הפעיל את האלגוריתם של דיקסטרה מקדקוד נתון בעץ T .

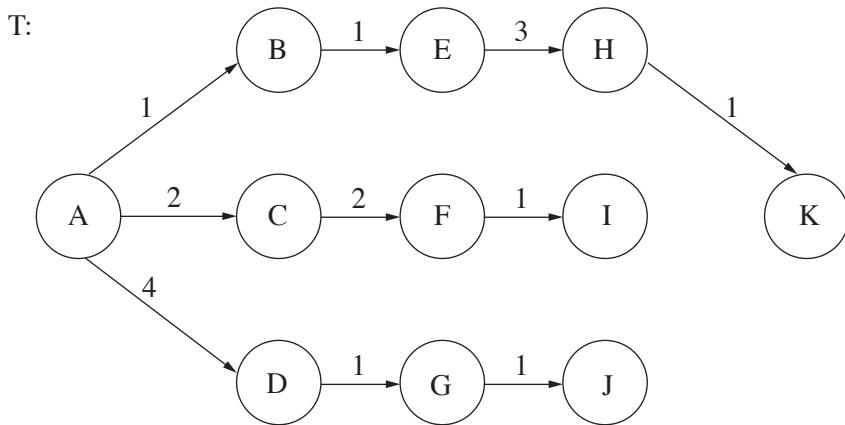
צעד 2: עבור כל קשת מכוכנת (u,v) , השיכת ל- E בראשת G וaina השיכת לעץ T , בצע:

אם $d[v] > d[u] + w(e)$ — החזר "שקר".

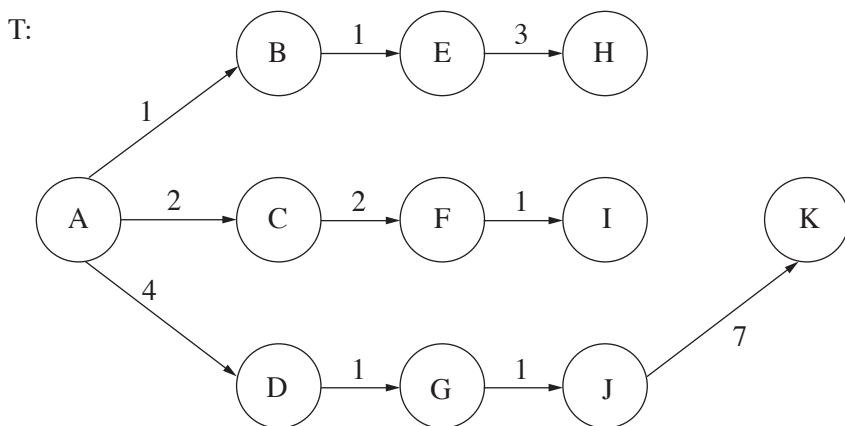
((v) מציין את אורך המסלול הקצר ביותר מקדקוד מקור לקדקוד v .)
צעד 3: החזר "אמת".

(שים לב: המשך סעיף ב בעמוד הבא.)

- (1) מהו הערך המוחזר לאחר הפעלת האלגוריתם על הרשות הנתונה G ועל העץ T שלפניך, החל מקדוק A ? כתוב את כל הצעדים של האלגוריתם ואת התוצאה המתתקבלת מכל צעדי.



- (2) מהו הערך המוחזר לאחר הפעלת האלגוריתם על הרשות הנתונה G ועל העץ T שלפניך, החל מקדוק A ? כתוב את כל הצעדים של האלגוריתם ואת התוצאה המתתקבלת מכל צעדי.



12. בשאלת זו חמישה סעיפים, א-ה, שאין קשר ביניהם. עונה על כל הסעיפים.

א. בטבלה של פניר נתון חלק מפתרון בסיסי אפשרי לבועית התובלה: $0 = x_{22}$.

מקורות	יעדים			היצוא
	1	2	3	
1	10	0	20	20
2	12	7	9	120
3	0	14	16	60
בקיש	20	0	180	

(1) העתק למחברתך את הטבלה, והשלם בה את הפתרון הבסיסי האפשרי לפי שיטת הפינה הצפוןית-מערבית.

(2) משנים את המחיר בתא (1), מ- 0 ל- 15. האם ישנה הפתרון הבסיסי האפשרי שמצוות בתת-סעיף (1) ? נמק.

ב

ב

 בטבלה שלפניך נתון פתרון בסיסי אפשרי לבעיהת תובלה, ונתון הערך של E_1 .

מקורות	יעדים			היצע	u_i
	1	2	3		
1	5	8	10	5	0
2	5	20	12	8	
3	1	13	14	16	
ביקוש	12	5	12		
v_j					

(1) העתק למחברתך את הטבלה, והשלם בה את הערכיהם של v_3 , v_2 , v_1 , u_3 , u_2 , u_1 .

(ג) הסבר מדוע הכתרוו הנטו אינן כתרוו אובייקטימי.

(שים לב: המשך השאלה בעמוד הבא).
/המשך בעמוד הבא | המהלך יופיע מחר.

ג. בטבלה שלפניך נתון חלק מפתרון בסיסי אפשרי לבועית תובלה, ונתונים הערכים של $u_1, u_2, u_3, v_1, v_2, v_3, v_4$, שמתאימים לפתרון זה.

מקורות	יעדים				היצוא	u_i
	1	2	3	4		
1	8	9	4	6	280	0
2	4	12	13	8	160	-10
3	15	10	40	12	270	1
ביקוש	200	320	160	30		
v_j	14	9	4	6		

העתק למחברתך את הטבלה, והשלם בה את הפתרון, בהתחשב בערכים של $u_1, u_2, u_3, v_1, v_2, v_3, v_4$ כך שיתקבל פתרון בסיסי אפשרי.

(שים לב: המשך השאלה בעמוד הבא.)

ד. בטבלה שלפניך נתון פתרון לא אופטימלי שהתקבל לאחר k איטרציות בעבר בעיית תובלה

מסויימת, ונתונים הערכים של $u_1, u_2, u_3, v_1, v_2, v_3$.

מקורות	יעדים			היצע	u_i
	1	2	3		
1	22	15	17	200	0
2	10	18	4	100	-15
3	20	10	8	150	-5
ביקוש	200	150	100		
v_j	25	15	17		

עליך לבצע איטרציה נוספת, ככלומר איטרציה $k + 1$.

(1) מהו המשנה שיוצא מן הבסיס באיטרציה זו?

(2) סרטט במחברתך טבלה חדשה, ורשום בה את הפתרון שיתקבל לאחר איטרציה זו.

(שים לב: סעיף ה של השאלה בעמוד הבא.)

/המשך בעמוד 33/

ה.

לחברה ביינלאומית גדולה יש 3 מפעלים A, B, C, המסוגלים לייצר 120 מיליון, 80 מיליון, 50 מיליון פרוסות סיליקון בשנה בהתאם. כדי לצרוכו שבבים מפרוסות הסיליקון מועברות הפרושות ל- 3 מפעלים אחרים 1, 2, 3. כושר העבודה של מפעלים אלה הוא 150 מיליון, 80 מיליון ו- 70 מיליון פרוסות סיליקון בשנה בהתאם. מפעל 1 חייב לקבל את כל הכמות המתאימה לכושר העבודה שלו. הבעיה הנתונה אינה "מאוזנת", שכן הציג סטודנט להוסיף "מקור דמיוני" D אשר יספק 50 מיליון פרוסות סיליקון בשנה. הבעיות היחידות מכל מקור לכל יעד נתונות בטבלה ש לפניה.

מקורות	יעדים			הציג
	1	2	3	
A	25	18	18	120
B	34	15	32	80
C	58	24	18	50
D	x	y	z	50
ביקוש	150	80	70	

ערך הבעיות של המקור D נתונים בטבלה כפרמטרים x, y, z. מה צריך להיות הערך של כל אחד מן הפרמטרים x, y, z כדי שייהי אפשר למצוא פתרון בסיסי אפשרי בעיית התובלה הנתונה? בחר בתשובה הנכונה מרבע התשובות שלפניך, העתק אותה למחברתך, ונמק את בחירתך. (M) מוגדר מספר גדול מאוד).

$$x = 0, y = 0, z = 0 \quad .1$$

$$x = M, y = M, z = M \quad .2$$

$$x = M, y = 0, z = 0 \quad .3$$

$$x = 0, y = M, z = M \quad .4$$

מודלים חישוביים

אם למדת מסלול זה, ענה על שתיים מהשאלות 13-16 (לכל שאלה – 25 נקודות).

13. בנֶה מכונת טיורינג שעלה סרט הזיכרון שלה כתובה מילה w מעל הא"ב $\{a\}$.

w כתובה מתחילה הסרט.

המכונה תכתוב על הסרט מתחילה את המילה $w?w$.

אם w היא המילה הrikha, המכונה תכתוב על הסרט מתחילה את המילה ?.

לדוגמה:

לפני פעולה המכונה הסרט נראה כך:

—	a	a	a	Δ	Δ	Δ
---	---	---	---	---	---	---	-------

לאחר פעולה המכונה ייראה הסרט כך:

—	a	a	a	?	a	a	a	Δ	Δ	Δ	Δ
---	---	---	---	---	---	---	---	---	---	---	---	-------

בשאלה זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שנייהם.

- א. לפניך השפה L מעל הא"ב $\{0, 1\}$. L היא אוסף המילים שבהן רצף התווים 01 מופיע לפחות פעמיים זהה למספר הפעמים שמוופיע רצף התווים 10.

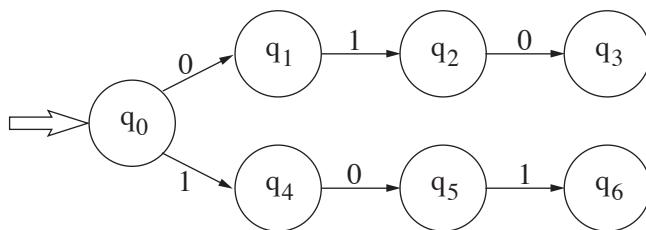
דוגמה למילים השויות לשפה L :

111, 0, 010, 0111010, 1110001000110101

דוגמה למילים שאינן שייכות לשפה L :

011, 0101, 100

לפניך סרטוט חלקי של אוטומט סופי דטרמיניסטי המקבל את השפה L .
בסרטוט חסרים מערבים, סימני קלט ומצבים מקבלים.



הסרטוט כולל את כל המ מצבים של האוטומט.

העתק למחברתך את הסרטוט, והשלם אותו כך שהאוטומט יקבל את השפה L .
עליך להשלים את המעברים החסרים ואת סימני הקלט החסרים, ולסמן את
כל המ מצבים מקבלים.

שים לב: אין להוסיף לאוטומט מצבים, ואין להוריד ממנו מצבים.

ב. (אין קשר לסעיף א).

(1) הבא דוגמה לשלווש שפות L_1 , L_2 , L_3 כך ש: $L_1 \cap L_2 = L_3$ והן שפות רגולריות,
� L_3 היא שפה לא רגולרית, ומתקיים $L_1 \cup L_2 = L_3$.

(2) הבא דוגמה לשלווש שפות L_1 , L_2 , L_3 כך ש: $L_1 \cap L_2 = L_3$ והן שפות רגולריות,
� L_3 היא שפה לא רגולרית, ומתקיים $L_1 \cup L_2 = L_3$.

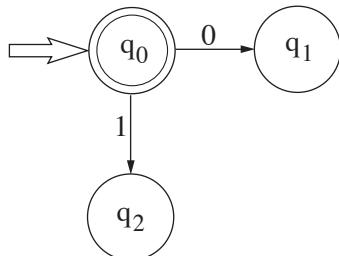
15. בשאלת זו שלושה סעיפים, א-ג, שאין קשר ביניהם. ענה על שלושת.

א. לפניך השפה L מעל הא"ב $\{0, 1\}$:

$$L = \left\{ w \mid \begin{array}{l} \text{שארית החלוקת ב-3 של מספר ה-0ים ב-} w \\ \text{שווה לשארית החלוקת ב-3 של מספר ה-1ים ב-} w \end{array} \right\}$$

לפניך סרטוט חלקו של אוטומט סופי דטרמיניסטי מקבל את השפה L .

בסרטוט חסרים מעברים וסימני קלט.



הסרטוט מכיל את כל המ מצבים של האוטומט ואת כל המ מצבים מקבלים.

העתק למחברתך את הסרטוט, והשלם אותו כך שהאוטומט יקבל את השפה L .

עליך להשלים את המעברים החסרים ואת סימני הקלט החסרים.

שים לב: אין להוסיף לאוטומט מצבים ואין להוריד מהם מצבים.

ב. לפניך השפות L_1 ו- L_2 מעל הא"ב $\{0, 1\}$:

$$L_1 = \{0^n \cdot x \mid \{0, 1\}^* \text{ מילה מעל הא"ב}, \quad 1 \leq |x| \leq n\}$$

$$L_2 = \{0^m \mid m \geq 1\}$$

נתונות שתי מילים ב- L_2 : $w_1 = 0^i$ ו- $w_2 = 0^j$ כך ש: $i < j$.

מצא מילה w כך שיתקיים:

$$w_1 \cdot w \in L_1$$

$$w_2 \cdot w \notin L_1$$

(שים לב: סעיף ג' של השאלה בעמוד הבא.)

ג. לפניך השפות L_1 ו- L_2 מעלה'a'b : {a,b,c}

$$L_1 = \{a^n b^m c^{n+m} \mid n \geq 1, m \geq 1\}$$

$$L_2 = \{a^s \mid s \geq 1\}$$

נתונות שתי מיללים ב- $w_1 = a^j$ ו- $w_2 = a^i$: L_2 כר' ש:

מצאה מילה w כר' שיתקיים:

$$w_1 \cdot w \in L_1$$

$$w_2 \cdot w \notin L_1$$

.16. לפניך השפה L מעלה'a'b : {a,b,c}

$$L = \{a^{i-j} b^j c^j \mid j \geq 1, i-j \geq 0\}$$

א. כתוב את המילה הקצרה ביותר בשפה L .

ב. בנה אוטומט מחסנית שיקבל את השפה L .

תכנות מונחה עצמים

אם למדת מסלול זה ואתה כותב ב- Java, ענה על שתיים מהשאלות (לכל שאלה – 25 נקודות)

.17. בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שנייהם.

א. לפני שתי מחלקות A, B :

```
public class A
{
    public A() {
        System.out.println( "A" );
    }
    public void one() {
        System.out.println( "one of A" );
    }
    public void two() {
        one();
    }
}
public class B extends A
{
    public B() {
        System.out.println( "B" );
    }
    public void one() {
        System.out.println( "one of B" );
    }
    public void two() {
        System.out.println( "two of B" );
    }
    public void three() {
        super.two();
    }
}
```

(**שים לב: המשך סעיף א בעמוד הבא.**)
/המשך בעמוד 39/

לפניך שבעה קטעי תוכנית, ו-iii. לכל אחד מהם קבע אם הוא תקין או אינו תקין.
 אם הקטע תקין — כתוב מהו הפלט שיתקבל מהרצאת קטע התוכנית.
 אם הקטע אינו תקין — נמק את קביעתך, וקבע אם יש שגיאת הידור (קומפילציה)
 או שגיאת זמן ריצה.

i A a1 = new A();

a1 . two();

((A) a1) . two();

ii A a2 = new A();

((B) a2) . two();

iii A a3 = new B();

a3 . two();

((B) a3) . three();

iv B b1 = new A();

b1 . two();

v B b2 = new B();

b2 . two();

((A) b2) . two();

vi B b3 = new B();

((A) b3) . three();

vii B b5 = new B();

((B) b5) . three();

(שים לב: סעיף ב של השאלה בעמוד הבא.)

/המשך בעמוד הבא/

(א) אין קשר לסעיף א). ב.

לפניך חתימה של המחלקות AAA, BBB וכוננותיהן:

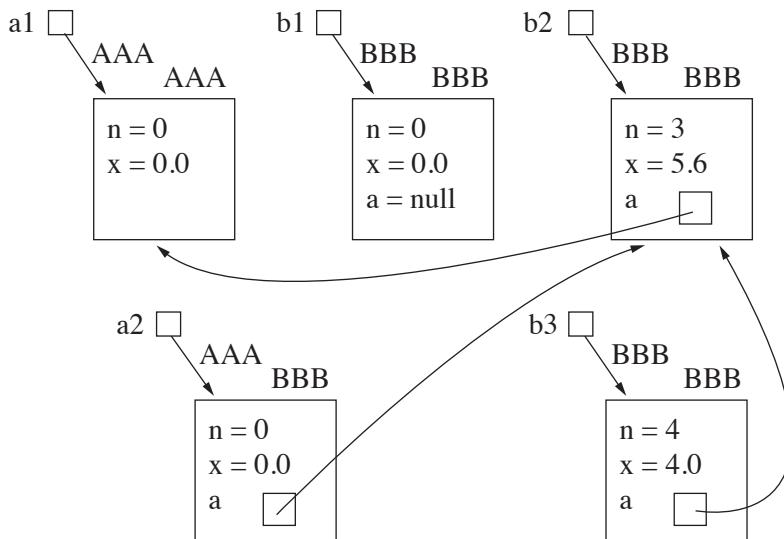
```
public class AAA
{
    private int n ;
    private double x ;
}

public class BBB extends AAA
{
    private AAA a;
}
```

לפניך המחלקה Test, ובה פעולה ראשית:

```
public class Test
{
    public static void main (String[] args )
    {
        AAA a1 = new AAA();
        BBB b1 = new BBB();
        BBB b2 = new BBB(3 , 5.6 , a1);
        AAA a2 = new BBB(b2);
        BBB b3 = new BBB(4 , b2);
    }
}
```

הרצת הפעולה הראשית main יוצרה את העצמים الآלה:



ממש ב- Java במחלקות BBB, AAA את הפעולות הבונות הנדרשות להרצת הפעולה main כך שיתקבלו העצמים המתוארים.

- .18. ב שאלה זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניהם.
- א. לפניך המחלקות C, D, UseCD. רשם את הפלט של הפעולה הראשית.

```

public class C {
    public C() {
        System.out.println("C");
    }
    public C(int n) {
        for (int i = 0; i < n; i++)
            new C();
        System.out.println("CC");
    }
}
public class D extends C {
    public D() {
        super();
        System.out.println("D");
    }
    public D(int n) {
        super(n);
        for (int i = 0; i < n; i++)
            new D();
    }
}
public class UseCD {
    public static void main(String[] args) {
        System.out.println("***");
        C c1 = new C(3);
        System.out.println("---");
        D d1 = new D(1);
        System.out.println("###");
        D d2 = new D(4);
    }
}

```

(שים לב: סעיף ב של השאלה בעמוד הבא.)

.ב. אין קשר לסעיף א.)

לפניך המחלקות AA, BB, ופעולה ראשית במחלקה Run. עקוב אחר הפעולה הראשית, ורשות את הפלט המתתקבל. במקבץ יש לכתוב את ערכי המשתנים, את ערכי תוכנות המחלקה (תוכנות סטטיות), ובמעבר כל עצם יש לכתוב את ערכי התוכנות שלו.

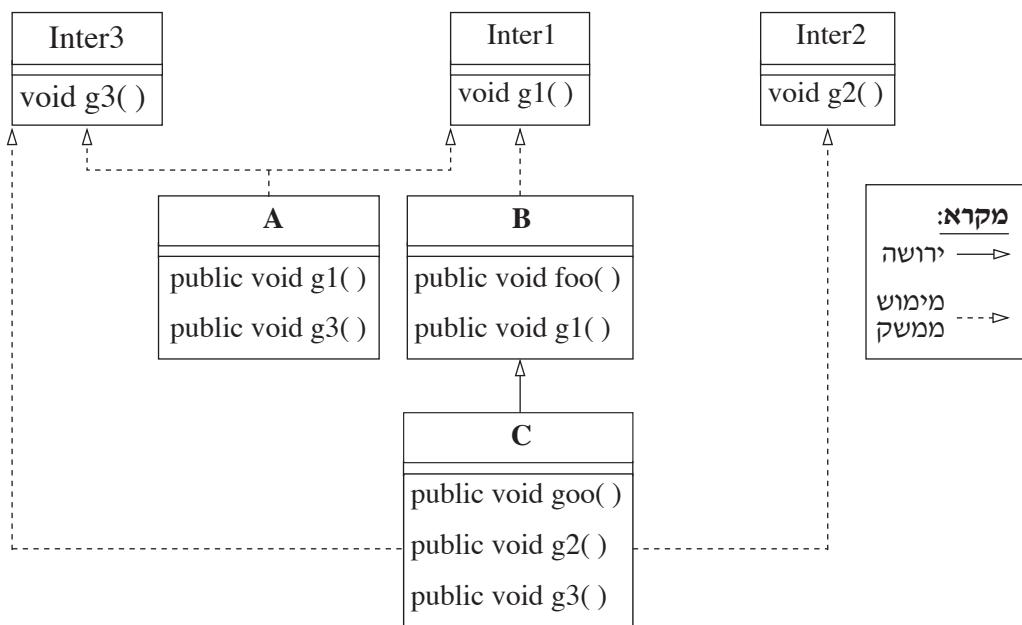
```
public class AA
{
    private int num1 = 0;
    private int num2 = 0;
    private static int count = 0;
    public AA(int num)
    {
        num1 = num;
        num2 = num;
        count++;
        System.out.println("AA ctor1");
    }
    public AA(int num1, int num2)
    {
        this.num1 = num1;
        this.num2 = num2;
        count++;
        System.out.println("AA ctor2");
    }
    public int sum()
    {
        return num1 + num2;
    }
    public static int getCount()
    {
        return count;
    }
}
```

```
public class BB extends AA
{
    private int num3 = 0;
    public BB(int num)
    {
        super (num);
        num3 = num;
        System.out.println("BB ctor1");
    }
    public BB(int num1, int num2, int num3)
    {
        super (num1, num2);
        this.num3 = num3;
        System.out.println("BB ctor2");
    }
    public int sum()
    {
        return super.sum() + num3;
    }
    public void setNum3(int num)
    {
        num3 = num;
    }
}
```

(שים לב: המשך סעיף ב בעמוד הבא)

```
public class Run
{
    public static void main(String[] args)
    {
        AA f1 = new AA(10);
        System.out.println("sum1 = " + f1.sum());
        System.out.println("count = " + AA.getCount());
        AA f2 = new AA(10, 20);
        System.out.println("count = " + AA.getCount());
        BB s1 = new BB(1);
        AA f3 = new BB(2);
        System.out.println("count = " + AA.getCount());
        f2 = s1;
        System.out.println("sum = " + f2.sum());
        s1.setNum3(2);
        System.out.println("sum = " + s1.sum());
        System.out.println("sum = " + f2.sum());
    }
}
```

. 19. לפניך תרשים UML חלקו.



. א. השלים את שלוש השורות החסרות כך שקטוע התכנית שלפניך יעבד בלי שגיאות

הידור (קומpileציה) ו/או זמן ריצה.

העתק למחברתך את השורות שהשלמות.

```

_____ x = new A();
_____ y = new B();
_____ z = new C();

x = y;
y = new C();
y.foo();
z.g3();
((C)z).goo();
z = new A();
  
```

. ב. כתוב ב- Java פעולה שתתקבל מערך חד-ממדי מטיפוס Inter1

הפעולה תעבור על איברי המערך, ותקרו לפעולה () g3 אם היא קיימת בעבור העצם.

אם הפעולה () g3 אינה קיימת — הפעולה שכותבת תקרו לפעולה () g1.

. type3 , type2 , type1 .²⁰ חברות מסוימות מייצרת שלושה סוגי טלפונים סלולריים:

טלפון מסוג type1 אפשר לבצע את הפעולות האלה:

- חיווג — dial
- קבלת שיחה — recCall
- שליחת הודעה — sendMsg
- קבלת הודעה — recMsg

טלפון מסוג type2 אפשר לבצע את הפעולות האלה:

- חיווג — dial
- קבלת שיחה — recCall
- שליחת הודעה — sendMsg
- קבלת הודעה — recMsg
- צילום תמונה — takePic
- הצגת תמונה — showPic

טלפון מסוג type3 אפשר לבצע את הפעולות האלה:

- חיווג — dial
- קבלת שיחה — recCall
- צילום תמונה — takePic
- הצגת תמונה — showPic
- התקנת אפליקציות — installApp

בכל אחד מסוגי הטלפונים:

נשמר מספר דקוט השיחה שנעשו מן הטלפון.

אפשר לאפס את מספר דקוט השיחה — . reset

אפשר להזיר חשבון טלפון עדכני — bill , המוחשך כך: מספר דקוט השיחה × 1 ש"ח.

טלפונים שיש בהם אפשרות לשלוח הודעות, החזרת חשבון הטלפון העדכני היא:

מספר דקוט השיחה × 1 ש"ח + מספר ההודעות שנשלחו × 0.5 ש"ח.

טלפונים האלה פועלות האיפוס מאפסת את מספר דקוט השיחה וכן את מספר ההודעות שנשלחו.

a. בעבור הטלפונים הסלולריים המתוארים לעיל, סרטט את היררכיית המחלקות והמשקים, באופן המתאים ביותר לעקרונות של תוכנות מונחה עצמים (הכמסה — inheritance , הורשה — encapsulation , פולימורפיזם — polymorphism). סמן בסרטוט את הקשרים בין המחלקות, ואת הקשרים בין המחלקות ובין המשקים. אין צורך לככלו הסרטוט תוכנות ופעולות.

b. לכל אחת מן המחלקות שבסרטוט כתוב ב- Java את כוורת המחלקה ואת התכונות שלה, וכותרות לכל הפעולות.

לכל אחד מן המשקים שבסרטוט כתוב ב- Java את הכוורת שלו, וכותרות לכל הפעולות.

אין צורך לכתוב את הפעולות get ו- set ופעולות בונות.

ממש ב- Java את הפעולה המחזירה את החשבון העדכני בכל אחת מן המחלקות שהיא מופיעה בהן.

תכנות מונחה עצמים

אם למדת מסלול זה ואתה כותב ב- C#, ענה על שתיים מהשאלות 21-24, (כל שאלה – 25 נקודות)

.21 בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שנייהם.

א. לפניך שתי מחלקות, A, B :

```
public class A
{
    public A() {
        Console.WriteLine( "A" );
    }
    public virtual void One() {
        Console.WriteLine( "One of A" );
    }
    public virtual void Two() {
        One();
    }
}
public class B : A
{
    public B() {
        Console.WriteLine( "B" );
    }
    public override void One() {
        Console.WriteLine( "One of B" );
    }
    public override void Two() {
        Console.WriteLine( "Two of B" );
    }
    public void Three() {
        base.Two();
    }
}
```

(שים לב: המשך סעיף א בעמוד הבא.)

לפניך שבעה קטיעי תכנית, i-vii. לכל אחד מהם קבוע אם הוא תקין או אינו תקין.

אם הקטע תקין — כתוב מה הפלט שיתקבל מהרצת קטע התכנית.

אם הקטע אינו תקין — נמק את קביעתך, וכותב אם יש שגיאת הידור (קומפילציה) או שגיאת זמן ריצה.

- i A a1 = new A();
a1 . Two();
((A) a1) . Two();

- ii A a2 = new A();
((B) a2) . Two();

- iii A a3 = new B();
a3 . Two();
((B) a3) . Three();

- iv B b1 = new A();
b1 . Two();

- v B b2 = new B();
b2 . Two();
((A) b2) . Two();

- vi B b3 = new B();
((A) b3) . Three();

- vii B b5 = new B();
((B) b5) . Three();

.ב. (אין קשר לסעיף א.)

לפניך חתימה של המחלקות AAA, BBB וכוכנותיהן:

```
public class AAA
{
    private int n ;
    private double x ;
}

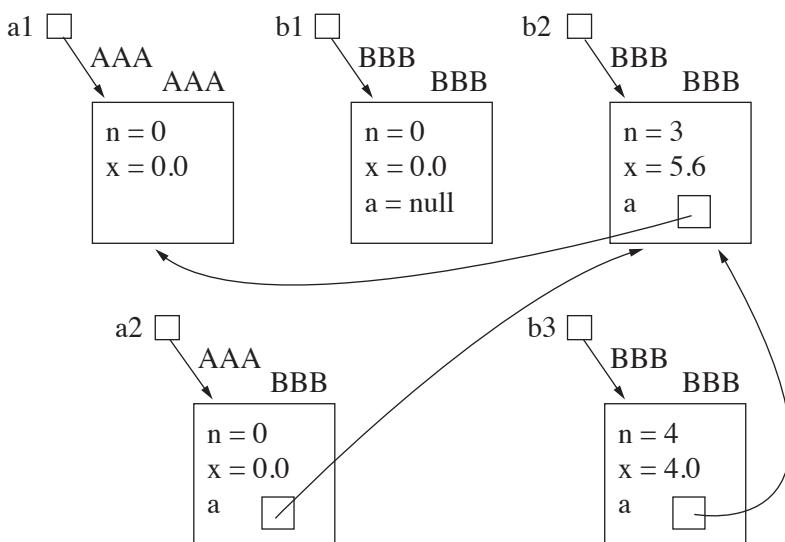
public class BBB : AAA
```

```
{    private AAA a;
```

לפניך המחלקה Test, ובה פעולה ראשית:

```
public class Test
{
    public static void Main()
    {
        AAA a1 = new AAA();
        BBB b1 = new BBB();
        BBB b2 = new BBB(3, 5.6, a1);
        AAA a2 = new BBB(b2);
        BBB b3 = new BBB(4, b2);
    }
}
```

הרצת הפעולה הראשית Main יוצרה את העצמים האלה:



ממש ב- C# במחלקות BBB, AAA את הפעולות הבוניות הנדרשות להרצת הפעולה Main כך שיתקבלו העצמים המתוארים.
המשך בעמוד 49/

.22. בשאלת זו שני סעיפים, א-ב, שאין קשר ביניהם. ענה על שניהם.

א. לפניך המחלקות C, D, UseCD. רשום את הפלט של הפעולה הראשית.

```

public class C {
    public C() {
        Console.WriteLine("C");
    }
    public C(int n) {
        for (int i = 0; i < n; i++)
            new C();
        Console.WriteLine("CC");
    }
}
public class D : C {
    public D() : base() {
        Console.WriteLine("D");
    }
    public D(int n) : base(n) {
        for (int i = 0; i < n; i++)
            new D();
    }
}
public class UseCD {
    public static void Main() {
        Console.WriteLine("****");
        C c1 = new C(3);
        Console.WriteLine("---");
        D d1 = new D(1);
        Console.WriteLine("###");
        D d2 = new D(4);
    }
}

```

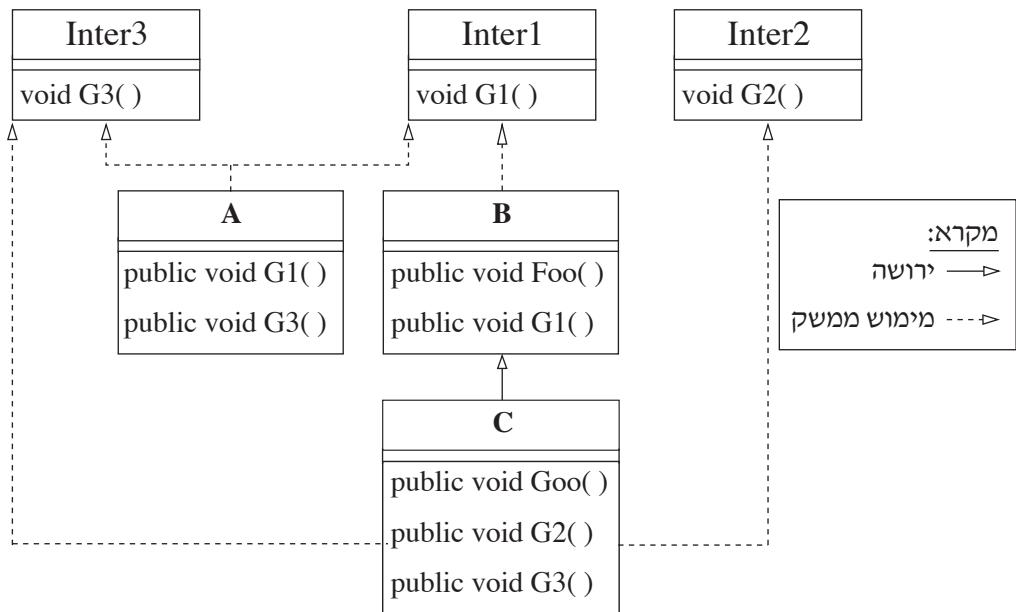
ב. (אין קשר לסעיף א.).

לפניך המחלקות AA, BB, Run. Run, ופעולה ראשית במחלקה Run. עקוב אחר הפעולה הראשית, ורשות את הפולט המתקבל. במקביל יש לכתוב את ערכי המשתנים, את ערכי תכונות המחלקה (תכונות סטטיות), ובמעבר כל עצם יש לכתוב את ערכי התכונות שלו.

```
public class AA
{
    private int num1 = 0;
    private int num2 = 0;
    private static int count = 0;
    public AA(int num)
    {
        num1 = num;
        num2 = num;
        count++;
        Console.WriteLine("AA ctor1");
    }
    public AA(int num1, int num2)
    {
        this.num1 = num1;
        this.num2 = num2;
        count++;
        Console.WriteLine("AA ctor2");
    }
    public virtual int Sum()
    {
        return num1 + num2;
    }
    public static int GetCount()
    {
        return count;
    }
}
```

```
public class BB : AA
{
    private int num3 = 0;
    public BB(int num)
        :base (num)
    {
        num3 = num;
        Console.WriteLine("BB ctor1");
    }
    public BB (int num1, int num2, int num3)
        :base(num1, num2)
    {
        this.num3 = num3;
        Console.WriteLine("BB ctor2");
    }
    public override int Sum()
    {
        return base.Sum() + num3;
    }
    public void SetNum3(int num)
    {
        num3 = num;
    }
}
```

```
public class Run
{
    public static void Main()
    {
        AA f1 = new AA(10);
        Console.WriteLine("sum1 = " + f1.Sum());
        Console.WriteLine("count = " + AA.GetCount());
        AA f2 = new AA(10, 20);
        Console.WriteLine("count = " + AA.GetCount());
        BB s1 = new BB(1);
        AA f3 = new BB(2);
        Console.WriteLine("count = " + AA.GetCount());
        f2 = s1;
        Console.WriteLine("sum = " + f2.Sum());
        s1.SetNum3(2);
        Console.WriteLine("sum = " + s1.Sum());
        Console.WriteLine("sum = " + f2.Sum());
    }
}
```



.א. השלים את שלוש השורות החסרות כך שקטוע התכנית שלפניך יעבד בלי שגיאות הידור (קומPILEציה) ו/או זמן ריצה.
העתק למחברתך את השורות שהשלמות.

```

_____ x = new A();
_____ y = new B();
_____ z = new C();

x = y;
y = new C();
y.Foo();
z.G3();
((C) z).Goo();
z = new A();
  
```

.ב. כתוב ב- C# פעליה שתתקבל מערך חד-ממדי מטיפוס **.Inter1**.
הפעולה תעבור על איברי המערך, ותקרו לפעליה () G3(). אם היא קיימת בעבר העצם.
אם הפעולה () G3() אינה קיימת – הפעולה שכתבה תקרא לפעליה () G1().

. type3 , type2 , type1 .²⁴ חברת מסוימת מייצרת שלושה סוגי טלפונים סלולריים:

טלפון מסוג type1 אפשר לבצע את הפעולות האלה:

—	חיוֹג —	Dial
—	קבלה שיחה —	RecCall
—	שליחת הודעה —	SendMsg
—	קבלת הודעה —	RecMsg

טלפון מסוג type2 אפשר לבצע את הפעולות האלה:

—	חיוֹג —	Dial
—	קבלה שיחה —	RecCall
—	שליחת הודעה —	SendMsg
—	קבלת הודעה —	RecMsg
—	צילום תמונה —	TakePic
—	תצוגת תמונה —	ShowPic

טלפון מסוג type3 אפשר לבצע את הפעולות האלה:

—	חיוֹג —	Dial
—	קבלה שיחה —	RecCall
—	צילום תמונה —	TakePic
—	תצוגת תמונה —	ShowPic
—	התקנת אפליקציות —	InstallApp

בכל אחד מסוגי הטלפונים:

נשמר מספר דקוטה השיחה שנעשו מן הטלפון.

אפשר לAPS את מספר דקוטה השיחה — . Reset

אפשר להחזיר חשבון טלפון עדכני — Bill , המוחשב כר: מספר דקוטה השיחה × 1 ש"ח.

טלפונים שיש בהם אפשרות שלוחה הודעות, החזרת חשבון הטלפון העדכני היא:

מספר דקוטה השיחה × 1 ש"ח + מספר ההודעות שנשלחו × 0.5 ש"ח.

טלפונים אלה פועלות האיפוס מאפסת את מספר דקוטה השיחה וכן את מספר ההודעות שנשלחו.

(שים לב: המשך השאלה בעמוד הבא.)

.א.

בעבור הטלפונים הסולריים המתוארים לעיל, סרטט את היררכיות המחלקות והמשקדים, באופן המתאים ביותר לעקרונות של תכונות מונחה עצמים (הכמה inheritance — הורשה — encapsulation — סמן בסרטוט את הקשרים בין המחלקות, ואת הקשרים בין המחלקות ובין המשקדים.).

.ב.

כל אחת מן המחלקות שבסרטוט כתוב ב- C# את כותרת המחלקה ואת התכונות שלה, וכותרות לכל הפעולות.

.ג.

כל אחד מן המשקדים שבסרטוט כתוב ב- C# את הכותרת שלו, וכותרות לכל הפעולות. אין צורך לכתוב את הפעולות Get ו- Set ופעולות בונות. ממש ב- C# את הפעולה המחזירה את החשבון העדכני בכל אחת מן המחלקות שהיא מופיעה בהן.

בצלחה!

זכות היוצרים שורוה למדינת ישראל
אין להעתיק או לפרסם אלא ברשות משרד החינוך