

סוג הבדיקה: א. בגרות לבתי ספר על-יסודיים
ב. בגרות לנבחנים חיצוניים
מועד הבדיקה: קיץ תשע"א, 2011
מספר השאלה: 603, 899205

מִדְעֵי הַמְחַשֵּׁב ב'

2 ייחדות לימוד (השלמה ל-5 יח"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.
פרק שני – בפרק זה שאלות בארבעה מסלולים שונים – (2x2) – 50 נקודות
עננה על שאלות בק במסלול שלמדת,
לפי ההוראות בקבוצת השאלות במסלול זה.
סה"כ – 100 נקודות
- ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ מחשב הנitin לתכנות.

הוראות מיוחדות:

- (1) את בל התכניות שאתה נדרש לכתוב בשפת מחשב בפרק הראשון
כתבו בשפה אחת בלבד – C# או Java.
- (2) רשום על הכירכה החיצונית של המחברת באיזו שפה אתה כותב – C# או Java.
- (3) רשום על הכירכה החיצונית של המחברת את שם המסלול שלמדת,
אחד מארבעת המסלולים: מערכות מחשב ואסטטיקה, מבוא לחקור ביצועים,
מודלים חישוביים, תכנות מונחה עצמים.

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

כתוב במחברת הבדיקה בלבד, בעמודים נפרדים, כל מה שברצונך לכתוב בטויטה (ראשי פרקים, חישובים וכדומה).
רשום "טיוטה" בראש כל עמוד טיווטה. רישום טווטות כלשהן על דפים שמחוץ למחברת הבדיקה עלול לגרום לפסילת הבדיקה!

הנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות לנבחנים כאחד.

בהתלה!

/המשך 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 מימוש אתה יכול להשתמש בפעולות של המחלקות `BinTreeNode<T>`, `Stack<T>`, `Queue<T>`, `Node<T>`, `ArrayList<T>` ו`String`.

בלי למשם אותן. אם אתה משתמש בפעולות נוספות عليك למשם אותן.

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

שים לב: לשאלת זו שני נוסחים: אחד ב- Java (עמודים 3-2), ואחד ב- C# (עמודים 4-5). עבוד על פי השפה שלמדת.

לפתרונות ב- Java

- א.** לפני כמה מחלקות. עקב אחר הפעולה `main` שבמחלקה `Program` ורשום את הפלט.
במקרה יש לכלול את ערכי המשתנים, ובabbo כל עצם – את ערכי התכונות שלו.

```

public class One
{
    private double n;

    public One(double n) { this.n = n; }

    public double getN() { return this.n; }

    public void f() { System.out.println("f of one "); }

    public void g() { System.out.println("g of one " + this.getN()); }
}

public class Two
{
    private Queue<One> q;

    public Two(int n, int m)
    {
        this.q = new Queue<One>();
        for (int i = n; i < m; i++)
            this.q.insert(new One(Math.pow(2, i)));
    }

    public void f()
    {
        System.out.print("f of two ");
        if (!this.q.isEmpty())
            System.out.println(this.q.remove().getN());
    }
}

```

```

public class Together
{
    private double x;
    private int from , to;
    private One first;
    private Two second;

    public Together(double x , int from , int to)
    {
        this.x = x;
        this.from = from;
        this.to = to;
        this.first = new One(x);
        this.second = new Two(from , to);
    }

    public void methodA()
    {
        System.out.println("-- MethodA() -- ");
        this.first.f();
    }

    public void methodB()
    {
        System.out.println("-- MethodB() -- ");
        this.first.g();
        this.second.f();
    }
}

public class Program
{
    public static void main(String[] args)
    {
        Together tg = new Together(5.0 , 2 , 6);
        tg.methodA() ;
        tg.methodB() ;
    }
}

```

ב. בהנחה כי הפעולה הבונה במחלקה Together מקבלת מספרים גדולים מ- 0 בלבד – מה מבצעת הפעולה f() במחלקה Two ? המשך בעמוד 4/

- א. לפניך כמה מחלקות. עקב אחר הפעולה Main שבמחלקה Program ורשות את הפלט. במקבב יש לכלול את ערכיו המשתנים, ובמעבר כל עצם – את ערכי התוכנות שלו.

```

public class One
{
    private double n;
    public One(double n) { this.n = n; }
    public double GetN() { return this.n; }
    public void F() { Console.WriteLine("F of one "); }
    public void G() { Console.WriteLine("G of one " + this.GetN()); }
}

public class Two
{
    private Queue<One> q;
    public Two(int n, int m)
    {
        this.q = new Queue<One>();
        for (int i = n; i < m; i++)
            this.q.Insert(new One(Math.Pow(2, i)));
    }

    public void F()
    {
        Console.Write("F of two ");
        if (!this.q.IsEmpty())
            Console.WriteLine(this.q.Remove().GetN());
    }
}

```

```

public class Together
{
    private double x;
    private int from , to;
    private One first;
    private Two second;

    public Together(double x , int from , int to)
    {
        this.x = x;
        this.from = from;
        this.to = to;
        this.first = new One(x);
        this.second = new Two(from , to);
    }

    public void MethodA()
    {
        Console.WriteLine("-- MethodA() -- ");
        this.first.F();
    }

    public void MethodB()
    {
        Console.WriteLine("-- MethodB() -- ");
        this.first.G();
        this.second.F();
    }
}

public class Program
{
    public static void Main(string[] args)
    {
        Together tg = new Together(5.0 , 2 , 6);
        tg.MethodA();
        tg.MethodB();
    }
}

```

ב. בהנחה כי הפעולה הבונה במחלקה Together מקבלת מספרים גדולים מ- 0 בלבד – מה מבצעת הפעולה () F במחלקה Two ? /המשך בעמוד 6/

שים לב: לשאלה זו שני נוסחים: אחד ב- Java (עמודים 6-7), ואחד ב- C# (עמודים 8-9). עבود על פי השפה שלמדת.

.2

לפתרונות ב- Java

לפניך הפעולות sod ו- what המקבלות מערך a שאיבריו מティפוס שלם, ממויין בסדר עולה, ומספר שלם k . לשתי הפעולות אותה טענת יציאה.

```
public static boolean sod(int[] a , int k)
{
    for (int i = 0; i < a.length-1; i++)
    {
        int j = i+1;
        while (j < a.length)
        {
            if (a[i] + a[j] == k)
                return true;
            j++;
        }
    }
    return false;
}
```

```
public static boolean what(int[] a , int k)
{
    int left = 0 , right = a.length-1;
    while (left < right)
    {
        if (a[left] + a[right] == k)
            return true;
        if (a[left] + a[right] < k)
            left++;
        else
            right--;
    }
    return false;
}
```

a	2	4	7	12	18
---	---	---	---	----	----

א. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה `sod` בעבר המערך הנתון `a` והמספר $k = 11$. רשום את הערך המוחזר.

בטבלת המעקב יש לכלול עמודות בעבר: `[j]`, `a[i]`, `a[i]`, `i`, ועמודה נוספת שבה יצוין אם התנאי שבפקודת `if` מתקיים או אינו מתקיים.

ב. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה `sod` בעבר המערך הנתון `a` והמספר $k = 10$. רשום את הערך המוחזר.

בטבלת המעקב יש לכלול את העמודות שפורטו בסעיף א.

ג. מהי טענת היציאה של הפעולה `sod` ?

ד. מהי סיבוכיות זמן הריצה של הפעולה `sod` ? נמק את תשובתך.

ה. עקוב בעזרת טבלת מעקב אחר ביצוע הפעולה `what` בעבר המערך הנתון `a` והמספר $k = 11$. רשום את הערך המוחזר.

בטבלת המעקב יש לכלול עמודות בעבר: `, left, right, a[left], a[right]`

ושתי עמודות נוספות לכל אחת מפקודות `if`. בכל עמודה יצוין אם התנאי בפקודת `if` מתקיים או אינו מתקיים.

ו. מהי סיבוכיות זמן הריצה של הפעולה `what` ? נמק את תשובתך.

ז. מי מבין שתי הפעולות — `sod` או `what` — יעילה יותר? נמק את תשובתך.

ח. טענת הכניסה של הפעולות `sod` ו- `what` שונות בכך שאפשר להעביר אליהן מערך `a` לא ממויין.

(1) האם טענת היציאה של הפעולה `sod` תשתנה? נמק את תשובתך.

(2) האם טענת היציאה של הפעולה `what` תשתנה? נמק את תשובתך.

C# ב-

לפניך הפעולות Sod ו- What המקבלות מערך a שאיבריו מטיבוס שלם, ממון בסדר עולה, ומספר שלם k. לשתי הפעולות אותה טענת יצאה.

```
public static bool Sod(int[] a , int k)
{
    for (int i = 0; i < a.Length-1; i++)
    {
        int j = i+1;
        while (j < a.Length)
        {
            if (a[i] + a[j] == k)
                return true;
            j++;
        }
    }
    return false;
}

public static bool What(int[] a , int k)
{
    int left = 0 , right = a.Length-1;
    while (left < right)
    {
        if (a[left] + a[right] == k)
            return true;
        if (a[left] + a[right] < k)
            left++;
        else
            right--;
    }
    return false;
}
```

a	2	4	7	12	18
---	---	---	---	----	----

a. עוקוב בעזרה טבלת מעקב אחר ביצוע הפעולה Sod בעבור המערך הנתון a והמספר $k = 11$. רשום את הערך המוחזר.

בטבלת המערך יש לכלול עמודות בעבורו: $[j]_a$, $a[i]$, i , j , ועומודה נוספת נספפת שבה יצוין אם התנאי שבקוד if מתקיים או אינו מתקיים.

b. עוקוב בעזרה טבלת מעקב אחר ביצוע הפעולה Sod בעבור המערך הנתון a והמספר $k = 10$. רשום את הערך המוחזר.

בטבלת המערך יש לכלול את העמודות שפורטו בסעיף a.

ג. מהי טענת היציאה של הפעולה Sod ?

ד. מהי סיבוכיות זמן הריצה של הפעולה Sod ? נמק את תשובתך.

ה. עוקוב בעזרה טבלת מעקב אחר ביצוע הפעולה What בעבור המערך הנתון a והמספר $k = 11$. רשום את הערך המוחזר.

בטבלת המערך יש לכלול עמודות בעבורו: $, left, right, a[left], a[right]$, ושתי עמודות נוספות לכל אחת מפקודות if . בכל עמודה יצוין אם התנאי בפקודת if מתקיים או אינו מתקיים.

ו. מהי סיבוכיות זמן הריצה של הפעולה What ? נמק את תשובתך.

ז. מי מבין שתי הפעולות – Sod או What – ייעילה יותר? נמק את תשובתך.

ח. טענת הכניסה של הפעולות Sod ו- What שונתה כך שאפשר להעביר אליהן מערך a לא ממויין.

(1) האם טענת היציאה של הפעולה Sod תשתנה? נמק את תשובתך.

(2) האם טענת היציאה של הפעולה What תשתנה? נמק את תשובתך.

שים לב: לשאלת זו שני נוסחים: אחד ב- Java (עמודים 10-11) , ואחד ב- C# (עמודים 12-13). עבוד על פי השפה שלמדת.

.3

לפותרים ב- Java

לפניך ממוקש של המחלקה **"טור-ביטול"** (**UndoQueue**) :

המחלקה מגדרה טיפוס אוסף עם פרוטוקול FIFO להכנסת והוצאת ערכיהם של למים ונגדלים מ-.

שם הפעולה	תיאור
UndoQueue()	הפעולה בונה טור-ביטול ריק.
boolean isEmpty()	הפעולה מחזירה true אם טור-ביטול הנוכחי ריק, אחרת — הפעולה מחזירה false.
void insert(int x)	הפעולה מכניסה את הערך x בסוף הטור-ביטול הנוכחי.
int remove()	הפעולה מוציאת את הערך שבראש טור-ביטול הנוכחי, ומחזירה אותו. הנחה: טור-ביטול הנוכחי אינו ריק.
int head()	הפעולה מחזירה את הערך של האיבר שבראש טור-ביטול בלי להוציאו. הנחה: טור-ביטול הנוכחי אינו ריק.
void undo()	הפעולה מבטלת את פעולות ה- remove או את פעולות ה- insert האחרונות שבוצעה, ומשחררת את טור-ביטול כפי שהוא לפני ביצוע הפעולה. אם אין פעולות remove או insert שבוצעו, הפעולה אינה עושה דבר. שים-לב: הפעלה כ פעמים של הפעולה undo, תחזיר את טור-ביטול במצב שבו הוא היה לפני ביצוע כ פעולות remove או insert לאחר שטרם בוטלו.

דוגמאות:

הפעולה	מצב הטור-ביטול לאחר ביצוע הפעולה
UndoQueue uq = new UndoQueue();	הטור-ביטול ריק
uq.insert(2);	→2→
uq.insert(3);	→3 , 2→
uq.remove();	→3→
uq.insert(4);	→4 , 3→
uq.remove();	→4→
uq.remove();	הטור-ביטול ריק
uq.undo();	→4→
uq.undo();	→4 , 3→
uq.insert(1);	→1 , 4 , 3→
uq.undo();	→4 , 3→
uq.undo();	→3→
uq.undo();	→3 , 2→
uq.undo();	→2→
uq.undo();	הטור-ביטול ריק
uq.undo();	הטור-ביטול ריק

א. לפניך כתע קוד.

```
UndoQueue q = new UndoQueue();
q.insert(1);
q.insert(2);
q.insert(3);
q.remove();
q.insert(4);
q.undo();
q.undo();
```

הראה את מצב הטור-ביטול לאחר הביצוע של כל הוראה בקטע הקוד.

ב. כתוב ב- Java את כוורתת המחלקה **UndoQueue**, ואת התכונות שלה. רשום תיעוד כל תוכנה.ג. ממש ב- Java את הפעולות **insert**, **remove** ו- **undo** המוצגות ממש מחלון המחלקה **UndoQueue**.ה. אתה יכול להשתמש בפעולות המשק האחריות של **UndoQueue** בלי למש אתם. אם אתה משתמש בפעולות נוספות, عليك למש אותן. /המשך בעמוד 12/

פתרונות ב- C#

לפניך ממשק של המחלקה "**תור-ביטול**" (**UndoQueue**) :

המחלקה מדירה טיפוס אוסף עם פרוטוקול FIFO להכנסה והוצאה של ערכים שלמים וגדולים מ- 0.

שם הפעולה	תיאור
UndoQueue()	הפעולה בונה תור-ביטול ריק.
bool IsEmpty()	הפעולה מחזירה true אם התור-ביטול הנוכחי ריק, אחרת — הפעולה מחזירה false.
void Insert(int x)	הפעולה מכניסה את הערך x לסוף התור-ביטול הנוכחי.
int Remove()	הפעולה מוציאה את הערך שבראש התור-ביטול הנוכחי, ומחזירה אותו. הנחה: התור-ביטול הנוכחי אינו ריק.
int Head()	הפעולהמחזירה את הערך של האיבר שבראש התור-ביטול בלי להוציאו. הנחה: התור-ביטול הנוכחי אינו ריק.
void Undo()	הפעולה מבטלת את פעולה ה- Remove או את פעולה ה- Insert האחרונה שבוצעה, ומשוחררת את התור-ביטול כפי שהיא לפני ביצוע הפעולה. אם אין פעולות Remove או Insert שבוצעו, הפעולה אינה עושה דבר. <u>שים-לב:</u> הפעלה זו פעמים של הפעולה Undo, תזריר את התור-ביטול למצב שבו הוא היה לפני ביצוע ח-פעולות Remove או Insert.

דוגמאות:

הפעולה	לאחר ביצוע הפעולה
הטור-ביטול ריק	UndoQueue uq = new UndoQueue();
→2→	uq.Insert(2);
→3 , 2→	uq.Insert(3);
→3→	uq.Remove();
→4 , 3→	uq.Insert(4);
→4→	uq.Remove();
הטור-ביטול ריק	uq.Remove();
→4→	uq.Undo();
→4 , 3→	uq.Undo();
→1 , 4 , 3→	uq.Insert(1);
→4 , 3→	uq.Undo();
→3→	uq.Undo();
→3 , 2→	uq.Undo();
→2→	uq.Undo();
הטור-ביטול ריק	uq.Undo();
הטור-ביטול ריק	uq.Undo();

א. לפניך קטע קוד.

```
UndoQueue q = new UndoQueue();
q.Insert(1);
q.Insert(2);
q.Insert(3);
q.Remove();
q.Insert(4);
q.Undo();
q.Undo();
```

הראה את מצב **הטור-ביטול** לאחר הביצוע של כל הוראה בקטע הקוד.

כתב ב- C# את כוורתת המחלקה **UndoQueue**, ואת התכונות שלה. רשום תיעוד לכל תכונה.

ממש ב- C# את הפעולות **Insert**, **Remove** ו- **Undo** המוצגות במסמך המחלקה **UndoQueue**.

אתה יכול להשתמש בפעולות המשק האחוריות של **UndoQueue** בלי למש אותן. אם אתה משתמש בפעולות נוספות, عليك למש אותן.
המשך בעמוד 14 /

.4. א. לפניך כותרת של פעולה: ב- Java

```
public static void leaves(BinTreeNode<Integer> t , Stack<Integer> s)
```

ב- C#:

```
public static void Leaves(BinTreeNode<int> t , Stack<int> s)
```

הפעולה מקבלת עץ BinTreeNode t של מספרים שלמים, ומחסנית ריקה s של מספרים שלמים.

הפעולה מכניסה למחסנית את ערכי כל העלים של העץ t, על פי סדר סריקה מימין לשמאל.

משה ב- Java או ב- C# את הפעולה.

כתב ב- Java או ב- C# פעולה בוליאנית שתקבל 2 עצים ביןרים לא ריקים

של מספרים שלמים, ותחזיר true אם מתקיימים שני התנאים האלה:

— יש להם אותו מספר עלים

— על פי סדר הסריקה מימין לשמאל, ערכי העלים שווים
אחרת — הפעולה תחזיר false.

עליך להשתמש בפעולה שסימנת בסעיף א.

פרק שני (50 נקודות)

בפרק זה שאלות באربعة מסЛОלים:

מערכות מחשב וasmblar, עמ' 15-20

מבוא לחקיר ביצועים, עמ' 21-31

מודלים חישוביים, עמ' 32-34

תכנות מונחה עצמים ב- Java, עמ' 36-45 ; תכנות מונחה עצמים ב- C#, עמ' 46-55

ענה רק על שאלות במסלול שלמדת.**מערכות מחשב וasmblar**

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

.5. במחסנית מאוחסנים 10 מספרים שלמים עם סימן, בגודל מילה.

. בנוסף, במקטע הנתונים מוגדרים משתנה K ושני מערכיים, ARR1 ו- ARR2.

. כל תאי המערך ARR1 מאותחלים ל- 0, והמשתנה K מאותחל ל- 0.

ARR1 DW 10 DUP (0)

ARR2 DW 10 DUP (?)

K DW 0

יש להעתיק לערך ARR1 את כל המספרים שמאוחסנים במחסנית באופן הזה:
כל המספרים השליליים יוחסנו בתחלת המערך לפי סדר הוצאתם מהמחסנית. אחרי
המספרים השליליים יוחסנו כל האפסים, ולאחר מכן יוחסנו כל המספרים החיוביים
לפי סדר הוצאתם מהמחסנית.

לדוגמא, בעבור המכחסנית שלפניך המכילה 6 איברים:

-5
0
10
-2
0
8

יראה המערך ARR1 כך:

0	1	2	3	4	5
-5	-2	0	0	10	8

לפניך קטע תכנית באסמלול, המבצע את הנדרש.
 מקטע התכנית הושםתו שורות במקומות המסומנים במספרים i-iv.
 רשום במחברתך את מספרי השורות המסומנות, וכתוב ליד כל אחד מהמספרים את
 שורת הפקודה החסורה, כדי שקטע התכנית יבצע את הנדרש.

	MOV	SI , 0
	MOV	DI , 0
	MOV	CX , 10
A1:	POP	AX
i	_____	_____
	JLE	A2
ii	_____	_____
	ADD	DI , 2
	JMP	A4
A2:	CMP	AX , 0
	JNE	A3
	INC	K
	JMP	A4
iii	A3:	_____
	ADD	SI , 2
A4:	LOOP	A1
	ADD	SI , K
	ADD	SI , K
	MOV	CX , DI
	SHR	CX , 1
	XOR	BX , BX
A5:	MOV	DX , ARR2[BX]
	MOV	ARR1[SI] , DX
iv	_____	_____
	ADD	BX , 2
	LOOP	A5

- .6. מערך A נקרא **מוגול** במערך B , אם האורך של מערך A אינו עולה על האורך של מערך B . וכל הערכים של איברי מערך A נמצאים במערך B ברכז ובאותו סדר כמו במערך A . בכל מערך מאוחסנים מספרים שלמים עם סימן, השוניים זה מזו. לדוגמה: בעבור המערכים A ו- B שלפניך, המערך A **מוגול** במערך B .
מערך A :

0	1	2	3	4	5
3	-5	0	8	5	11

מערך B :

0	1	2	3	4	5	6	7	8	9
15	-7	4	3	-5	0	8	5	11	2

איברי המערך B המסומנים באפור הם הערכים של איברי המערך A .

- א. בקטע הנתונים הוגדרו הנתונים כך:

ARR_B DB 100 DUP (?)

ARR_A DB 10 DUP (?)

V DB ?

P DB ?

כתוב באסמבלי שגורה (פרוצדורה) בשם TEST , שתקבל ערך שמאוחסן במשתנה V .

השגרה תבודוק אם הערך שמאוחסן במשתנה V נמצא במערך B . ARR_B

אם כן – השגרה תאחסן במשתנה P את האינדקס של האיבר המתאים,

אחרת – השגרה תאחסן במשתנה P את הערך 1 .

כתוב באסמבלי קטע תכנית שיבדק אם המערך ARR_B מוגול במערך ARR_A .

אם כן – קטע התכנית יאחסן 1 באוגר BL , אחרת – הוא יאחסן 0 באוגר BL .

עליך להשתמש בשגרה TEST שכותבת בסעיף A .

הנח שהאורך של המערך ARR_A אינו עולה על האורך של מערך ARR_B .

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

א. במקטע הנתונים הוגדר מערך : ARR

ARR DB 6 DUP (?)

יש להפוך את סדר האיברים במערך.

לדוגמה, נתון המערך :

0	1	2	3	4	5
3	2	10	0	6	7

לאחר הפיכת סדר האיברים, ייראה המערך כך :

0	1	2	3	4	5
7	6	0	10	2	3

לפניך 2 קטעים באסמבולר, i-ii, שטורתם להפוך את סדר האיברים במערך . ARR

עקוב בעזרת טבלת מעקב אחר הביצוע של כל אחד מהקטעים i-ii שלפניך, וקבע אם הוא מבצע את הנדרש או אינו מבצע את הנדרש.

i MOV SI , 3
 MOV DI , 2
 A1: MOV AL , ARR[DI]
 MOV AH , ARR[SI]
 ADD ARR[DI] , AH
 SUB ARR[DI] , AL
 MOV ARR[SI] , AL
 INC SI
 DEC DI
 JNZ A1

ii MOV CX , 3
 MOV SI , 3
 MOV DI , 2
 A1: MOV AL , ARR[DI]
 MOV AH , ARR[SI]
 MOV ARR[DI] , AH
 MOV ARR[SI] , AL
 INC SI
 DEC DI
 LOOP A1

ב. (אין קשר לסעיף א.)

במקטע הנתונים הוגדרו הנתונים כך:

T DB 55 , 90 , 110 , 1

שים לב: המספרים הם עשרוניים.

לפניך קטע תכנית באסטמבלר, שמטרתו לחשב את סכום האיברים במערך T.

MOV	CX , 4
XOR	AX , AX
LEA	BX , T
AGAIN:	ADD AL , [BX]
	INC BX
	LOOP AGAIN

קטע התכנית אינו מבצע את הנדרש.

(1) עקוב בעורף טבלת מעקב אחר ביצוע קטע התכנית, וכתוב מה יהיה התוכן של AL, ומה יהיה התוכן של דגל הנשא ושל דגל הגלישה בסיום הביצוע של קטע התכנית.

(2) שנה את קטע התכנית כך שייחסב את סכום האיברים במערך הנתון T. העתק למחברתך את קטע התכנית לאחר השינוי.

.8. לפניך קטע תוכנית באסמבלי:

i	PUSHF	
ii	PUSH	BP
iii	MOV	BP , SP
iv	MOV	AX , [BP+4]
v	MOV	BP , [BP]
vi	ADD	SP , 4
vii	POPF	

נתון שלפני ביצוע קטע התוכנית:

- התוכן של אוגר הדגלים הוא 3202H
- התוכן של האוגר SP הוא 96H
- התוכן של האוגר BP הוא 5678H

לפניך חלק מממחסנית המכילה מספרים הקסדצימליים.
לפני ביצוע קטע התוכנית, האוגר SP מצביע על התא המסומן בחץ, שכותובתו היחסית H 96.

.....	*	*	*	*	*	*	*	02H	32H	*	*
.....	90H	91H	92H	93H	94H	95H	96H	97H	98H	99H



תא המסומן ב- * הוא תא שאין משמעותו לתוכנו.

.א. עקב בעורת טבלת מעקב אחר ביצוע קטע התוכנית.

. AX , BP , SP , BP . בטבלת המעקב יש לכלול עמודות בעבור האוגרים:

.ב. על פי טבלת המעקב, ענה על התת-סעיפים (1)-(4).

(1) איזה מספר יירשם במחסנית הנתונה לאחר ביצוע הוראה ?

כתב את הכתובת היחסית של התאים שבהם יירשם המספר.

(2) מה יהיה התוכן של האוגר BP לאחר ביצוע הוראה ??? ?

(3) מה יהיה התוכן של האוגר AX לאחר ביצוע הוראה ?? ?

(4) מה יהיה התוכן של האוגר BP לאחר ביצוע הוראה ?? ?

מבוא למחקר ביצועים

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

9. בשאלת זו שלושה סעיפים א-ג, שאיןם תלויים זה זהה. ענה על בל הטעיפים.

א. יהיו $G = (V, E)$ גראף מכובן, V היא קבוצת הקדקודים ו- E היא קבוצת הקשתות.

G מיוצג על ידי מטריצת הסמיוכיות שלפני:

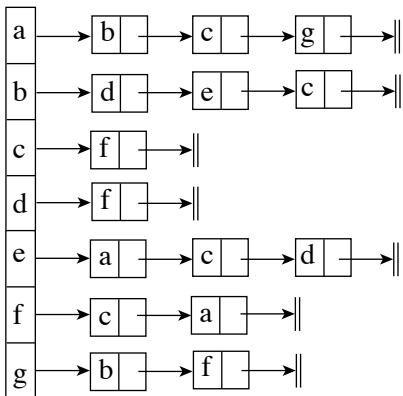
	a	b	c	d	e
a	0	1	1	0	0
b	0	0	1	0	0
c	0	1	0	1	0
d	0	0	0	0	1
e	1	0	1	0	0

(1) סרטט את הגרף G המיוצג על ידי המטריצה.

(2) מצא את רכיב / רכיבי הקשרות החזקה (רקי"ח / רקי"חים) שבגרף G .
בעבור כל רקי"ח שמצאת, רשום את קבוצת הקדקודים שלו.

(3) הסר קשת אחת מהגרף G שشرطתה, כך שמספר הרקי"חים יגדל ב- 1.
ציין את הקשת שהסרת, ולכל רקי"ח רשום את קבוצת הקדקודים שלו.

ב. יהיו $G = (V, E)$ גראף מכובן, V היא קבוצת הקדקודים ו- E היא קבוצת הקשתות.
 G מיוצג על ידי רשימת הסמיוכיות הזו:



(1) הפעל אלגוריתם סריקה לעומק (DFS) על הגרף הנוכחי החל מקדוקוד a.
סרטט במחברתך את העץ הפורש DFS שמתקיים.

(2) הפעל אלגוריתם סריקה לרוחב (BFS) על הגרף הנוכחי החל מקדוקוד f.
סרטט במחברתך את העץ הפורש BFS שמתקיים. (המשך בעמוד 22/)

ג. גרף מכון. ייְהִי G גרף מכון.

נפעיל את האלגוריתם סריקה לעומק (DFS) על הגרף המכון G .

ונדריך את הקשיות של הגרף G באמצעות העץ הפורש / הייר הפורש שנוצר.

קשת עץ היא קשת של הגרף G , הנמצאת גם בעץ הפורש / הייר הפורש של G .

קשת אחורייה היא קשת (v, u) בגרף G , שאינה נמצאת בעץ הפורש / הייר הפורש.

זו קשת מקדקוד ולקדקוד v בגרף G , כאשר v הוא אב קדמון של u

בעץ הפורש / הייר הפורש.

קשת קדמית היא קשת (v, u) בגרף G , שאינה נמצאת בעץ הפורש / הייר הפורש.

זו קשת מקדקוד ולקדקוד v בגרף G , כאשר v יצא של u

בעץ הפורש / הייר הפורש.

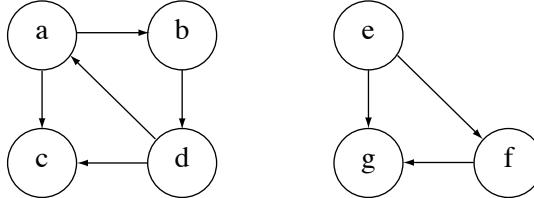
קשת חוצה היא קשת (v, u) בגרף G , שאינה נמצאת בעץ הפורש / הייר הפורש,

ואינה **קשת אחורייה או קשת קוזמית**. זו קשת מקדקוד ולקדקוד v בגרף G ,

בתנאי שאף אחד משני קדוקדים אלה איןן יצא של הקדקוד לאחר

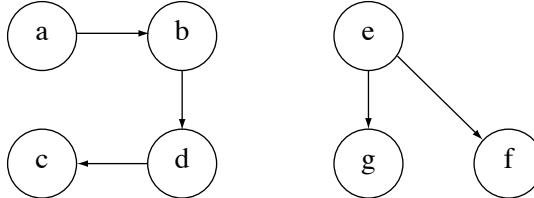
בעץ הפורש / הייר הפורש.

(1) לפניך גרף מכון G , המכיל יותר מرك"ח (רכיב קשיירות חזקה) אחד.



לאחר הפעלת אלגוריתם הסריקה לעומק (DFS) על הגרף המכון G הנתון,

התקבל הייר הפורש שלפניך:



העתק למחברתך את הגרף G . הסטמך על הייר הפורש הנתון וכתוב ליד

כל אחת מהקשיות של הגרף G אם היא:

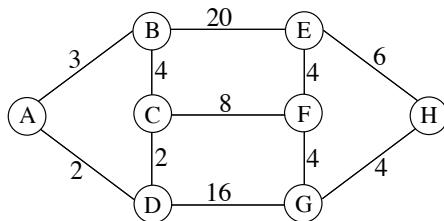
- **קשת עץ**
- **קשת אחורייה**
- **קשת קדמית**
- **קשת חוצה**

- (2) לפניך טענה שחרר בה ביטוי במקומות המסומנים בקוו.
- אם לאחר הפעלת האלגוריתם סריקה לעומק (DFS) על גראן מסוון G נמצאת ב- G _____, אזי יש ב- G מעגל / מעגלים.
- לפניך ארבעה ביטויים אפשריים להשלמת הטענה:
- **קשת עז**
 - **קשת אחורייה**
 - **קשת קדמית**
 - **קשת חוצה**

העתק למחברתך את האפשרות המתאימה להשלמת הטענה, וنمוק את קביעתך.

/המשך בעמוד 24/

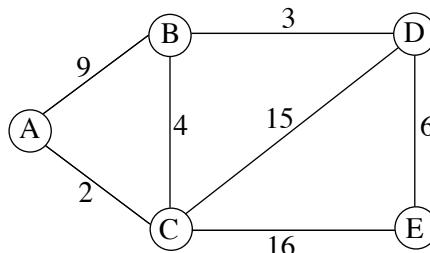
10. בשאלת זו שני סעיפים א-ב שאינם תלויים זה זהה. ענה על שני הסעיפים.
א. לפניך רשת (V, E) , $G = (V, E) = (V, E)$ היא קבוצת הקדקודים ו- E היא קבוצת הקשרות:



השתמש באלגוריתם של Prim למציאת עץ פורש מינימלי וسرטט את העץ הפורש המינימלי המתקיים עבור הרשת הנתונה, החל מקדוקוד C.

ב. (אין קשר לסעיף א).

נתונה הרשת:



- (1) השתמש באלגוריתם של Kruskal למציאת עץ פורש מינימלי וסרטט את העץ הפורש המינימלי של הרשת הנתונה.
(2) תלמיד הציע למורה אלגוריתם למציאת עץ פורש מינימלי לרשת $(V, E) = G$ כאשר V היא קבוצת הקדקודים ו- E היא קבוצת הקשרות של הרשת.

- צעד 1: צור גראף G_1 , שקבוצת הקדקודים שלו היא קבוצת הקדקודים של הרשת G וקבוצת הקשרות שלו, E_T , היא קבוצה ריקה.
צעד 2: מין את קשיות הרשת G בסדר עולה, על פי המשקלות המוחושים להן. תתקבל קבוצת קשיות ממוגנות שנסמנה E^* .

צעד 3: לכל קשת (v, u) ב- E^* , החל מהקשת בעלת המשקל

הקטן ביותר, בצע:

אם הוספת הקשת (v, u) **איינה** יוצרת מעגל ב- G_1 – **ווסף**

את הקשת (v, u) ל- E_T .

אחרת – החזר את הגראף G_1 שהינו עז פורש מינימלי.

צעד 4: החזר את G_1 שהינו עז פורש מינימלי.

הפעל את האלגוריתם של התלמיד על הרשות הנתונה.

ענה על הסעיפים (i)-(iv).

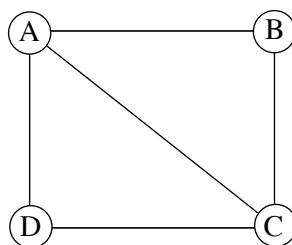
(i) סרטט במחברתך את G_1 שנוצר אחרי ביצוע צעד 1.

(ii) רשם במחברתך את כל הקשתות ב- E^* אחרי ביצוע צעד 2.

(iii) סרטט במחברתך באחד מהגרפים G_1 שהתקבלו במהלך צעד 3.

(iv) סרטט במחברתך את העז הפורש שהאלגוריתם החיזיר.

(3) המורה השיבה לתלמיד שאלגוריתם שהוא הציע שגוי, כי הפלט שלו אינו בהכרח עז פורש מינימלי. למשל, בעבר הרשות הזה:



אפשר ליחס משקלות לקשתות הרשות כך שאם נשתמש באלגוריתם של התלמיד לא יוחזר העז הפורש המינימלי.

העתק למחברתך את הרשות ורשם משקלות לצדן של הקשתות, כך שהעז שיוחזר על ידי האלגוריתם של התלמיד לא יהיה עז פורש מינימלי.

11. בשאלת זו שני סיעיפים א-ב שאינם תלויים זה זהה. ענה על שני הסיעיפים.

א. נתונה בעיית תכנון לינארית:

$$\max \{z = kx_1 + 20x_2\}$$

בכפוף לאיולוצים האלה:

$$-3x_1 + 4x_2 \leq 24$$

$$x_1 + 2x_2 \geq 0$$

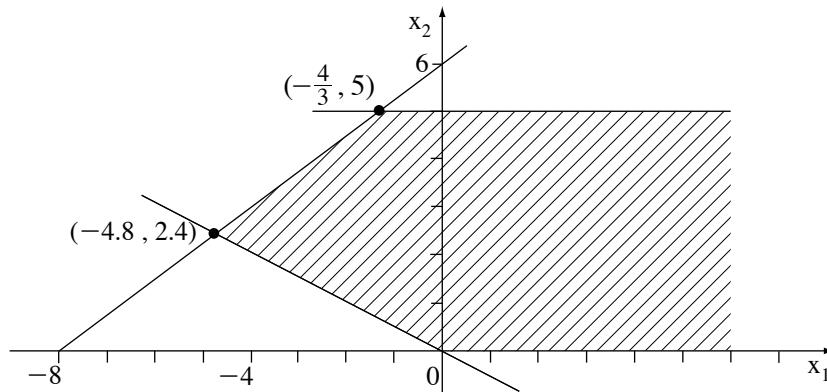
$$x_2 \leq 5$$

$$x_2 \geq 0$$

x_1 אינו מוגבל בסימן.

k הוא פרמטר.

לפניך סרטוט של תחום הפתרונות האפשריים של הבעיה הנתונה.



לפניך ארבעה תת-סיעיפים (1)-(4), שבכל אחד מהם נתון ערך מסוים של הפרמטר k .
 (1) . $k = 20$

(2) . $k = -20$

(3) . $k = -15$

(4) . $x_1 \leq -5$, ומוסיפים את האילוץ: $x_1 \leq -5$
התת-סיעיפים איןט תלויים זה זהה.

בעבור כל אחד מהתת-סיעיפים (1)-(4):

— אם הפתרון האופטימלי הוא יחיד, עליך למצוא את הפתרון האופטימלי היחיד, ואת הערך של פונקציית המטרה בפתרון זה.

— אם יש אין-סוף פתרונות אופטימליים, עליך לרשום את הפתרון האופטימלי הכללי לבעיה, ואת הערך של פונקציית המטרה בפתרון זה.

— אם הפתרון האופטימלי לא חסום, נמק את קביעותך.

— אם אין פתרון אפשרי, נמק את קביעותך.

ב. (אין קשר לסעיף א.)

נתונה בעיית תכנון לינארי:

$$\max \{z = ax_1 + 3x_2\}$$

בכפוף לאילוצים האלה:

$$x_1 + \frac{3}{8}x_2 \leq 3$$

$$x_1 + x_2 \leq 5$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

$0 < a$ הוא פרמטר של הבעיה.

לפניך סרטוט של תחום הפתרונות האפשריים של הבעיה הנתונה.



לפניך ארבעה תחומיים, i-iv , בעבור הפרמטר a :

$$a < 3 \quad \text{i}$$

$$a > 8 \quad \text{ii}$$

$$3 < a < 8 \quad \text{iii}$$

$$3 < a < 24 \quad \text{iv}$$

קבע בעבור איזה מבין התחומיים i-iv , הפתרון $(1.8, 3.2)$ יהיה פתרון אופטימלי **יחיד**. העתק את התחום למחברתך, ונמك את קביעתו.

12. בשאלת זו חמשה סעיפים, א-ה, שאינם תלויים זה זהה. ענה על כל הסעיפים.

א. בטבלה שלפניך נתון חלק מפתרון בסיסי אפשרי לביעית תובלה:

$$x_{21} = 20, \quad x_{11} = 80$$

מקורות	יעדים			היצע
	1	2	3	
1	10 80	15	17	80
2	10 20	18	14	120
3	15	20	18	50
ביקוש	100	100	50	

העתק את הטלבה למחברתך, והשלם אותה לפי שיטת הפינה הצפונית מערבית.

ב. בטבלה שלפניך נתון חלק מפתרון בסיסי אפשרי לביעית תובלה, ונתונים ערכיים של

$$u_3, u_2, u_1, v_3, v_2, v_1$$

מקורות	יעדים			היצע	u_i
	1	2	3		
1	10 80	15	12	80	0
2	10	18 100	10	100	3
3	20	10	14 50	50	-5
ביקוש	80	100	50		
v_j	10	15	19		

העתק את הטלבה למחברתך, והשלם אותה לפי שיטת הפינה הצפונית מערבית מתוך

התוצאות בערכים של u_i -ים ו- v_j -ים, כך שיתקבל פתרון בסיסי אפשרי.
/המשך בעמוד 29/

ג. בטבלה שלפניך נתון פתרון בסיסי אפשרי לביעית תובלה, ונתונים ערכיים של

$$\cdot \quad u_3, u_2, u_1, v_3, v_2, v_1$$

מקורות	יעדים			היצע	u_i
	1	2	3		
1	10 100	18	13 30	130	0
2	6 100	21	12	100	-4
3	11	16 50	14 100	150	1
ביקוש	200	50	130		
v_j	10	15	13		

האם הפתרון הוא אופטימלי? נמק את תשובה.

- . ८. בטבלה שלפניך נתון פתרון אופטימלי לבעיתת תובלה, ונתונים ערכיים של $v_1, v_2, v_3, u_1, u_2, u_3$, שמתאימים לפתרון זה.

מקורות	יעדים			היצע	u_i
	1	2	3		
1	10 20	25	16	20	10
2	10 30	22	14 20	50	10
3	18 40	20 20	20 20	60	16
ביקוש	50	40	40		
v_j	0	4	4		

(1) האם הפתרון הנתון הוא אופטימלי ייחיד? נמק את תשובהך.

(2) בטבלה הנתונה בראש הסעיף החסר 4 מהעלות ליחידה מכל מקור לכליעד. סרטט את הטבלה שתתקבל.

קבע איזה מרבעת הגידים 1-z שלפניך הוא הנכון.

העתק אותו למחברתך, ונמק את בחירתך.

i הפתרון הנתון אינו פתרון אופטימלי בעבור הבעה שהתקבלה.

ii הפתרון הנתון הוא פתרון בסיסי אפשרי אך אינו אופטימלי בעבור הבעה שהתקבלה.

iii הפתרון הנתון הוא פתרון אופטימלי ייחיד בעבור הבעה שהתקבלה.

iv הפתרון הנתון אינו פתרון אופטימלי אך אינו פתרון אופטימלי ייחיד בעבור הבעה שהתקבלה.

ה. בטבלה שלפניך נתון פתרון לא אופטימלי שהתקבל לאחר k איטרציות בעבר בעיית תובלה מסוימת, ונתונים ערכיהם של $v_1, v_2, v_3, v_4, u_1, u_2, u_3$.

מקורות	יעדים				היצע	u_i
	1	2	3	4		
1	5 150	2 80	4 -4	2 -6	230	0
2	2 70	3 4	5 80	3 -2	150	-3
3	7 7	4 7	3 200	3 340	540	-5
ביקוש	220	80	280	340		$Z = 3070$
v_j	5	2	8	8		

פתרונות המתואר בטבלה זו אינו אופטימלי כיון שיש משתנים מחוץ לבסיס

שערץ ה- $(v_j - u_i - c_{ij})$ שלהם שלילי.

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

(1) מהו המסתנה שיוצא מהבסיס באיטרציה זו?

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

מודלים חישוביים

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

. 13. לפניך השפה L מעל הא"ב $\{a, b, c\}$:

$$L = \{a^n b^{3k+1} c^k \mid n > 0, k > 0\}$$

. א. כתוב את המילה הקצורה ביותר בשפה L .

. ב. בנה אוטומט מחסנית שיקבל את השפה L .

. 14. בשאלת זו שני סעיפים א-ב, שאין קשר ביניהם. ענה על שניים.

. א. לפניך השפה L מעל הא"ב $\{a, b\}$:

$$L = \{w \mid |w| > 0 \text{ ו } w \text{ מופיע בכל המילה מספר זוגי של פעמים, } 0 \text{ לפחות}\}$$

לדוגמא:

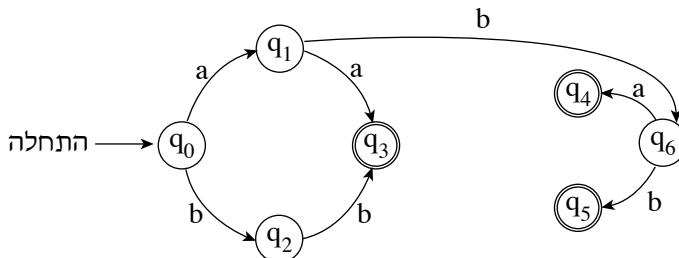
המילה $abaabbab$ ש匹כת לשפה.

המילה $abbabaa$ ש匹כת לשפה.

המילה $babbaba$ לא ש匹כת לשפה.

. לפניך סרטוט חלקי של אוטומט סופי דטרמיניסטי המקבל את השפה L .

בסרטוט חסרים מעברים וסימני קלט.



הסרטוט מכיל את כל המ מצבים של האוטומט, ואת כל המ מצבים המתקבלים.

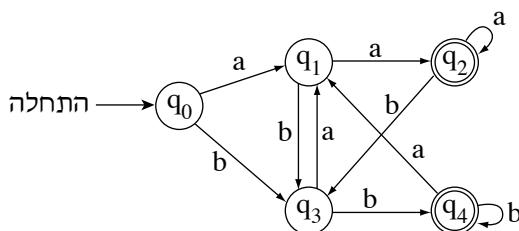
העתק למחברתך את הסרטוט, והשלם אותו כך שהאוטומט יקבל את השפה L .

עליך להשלים את המעברים החסרים ואת סימני הקלט החסרים.

שים לב: אין להוסיף מצבים לאוטומט או להוריד ממנו מצבים.

ב. (אין קשר לסעיף א.)

לפניך אוטומט סופי דטרמיניסטי המקבל את שפה L מעל הא"ב $\{a, b\}$.



(1) קבע לכל אחת מארבע המיללים (i)-(iv) שלפניך אם היא מתתקבלת על ידי האוטומט. אם המילה מתתקבלת על ידי האוטומט, רשום את המסלול המתקבל בעבורו מילה זו.

aaba **(i)**

bbaabb **(ii)**

abaa **(iii)**

bb **(iv)**

(2) מהי השפה L המוגדרת על ידי האוטומט?

. 15. לפניך השפה L מעל הא"ב $\{0, 1, 2\}$:

$$L = \{0^n 1^k 2 \mid n > k \geq 0\}$$

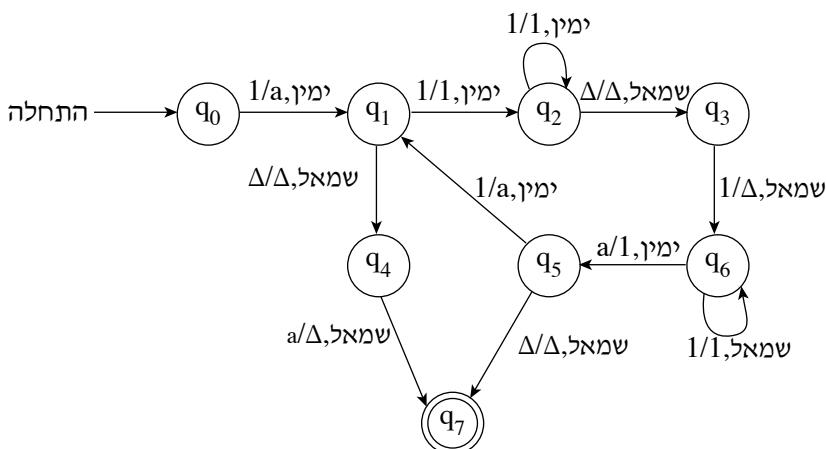
א. הוכח כי השפה L איינה רגולרית.

ב. נתונה השפה L_1 מעל הא"ב $\{1, 2\}$:

$$L_1 = \{1^n 2 \mid n \geq 0\}$$

מהי השפה $L \cap L_1$? נמק.

- לפניך מכונת טיורינג המכשפת פונקציה $f(x)$.
- המכונה מקבלת קלט מספר x שלם וגדול מ-0, הרשום על הסרט כמספר אונרי על ידי x תווים של 1, ואחריהם הסימן Δ .
- המכונה רושמת את תוצאה החישוב של $f(x)$ על הסרט כמספר אונרי אחרי הסימן \vdash , ולאחר המספר מופיע הסימן Δ .



- a. מה יכול הסרט לאחר חישוב $f(3)$? הראה את מסלול החישוב של המכונה.
- בכל שלב רשום את מצב הסרט, סמן היכן נמצא ראש המכונה, ורשום באיזה מהמצבים $q_0 - q_7$ המכונה נמצאת.
- b. מה יכול הסרט לאחר חישוב $f(5)$?
- c. מה יכול הסרט לאחר חישוב $f(6)$?
- d. מה הפונקציה $f(x)$ שהמכונה מחשבת?
- e. הוסף למכונה מעבר/ מעברים כדי שתחשב גם את $f(0)$. כתוב במחברתך את המעבר/ מעברים שהוספה.
- לכל מעבר רשום: מאיזה מצב הוא יוצא, לאיזה מצב הוא מגע ומה רשום עליו.

שים לב: תכנות מונחה עצמים מתחילה בעמוד 36.

תכונות מונחה עצמים

אם למדת מסלול זה ואתה כותב ב- Java, ענה על שתיים מהשאלות 17-20.
(כל שאלת – 25 נקודות)

.17. לפניך המחלקות **AA** ו- **BB** :

```
public class AA
{
    private String st;

    public AA()          { this.st = "excellent"; }
    public AA(String st) { this.st = st; }
    public String getSt() { return this.st; }
    public void setSt(String st) { this.st = st; }
    public String toString() { return "st = " + this.st; }
}

public class BB extends AA
{
    private int num;

    public BB()          { super(); this.num = 1; }
    public BB(int num, String st) { super(st); this.num = Math.abs(num); }
    public int getNum()   { return this.num; }
    public void setNum(int num) { this.num = num; }
    public String toString() { return super.toString() + " num = " + this.num; }
}
```

.א. הגדר במחלקה **AA** פעליה בוליאנית בשם isLike(Object obj) המקבלת עצם obj

מтиיפוס Object . אם העצם obj הינו מטיפוס **AA** וגם תוכן המחרוזת של obj זהה לתוכן המחרוזת st של העצם הנוכחי – הפעולה תחזיר true, אחרת – false.

.ב. הגדר במחלקה **BB** פעליה הדורשת את הפעולה שהגדרת בסעיף א.

אם העצם obj הינו מטיפוס **BB** וגם ערך התכונה num שלו זהה לערך התכונה num של העצם הנוכחי – הפעולה תחזיר true , אחרת – false
/המשך בעמוד 37/

ג. לפני קטיעת פעולה ראשית:

```
AA a = new AA("excellent");
BB b = new BB();
a = b;
if (a.isLike(b)) System.out.println(a);
```

האם קטיעת התוכנית תקין?

אם כן — מה יהיה פלט הקטיע? רשוםஇயோ גרסה של הפעולה `isLike` תופעל — זו של `AA` או זו של `BB`.
אם לא — הסבר מהי השגיאה ומתי היא תתגללה: בזמן קומpileציה או בזמן ריצה.

ד. לפני קטיעת פעולה ראשית:

```
AA aa = new AA();
BB bb = new BB(2 , "excellent");
bb = aa;
if (bb.isLike(aa)) System.out.println(bb);
```

האם קטיעת התוכנית תקין?

אם כן — מה יהיה פלט הקטיע? רשוםஇயோ גרסה של הפעולה `isLike` תופעל — זו של `AA` או זו של `BB`.
אם לא — הסבר מהי השגיאה ומתי היא תתגללה: בזמן קומpileציה או בזמן ריצה.

ה. כתוב פעולה חיצונית בשם `longString` המקבלת מערך של עצמים מטיפוס `Object`.

הפעולה מחזירה מחרוזת המורכבת משרשור התוכונה `st` של עצמים מטיפוס `AA` במבנה, באופן זהה:

- אם לעצם יש בק התוכונה `st`, נשורשר המחרוזות שבתוכונה `st` פעם אחת.
- אם לעצם יש גם התוכונה `sumt`, המחרוזות שבתוכונה `st` נשורשר `num` פעמיים.
- אם אין במבנה אף עצם מטיפוס `AA`, תוחזר מחרוזת ריקה.

לפניך פרויקט ובו המחלקות: **OopTest** , **A** , **D** , **B**

public class **B**

{

 private static int numB = 0;

 private int m1;

 private int m2;

 public **B**(int m1, int m2)

{

 this.m1 = m1;

 this.m2 = m2;

 numB++;

 System.out.println("B(" + m1 + ", " + m2 + ") , #" + numB);

}

}

public class **D** extends **B**

{

 private static int numD = 0;

 private double d;

 public **D**(double d, int x)

{

 super(x, x);

 this.d = d;

 numD++;

 System.out.println("D(" + d + ", " + x + ") , #" + numD);

}

 public **D**(double d, int x, int y)

{

 super(x, y);

 this.d = d;

 numD++;

 System.out.println("D(" + d + ", " + x + ", " + y + ") , #" + numD);

}

}

```
public class A
```

```
{
```

```
    private static int numA = 0;
```

```
    private A a;
```

```
    private B b;
```

```
    public A(A a, B b)
```

```
{
```

```
        this.a = a;
```

```
        this.b = b;
```

```
        numA++;
```

```
        System.out.println("A Constructor , #" + numA);
```

```
}
```

```
}
```

```
public class OopTest
```

```
{
```

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

```
{
```

```
        B w1 = new B(2, 3);
```

```
        B w2 = new D(1.5, 6);
```

```
        B w3 = new D(2.3, 8, 9);
```

```
        A w4 = new A(null, w1);
```

```
        A w5 = new A(w4, w3);
```

```
}
```

```
}
```

כתב מעקב אחר הפעולה main במחלקה **OopTest** , וכותב את הפלט.

במעקב יש לכתוב את ערכי המשתנים, ובעבור כל עצם – את ערכי התכונות שלו.

/המשך בעמוד 40/

19. חברת המפיתה לומדות פיתחה לומדה העוסקת בסדרות של מספרים שלמים.

המערכת פותחה בשלבים.

בכל סדרה של מספרים מתייחסים אל:

- (1) האיבר הראשון בסדרה שמספרו הסידורי הוא 1.
- (2) האיבר שמספרו הסידורי בסדרה הוא 2.
- (3) הדפסת כל האיברים הראשונים בסדרה.

בשלב הראשון פותחו שתי מחלקות:

סדרה חשבונית (ASeq) – סדרה שבה ההפרש בין כל איבר לקודמו הוא ערך קבוע.

סדרה הנדסית (GSeq) – סדרה שבה המנה בין כל איבר לקודמו היא ערך קבוע.

להלן קוד המחלקות שפותחו בשלב הראשון:

```
public class ASeq
```

```
{
```

```
    private int first;
```

```
    private int difference;
```

```
    public ASeq(int first, int difference)
```

```
{
```

```
        this.first = first;
```

```
        this.difference = difference;
```

```
}
```

```
    public int theNElement(int n)
```

```
{
```

```
        return this.first + (n-1) * this.difference;
```

```
}
```

```

public void displayNElements(int n)
{
    System.out.print("The sequence elements: ");
    for (int i = 0; i < n - 1; i++)
        System.out.print(this.theNElement(i+1) + " , ");
    System.out.println(this.theNElement(n));
}

public class GSeq
{
    private int first;
    private int product;

    public GSeq(int first , int product)
    {
        this.first = first;
        this.product = product;
    }

    public int theNElement(int n)
    {
        return this.first * (int)Math.pow(this.product , n-1);
    }

    public void displayNElements(int n)
    {
        System.out.print("The sequence elements: ");
        for (int i = 0 ; i < n-1; i++)
            System.out.print(this.theNElement(i+1) + " , ");
        System.out.println(this.theNElement(n));
    }
}

```

(שים לב: המשך השאלה בעמוד הבא.)

א. עקוב אחר קטע התכנית שלפניך. במקבץ הצג את העצם שנבנה, את התוכנות שלו ואת הפלט.

```
ASeq aSeq = new ASeq(2 , 3);
System.out.println(aSeq.theNElement(4));
aSeq.displayNElements(5);
```

בשלב השני של הפיתוח הוחלט שמתאים לפתח מחלקה חדשה המתארת סדרה קבועה (**Sequence**), כך שהמחלקות **ASeq** ו- **GSeq** יירשו מן המחלקה החדשה. בסדרה קבועה מוגדר ערך האיבר הראשון, וכל יתר האיברים זהים לאיבר הראשון.

ב. השלים את הפיתוח של שלב השני באופן המתאים ביותר לעקרונות של תוכנות מונחה עצמים, ובהתאם להנחיות (i)-(ii):

(i) ממש באופן מלא את מחלקת העל **Sequence**. המחלקה צריכה להתייחס אל:

- (1) האיבר הראשון בסדרה שמספרו הסידורי 1.
- (2) האיבר שמספרו הסידורי בסדרה הוא 2.
- (3) הדפסת כל האיברים הראשונים של הסדרה.

(ii) ממש מחדש את המחלקה **ASeq** כך שתירוש מחלקת **Sequence**.

בשלב השלישי של הפיתוח הוחלט להרחיב את הפרוייקט שככלו את שלוש המחלקות שפותחו בשלב השני (**Sequence** , **ASeq** , **GSeq**), כך שבBOR כל סדרה יהיה אפשר להפעיל פעולה המחשבת ומחזירה את סכום כל האיברים הראשונים של הסדרה.

הנמ' כי המחלקה **GSeq** מומשה מחדש, כך שהיא יורשת מחלקת **Sequence**.

ג. בעבור כל אחת מן המחלקות **Sequence**, **ASeq** , **GSeq** , כתוב אם יש לעשות בה שינויים כך שהפרויקט יענה על דרישות הפיתוח של שלב השלישי באופן המתאים ביותר לעקרונות של תוכנות מונחה עצמים. אם יש לעשות שינויים — פרט ומש אוטם.

- בשלב הרביעי** של הפניות הוחלט לפתח פעולה סטטית `check` המקבלת ערך שלם n ,
ושני עצמים של סדרות: האחד מטיפוס **ASeq** והאחר מטיפוס **GSeq**.
הפעולה מחשבת את סכום π האיברים הראשונים בכל אחת משתי הסדרות ומהזירה:
 - את התו 'A' — אם סכום π האיברים הראשונים של הסדרה מטיפוס **ASeq** הוא גדול מבין שני הסכומים.
 - את התו 'G' — אם סכום π האיברים הראשונים של הסדרה מטיפוס **GSeq** הוא גדול מבין שני הסכומים.
 - את התו 'E' — אם סכום π האיברים הראשונים של שתי הסדרות שווה.
 - ד. ממש את הפעולה הסטטית `check` על פי הדרישות שהוגדרו בשלב הרביעי של הפניות.

20. מרפאה וטרינרית של חיות מחמד מרכזת מידע על הווטרינרים העובדים במרפאה ועל חיות המחמד המתופלות בה. מספר הווטרינרים העובדים במרפאה הוא לכל היוטר 10, ומספר חיות המחמד המתופלות במרפאה הוא לכל היוטר 500.

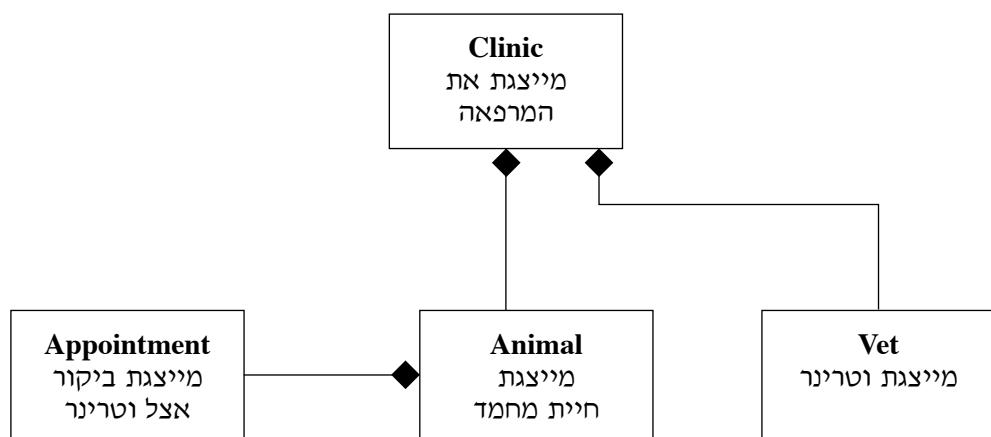
הקוויים המוחכים לניהול המידע הם:

- לכל וטרינר נשמר: מספר תעודת הזיהות, שם, ותיק בעבודה בשנים.
- לכל חיית מחמד נשמר: מספר הרישוי, שם, סוג (למשל: כלב, חתול או ארנב), גיל, פירוט של עד 50 הביקורים האחרונים במרפאה, מספר הביקורים השמורים.
- לכל ביקור במרפאה נשמר: מספר תעודת הזיהות של הווטרינר שטיפול בחיה מחמד בביוק זה, ומחרוזת של תווים המכילה קודים של טיפולים שחיה מחמד קיבלה באותו ביקור. קוד של טיפול הוא אות גדולה מותוך כל אותיות ה- ABC.

חלק מהפעולות שהמערכת יכולה לבצע הן:

- הפקט דוח של כל חיית מחמד מסווג מסוימים (למשל כל הארנבים). לכל חייה הדוח יכלול את מספר הרישוי שלו, את שמה ואת גילו.
- בתחלת שנה: עדכון הגיל של כל חיית מחמד, כולל הגדרת הגיל ב-1, ועדכון הוותק של כל וטרינר, כולל הגדרת הוותק ב-1.
- החזרת שם של וטרינר על פי מספר תעודת הזיהות שלו.

לפניך תרשימים של היררכיית המחלקות הנדרשות בעברור ניהול המידע של המרפאה:



הסימן —————♦ מייצג הכללה.

א. בעבר כל מחלקה בתרשים, הגדר את התכונות ואת הפעולות שלה.
עליך להגדיר רק את הפעולות הנחוצות כדי לענות על הדרישות שתוארו בפתחה לשאלת שבעמוד (44) ובתרשים.

הנה שנותנות פעולות בוניות, המקבלות פרמטר לכל תכונה, ופעולות מאחרות (פעולות `get`), ואין צורך לכתוב אותן. לא נתנו פעולות קבועות (פעולות `set`)
בעבור כל תכונה – רשום את ההגדרה שלה ב- Java , ורשום את התיעוד שלה.
בעבור כל פעולה – רשום את הcotरת שלה ב- Java , ורשום תיעוד הכלול מה היא מקבלת ומה היא מחזירה. אין צורך למש את הפעולה.

ב. הנה כי המערכת פותחה בהתאם לתכנון שהציג בסעיף א, וכל הפעולות הנתונות ואלו שהגדרת ממומשות.

במחלקה **Clinic** מוסיפים את הפעולה:

```
public void addAppointment (Animal p , String t , Vet v)
```

המקבלת חיית מוחמד k , מחרוזת t של קודים של טיפולים שקיבלה החיה בבדיקה הנוכחי, ואת הווטרינר v שטיפול בה. הפעולה מוסיפה את הביקור לחיה המוחמד. ממש באופן מלא את הפעולה.

הנה שמספר הביקורים הקודמים של החיה במרפאה קטן מ- 50 .

אם נוסף על הפעולות הנתונות ועל הפעולות שהגדרת בסעיף א אתה משתמש בפעולות אחרות, עלייך למש אותן באופן מלא ולצין בעבר כל פעולה באיזו מחלקה יש למש אותה.

תכנות מונחה עצמים

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

.21. לפניך המחלקות **AA** ו-**BB**:

public class **AA**

{

 private string st;

 public AA() { this.st = "excellent"; }

 public AA(string st) { this.st = st; }

 public string GetSt() { return this.st; }

 public void SetSt (string st) { this.st = st; }

 public override string ToString() { return "st = " + this.st; }

}

public class **BB** : **AA**

{

 private int num;

 public BB() : base() { this.num = 1; }

 public BB(int num, string st) : base(st) { this.num = Math.Abs(num); }

 public int GetNum() { return this.num; }

 public void SetNum(int num) { this.num = num; }

 public override string ToString() { return base.ToString() + " num = " + this.num; }

}

.א. הגדר במחלקה **AA** פועלה בוליאנית הניתנת לדרישה, בשם (Object obj) המקבלת עצם obj מטיפוס **Object**. אם העצם obj הינו מטיפוס **AA** וגם תוכן המחרוזת st של obj זהה לתוכן המחרוזת st של העצם הנוכחי – הפועלה תחזיר true, אחרת – תחזיר false.

.ב. הגדר במחלקה **BB** פועלה הדורשת את הפועלה שהגדרת בסעיף א. אם העצם obj הינו מטיפוס **BB** וגם ערך התכונה sum שלו זהה לערך התכונה num של העצם הנוכחי – הפועלה תחזיר true, אחרת – תחזיר false.

ג. לפניך קטע מפעולה ראשית:

```
AA a = new AA("excellent");
BB b = new BB();
a = b;
if (a.IsLike(b)) Console.WriteLine(a);
```

האם קטע התוכנית תקין?

אם כן — מה יהיה פלט הקטע? רשוםஇயோ גרסה של הפעולה IsLike תופעל — זו של AA או זו של BB.
אם לא — הסבר מהי השגיאה ומתי היא מתגללה: בזמן קומpileציה או בזמן ריצה.

ד. לפניך קטע מפעולה ראשית:

```
AA aa = new AA();
BB bb = new BB(2 , "excellent");
bb = aa;
if (bb.IsLike(aa)) Console.WriteLine(bb);
```

האם קטע התוכנית תקין?

אם כן — מה יהיה פלט הקטע? רשוםஇயோ גרסה של הפעולה IsLike תופעל — זו של AA או זו של BB.
אם לא — הסבר מהי השגיאה ומתי היא מתגללה: בזמן קומpileציה או בזמן ריצה.

ה. כתוב פעלת חיצונית בשם LongString המתקבלת מערך של עצמים מטיפוס Object. הפעלת מחזירה מחרוזת המורכבת משרשור התכונה st של עצמים מטיפוס AA במערך, באופן זה:
— אם לעצם יש רק התכונה st , תשורשר המחרוזות שבתכונה st פעם אחת.
— אם לעצם יש גם התכונה sum , המחרוזות שבתכונה st תשורשר sum פעמיים.
— אם אין במערך אף עצם מטיפוס AA , תוחזר מחרוזת ריקה.

לפניך פרויקט ובו המחלקות **A**, **B** ו- **OopTest**.

```
public class B
{
    private static int numB = 0;
    private int m1;
    private int m2;

    public B(int m1, int m2)
    {
        this.m1 = m1;
        this.m2 = m2;
        numB++;
        Console.WriteLine("B(" + m1 + ", " + m2 + ") , #" + numB);
    }
}

public class D : B
{
    private static int numD = 0;
    private double d;

    public D(double d, int x) : base (x , x)
    {
        this.d = d;
        numD++;
        Console.WriteLine("D(" + d + ", " + x + ") , #" + numD);
    }

    public D(double d, int x, int y) : base(x , y)
    {
        this.d = d;
        numD++;
        Console.WriteLine("D(" + d + ", " + x + " , " + y + ") , #" + numD);
    }
}
```

```

public class A
{
    private static int numA = 0;
    private A a;
    private B b;

    public A(A a, B b)
    {
        this.a = a;
        this.b = b;
        numA++;
        Console.WriteLine("A Constructor , #" + numA);
    }
}

public class OopTest
{
    public static void Main(string[] args)
    {
        B w1 = new B(2, 3);
        B w2 = new D(1.5, 6);
        B w3 = new D(2.3, 8, 9);
        A w4 = new A(null, w1);
        A w5 = new A(w4, w3);
    }
}

```

כתב מעקב אחר הפעולה Main במחלקה **OopTest** , וכותב את הפלט.
במעקב יש ל כתוב את ערכי המשתנים, ובעבור כל עצם – את ערכי התכונות שלו.

.23. חברת המפיקת לומדות פיתחה לומדה העוסקת בסדרות של מספרים שלמים. המערך פותחה בשלבים.

עבור כל סדרה של מספרים מתייחסים אל:

- (1) האיבר הראשון בסדרה שמספרו הסידורי הוא 1.
- (2) האיבר שמספרו הסידורי בסדרה הוא 2.
- (3) הדפסת כל האיברים הראשונים בסדרה.

בשלב הראשון פותחו שתי המחלקות:

סדרה חשבונית (**ASeq**) – סדרה שבה הפרש בין כל איבר לקודמו הוא ערך קבוע.
סדרה הנדסית (**GSeq**) – סדרה שבה המנה בין כל איבר לקודמו היא ערך קבוע.

להלן קוד המחלקות שפותחו בשלב הראשון:

```
public class ASeq
```

```
{
```

```
    private int first;
```

```
    private int difference;
```

```
    public ASeq(int first, int difference)
```

```
{
```

```
        this.first = first;
```

```
        this.difference = difference;
```

```
}
```

```
    public int TheNElement(int n)
```

```
{
```

```
        return this.first + (n - 1) * this.difference;
```

```
}
```

```

public void DisplayNElements(int n)
{
    Console.WriteLine("The sequence elements: ");
    for (int i = 0; i < n - 1; i++)
        Console.WriteLine(this.TheNElement(i + 1) + " , ");
    Console.WriteLine(this.TheNElement(n));
}

public class GSeq
{
    private int first;
    private int product;

    public GSeq(int first, int product)
    {
        this.first = first;
        this.product = product;
    }

    public int TheNElement(int n)
    {
        return this.first * (int)Math.Pow(this.product, n - 1);
    }

    public void DisplayNElements(int n)
    {
        Console.WriteLine("The sequence elements: ");
        for (int i = 0; i < n - 1; i++)
            Console.WriteLine(this.TheNElement(i + 1) + " , ");
        Console.WriteLine(this.TheNElement(n));
    }
}

```

א. עקוב אחר קטע התוכנית שלפניך. במקבץ הצג את העצם שנבנה, את התכונות שלו ואת הפלט.

```
ASeq aSeq = new ASeq(2 , 3);
Console.WriteLine(aSeq.TheNElement(4));
aSeq.DisplayNElements(5);
```

בשלב השני של הפיתוח הוחלט שמתאים לפתח מחלקה חדשה המותאמת סדרה קבועה (**Sequence**), כך שהמחלקות **ASeq** ו- **GSeq** יירשו מן המחלקה החדשה. בסדרה קבועה מוגדר ערך האיבר הראשון, וכל יתר האיברים זהים לאיבר הראשון.

ב. השלם את הפיתוח של שלב השני באופן המתאים ביותר לעקרונות של תכונות מונחה עצמים ובהתאם להנחיות (i)-(ii):

(i) ממש באופן מלא את מחלקת העל **Sequence**. המחלקה צריכה להתייחס אל:

- (1) האיבר הראשון בסדרה שמספרו הסידורי 1.
- (2) האיבר שמספרו הסידורי בסדרה הוא ז.
- (3) הדפסת ז האיברים הראשונים של הסדרה.

(ii) ממש מחדש את המחלקה **ASeq** כך שתירוש מן המחלקה **Sequence**.

בשלב השלישי של הפיתוח הוחלט להרחיב את הפרויקט שכולל את שלוש המחלקות שפותחו בשלב השני (**Sequence** , **ASeq** , **GSeq**), כך שבמעבר כל סדרה יהיה אפשר להפעיל פעולה המחשבת ומהזירה את סכום ז האיברים הראשונים של הסדרה. הנתן כי המחלקה **GSeq** מומשה מחדש כך שהיא יורשת מן המחלקה **Sequence**.

ג. מעבור כל אחת מן המחלקות כך שהיא יונחה עצמים. כתוב אם יש לעשות בה שינויים כך שהפרויקט יענה על דרישות הפיתוח של שלב השלישי באופן המתאים ביותר לעקרונות של תכונות מונחה עצמים. אם יש לעשות שינויים — פרט וממש אותם.

- בשלב הרביעי** של הפיתוח הוחלט לפתח פעולה סטטיט Check המקבלת ערך שלם או שני עצמים של סדרות: האחד מטיפוס **ASeq** והאחר מטיפוס **GSeq**. הפעולה מחשבת את סכום מה האיברים הראשונים בכל אחת משתי הסדרות ומחזירה:
- את התו 'A' – אם סכום מה האיברים הראשונים של הסדרה מטיפוס **ASeq** הוא גדול מבין שני הסכומים.
 - את התו 'G' – אם סכום מה האיברים הראשונים של הסדרה מטיפוס **GSeq** הוא גדול מבין שני הסכומים.
 - את התו 'E' – אם סכום מה האיברים הראשונים של שתי הסדרות שווה.
2. ממש את הפעולה הסטטיט Check על פי הדרישות שהוגדרו בשלב הרביעי של הפיתוח.

24. מרפאה וטורינרית של חיות מחמד מרכזת מידע על הווטרינרים העובדים במרפאה ועל חיות המחמד המטופלות בה. מספר הווטרינרים העובדים במרפאה הוא לכל היוטר 10 , ומספר חיות המחמד המטופלות במרפאה הוא לכל היוטר 500 .

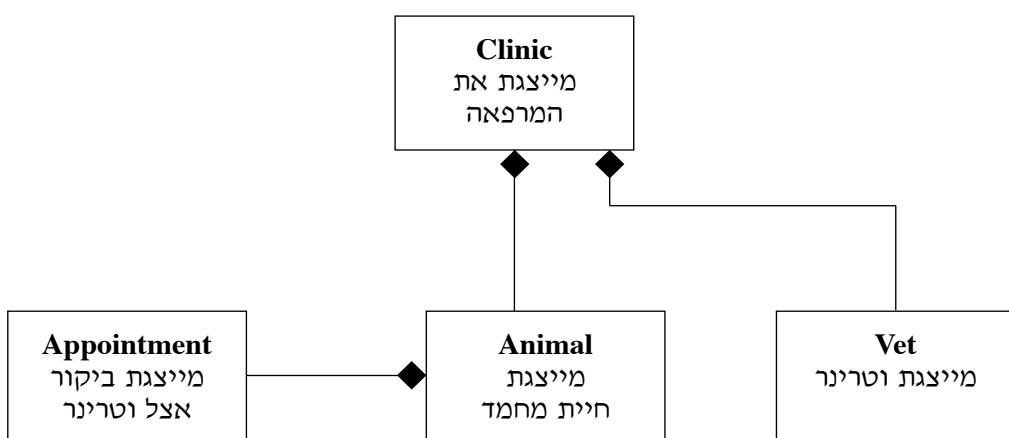
הकווים המנחים לניהול המידע הם:

- לכל וטורינר נשמר: מספר תעודת הזהות, שם, ותק בעבודה בשנים.
- לכל חיית מחמד נשמר: מספר הרישוי, שם, סוג (למשל: כלב, חתול או ארנב), גיל, פירוט של עד 50 הביקורים האחוריים במרפאה, מספר הביקורים השמורים.
- בכל ביקור במרפאה נשמר: מספר תעודת הזהות של הווטרינר שטיפול בחיות המחמד בבייקור זה, ומחזורות של תווים המכילה קודים של טיפולים שחיהת המחמד קיבלה באותו ביקור. קוד של טיפול הוא אות גדולה מתוך כל אותיות ה- ABC .

חלק מהפעולות שהמערכת יכולה לבצע הן:

- הפקט דוח של כל חיית המחמד מסווג מסוימים (למשל כל הארנבים). לכל חייה הדוח יכלול את מספר הרישוי שלו, את שמה ואת גילו.
- בתחלת כל שנה: עדכון הגיל של כל חיית מחמד, קלומר הגדלת הגיל ב-1, ועדכון הוותק של כל וטורינר, קלומר הגדלת הוותק ב-1 .
- החזרת שם של וטורינר על פי מספר תעודת הזהות שלו.

לפניך תרשימים של היררכיית המחלקות הנדרשות בעבר ניהול המידע של המרפאה:



הסימן —————♦ בתרשימים מייצג הכללה.

א. בעבר כל מחלקה בתרשימים, הגדר את התוכנות ואת הפעולות שלה. עליך להגדיר רק את הפעולות הנחוצות כדי לענות על הדרישות שתוארו בפתחה לשאלת (שבעמוד 54) ובתרשים.

הנח נתוננות פעולות בונות, המקבלות פרמטר לכל תוכנה, ופעולות מאחזורת (פעולות Get), ואין צורך לכתוב אותן. לא נתונות פעולות קובעות (פעולות Set). בעבר כל תוכנה – רשום את ההגדרה שלה ב- C#, ורשום את התיעוד שלה. בעבר כל פעולה – רשום את הcotורת שלה ב- C#, ורשום תיעוד הכלול מה היא מקבלת ומה היא מחזירה. אין צורך למש את הפעולה. **ב.** הנח כי המערכת פותחה בהתאם לתוכנו שהציג בסעיף א, וכל הפעולות הנתונות ואלו שהגדרת ממושכות.

במחלקה **Clinic** מוסיפים את הפעולה:

```
public void AddAppointment (Animal p , string t , Vet v)
```

המקבלת חיית מוחמד k , מחרוזת t של קודים של טיפולים שקיבלה החיה בבדיקה הנוכחי ואת הווטרינר שטיפול בה. הפעולה מוסיפה את הביקור לחיה המוחמד. ממש באופן מלא את הפעולה.

הנח שמספר הביקורים הקודמים של החיה במרפאה קטן מ- 50 .

אם נוסף על הפעולות הנתונות ועל הפעולות שהגדרת בסעיף א אתה משתמש בפעולות אחרות, עליך למש אותן באופן מלא ולציין בעבר כל פעולה באיזו מחלקה יש למש אותה.

בהצלחה!

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