GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: קיץ 2024 מועד א'
סמל השאלון: 97105
שם התלמיד/ה: _____

מבנה נתונים ותכנות מונחה עצמים

הנדסת תוכנה · הנדסאים וטכנאים

הוראות

א. משך הבחינה: ארבע שעות וחצי.

ב. מבנה השאלון ומפתח ההערכה: **בשאלון זה 2 מבחנים, עליכם לענות על מבחן אחד בלבד בהתאם למוסד הלימודים:** מבחן ב-Java (עמוד 2) מבחן ב-C# (עמוד 12) בכל מבחן 12 שאלות. **חלק א' – 48 נקודות** שאלות 1-5: מתוכן יש לענות על **3** שאלות בלבד. ערך כל שאלה 16 נקודות. **חלק ב' – 36 נקודות** שאלות 6-10: מתוכן יש לענות על **2** שאלות בלבד. ערך כל שאלה 18 נקודות. **חלק ג' – 16 נקודות** שאלות 11-12: מתוכן יש לענות על שאלה **1** בלבד. ערך שאלה 16 נקודות. **בסך הכול: 100 נקודות.**

ג. חומר עזר מותר בשימוש: 1. מחשבון (אין להשתמש במחשב כף יד או במחשבון עם תקשורת חיצונית). 2. קלסר **אחד בלבד** עם **חומר ההרצאות**. אין להוציא דפים מהקלסר. **אין לצרף ספרים או חוברות עם פתרונות.**

ד. הוראות מיוחדות: 1. יש לקרוא בעיון את ההנחיות בדף השער ואת כל שאלות הבחינה, ולוודא שהן מובנות. 2. את התשובות יש לכתוב בצורה מסודרת, בכתב יד ברור ונקי (גם בכך תלויה הערכת הבחינה). 3. יש להשאיר את העמוד הראשון במחברת הבחינה ריק. בסיום המבחן יש לרשום בעמוד זה את מספרי התשובות לבדיקה. התשובות ייבדקו לפי סדר כתיבתן בעמוד זה. לא ייבדקו תשובות עודפות. 4. יש לכתוב את התשובות במחברת הבחינה **בעט בלבד**, בכתב יד ברור. 5. יש להתחיל כל תשובה בעמוד חדש ולציין את מספר השאלה ואת הסעיף. אין צורך להעתיק את השאלה עצמה. 6. טיוטה יש לכתוב במחברת הבחינה בלבד. יש לרשום את המילה "טיוטה" בראש העמוד ולהעביר עליו קו כדי שלא ייבדק. 7. יש להציג פתרון מלא ומנומק, כולל חישובים לפי הצורך. הצגת תשובה סופית ללא שלבי הפתרון לא תזכה בניקוד. 8. יש להסביר בפירוט כל תוכנית שנכתבה, **תוכנית ללא הסבר מפורט לא תזכה בניקוד.** 9. אם לדעתכם חסר בשאלה נתון, יש לציין זאת ולהוסיף נתון מתאים שיאפשר לכם להמשיך בפתרון השאלה, נמקו את בחירתכם. **חל איסור מוחלט להוציא שאלון או מחברת בחינה מחדר הבחינה!** בשאלון זה 21 עמודי בחינה ו-7 עמודי נספחים: נספח ממשקים לבחינה JAVA, נספח ממשקים לבחינה C# ומילון עזר.

השאלות בשאלון זה מנוסחות בלשון רבים, אף על פי כן על כל תלמידה וכל תלמיד להשיב עליהן באופן אישי.

בהצלחה!

השאלות

1. **א.** מחסנית של מספרים שלמים נקראת "מאוזנת" אם יש לה אותה כמות מספרים גם חיוביים, גם שליליים וגם אפסים.

כתבו פעולה המקבלת מחסנית של מספרים שלמים ובודקת אם היא "מאוזנת". יש לשמור על תוכן של המחסנית אחרי ביצוע הפעולה.

כותרת הפעולה:

Java:

```
public static boolean isBalanced(Stack<Integer> st)
```

C#:

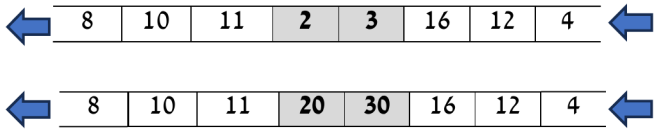
```
public static bool IsBalanced(Stack<int> st)
```

(12 נקודות)

ב. מהי הסיבוכיות של הפעולה שכתבתם בסעיף א'? **הסבירו את תשובתכם.**

(4 נקודות)

2.



א. כתבו פעולה המקבלת תור של מספרים שלמים ו-2 מספרים נוספים x ו-y. אם אורך התור זוגי, הפעולה **תחליף** את 2 האיברים האמצעיים לערכים x ו-y. אם אורך התור אי-זוגי, הפעולה תיצור מספר אקראי בין x ל-y (כולל) **ותחליף** את האיבר האמצעי של התור לערך אקראי שנוצר. שאר איברי התור יישארו ללא שינוי.

לדוגמה:

עבור התור q הבא ופרמטרים x=20, y=30

התור q אחרי זימון הפעולה יהיה

(12 נקודות)

ב. מהי הסיבוכיות של הפעולה שכתבתם בסעיף א'? **הסבירו את תשובתכם.**

(4 נקודות)

3. סדרה יורדת היא הסדרה שכל איבר החל באיבר ה-3 שווה להפרש של 2 קודמיו בערך מוחלט. לדוגמה:

אם 2 האיברים של הסידרה הם 55 ו-34 אז סדרה יורדת תהיה:

55 34 21 13 8 5 3 2 1 1 0 1 1 ....

אם 2 האיברים של הסידרה הם 15 ו-40 אז סדרה יורדת תהיה:

15 40 25 10 15 5 10 5 5 0 ...

תזכורת: לחישוב הפרש יש להשתמש בפונקציה של ערך מוחלט (**Math.abs**)

א. כתבו פעולה המקבלת הפנייה לשרשרת חוליות של מספרים שלמים. הפעולה תבדוק אם ערכים של השרשרת הם **סדרה יורדת**. אם כן - הפעולה תחזיר ערך true, ולא - הפעולה תחזיר ערך false.

כותרת הפעולה:

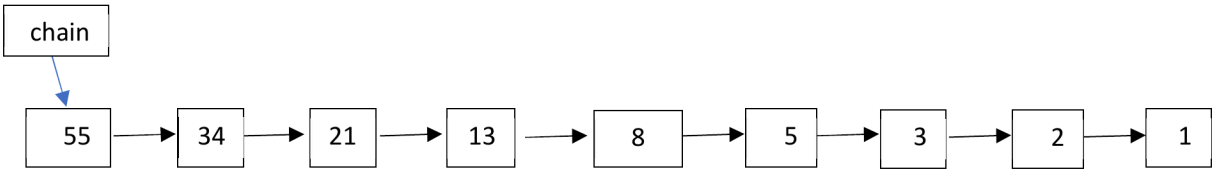
Java:

```
public static boolean isDownList (Node<Integer> chain)
```

C#:

```
public static bool IsDownList (Node<int> chain)
```

(6 נקודות)



ב. כתבו פעולה המקבלת מספר שלם וחיובי num וזוג מספרים שלמים a ו-b. הפעולה תחזיר הפנייה לחוליה הראשונה של שרשרת חוליות שאבריה הם num המספרים הראשונים של סדרה יורדת המתחילה במספרים a ו-b.

כותרת הפעולה:

Java:

```
public static Node<Integer> buildDown (int num, int a, int b)
```

C#:

```
public static Node<int> BuildDown (int num, int a, int b)
```

לדוגמה: עבור הזימון buildDown(9, 55, 34) הפעולה תחזיר הפנייה לחוליה הראשונה של שרשרת החוליות הבאה:

(6 נקודות)

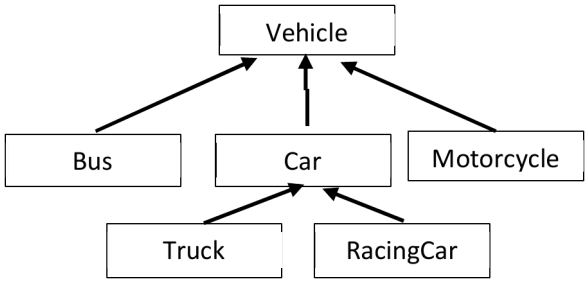
ג. מהי הסיבוכיות של הפעולות שכתבתם בסעיפים א' ו-ב'? **הסבירו את תשובתכם.**

(4 נקודות)

4.

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

נתון עץ הירושה הבא המתאר סוגי כלי תחבורה שונים:



א. לפניכם קטע קוד מתוך הפעולה הראשית:

```
Vehicle v1 = new Vehicle();
Vehicle v2 = new Car();
Vehicle v3 = new Truck();
Car c = new Truck ();
```

לפניכם 8 קטעי קוד. עבור כל אחד מהסעיפים יש לציין, אם הקוד תקין או לא. אם הקוד אינו תקין - יש להסביר למה הוא אינו תקין ולציין את סוג השגיאה (שגיאת קומפילציה / הידור או שגיאת זמן ריצה).

```
1. Car c1 = new Vehicle ();
2. Car c2 = v1;
3. Car c3 = (Car) v2;
4. Car c4 = (Car) v3;
5. Car c5 = (Car) v1;
6.
  Motorcycle m = new Motorcycle();
  Vehicle v6 = m;
  Car c6 = (Car) s6;
7. Vehicle v7 = (Car)(new Truck ());
8. Car c8 = (Vehicle)(new Truck ());
```

(8 נקודות)

ב. רוצים לכתוב פעולה אשר מקבלת מערך של כלי תחבורה (מערך הפניות לעצמים מטיפוס Vehicle) ומחזירה את המהירות הממוצעת של כל האופנועים (Motorcycle) ומכוניות המירוץ (RacingCar). באיזו מחלקה או מחלקות יש להוסיף את הפעולה double speed() המחזירה את המהירות?

הסבירו את תשובתכם, אין צורך לממש את הפעולה speed().

(4 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

ג. כתבו פעולה המקבלת מערך כלי תחבורה ומחזירה את האופנוע שמהירותו היא הגבוהה ביותר. אם במערך אין אף אופנוע, הפעולה תחזיר null.

כותרת הפעולה:

Java:

```
public static Motorcycle maxSpeed(Vehicle[] arr)
```

C#:

```
public static Motorcycle MaxSpeed(Vehicle[] arr)
```

(4 נקודות)

5.

נתונה הפעולה why הבאה:

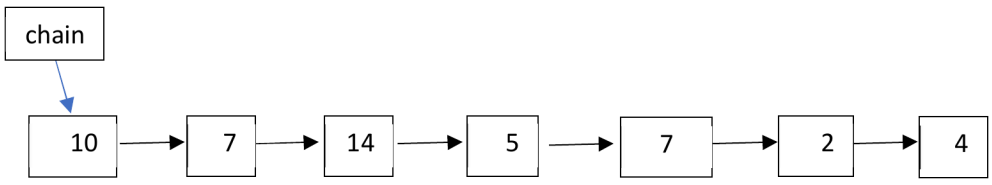
Java:

```
public static Node<Integer> why(Node<Integer> chain, boolean flag)
{
    int count = 0;
    Node<Integer> p = chain;
      while(p != null)
    {
      int x = p.getValue();
       boolean f = (x % 2 ==0);
      if( f == flag)
         count ++;
      p = p.getNext();
    }
    return new Node<Integer>(count);
 }
```

C#:

```
public static Node<int> Why(Node<int> chain, bool flag)
{
    int count = 0;
    Node<int> p = chain;
      while(p != null)
    {
      int x = p.GetValue();
      bool f = (x % 2 ==0);
      if( f == flag)
        count ++;
      p = p.GetNext();
    }
    return new Node<int>(count);
 }
```

נתונה שרשרת חוליות הבאה:



א. מה תחזיר הפעולה why(chain, false)?

יש להראות מעקב אחרי ביצוע הפעולה why.

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

(6 נקודות)

ב. נתונה הפעולה what הבאה:

Java:

```
public static void what(Node<Integer> chain)
{
   Node<Integer> p = chain;
   Node<Integer> temp;
   while (p != null)
   {
      if(p.getValue()%2 ==0)
       temp = why(p.getNext(), false);
      else
       temp = why(p.getNext(), true);
      temp.setNext(p.getNext());
       p.setNext(temp);
      p = temp.getNext();
   }
}
```

C#:

```
public static void What(Node<int> chain)
{
   Node<int> p = chain;
   Node<int> temp;
   while (p != null)
   {
      if(p.GetValue()%2 ==0)
        temp = Why(p.GetNext(), false);
      else
        temp = Why(p.GetNext(), true);
      temp.SetNext(p.GetNext());
      p.SetNext(temp);
      p = temp.GetNext();
   }
}
```

עקבו אחרי ביצוע זימון הפעולה what(chain) ורשמו איך תראה השרשרת אחרי זימון הפעולה.

יש להראות מעקב אחרי ביצוע הפעולה what, אין צורך במעקב אחרי why.

(6 נקודות)

ג. מה מבצעות הפעולות what(chain) ו- why(chain,flag) באופן כללי?

(4 נקודות)

6. א. כתבו פעולה המקבלת מחסנית של מספרים שלמים st ומספר נוסף num. הפעולה תמחק ממחסנית st את כל המופעים של num.

כותרת הפעולה:

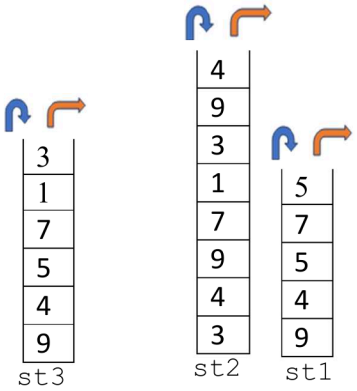
Java:

```
public static void remove(Stack<Integer> st, int num)
```

C#:

```
public static void Remove(Stack<int> st, int num)
```

(7 נקודות)



ב. כתבו פעולה המקבלת 2 מחסניות של מספרים שלמים st1 ו-st2. הפעולה תחזיר מחסנית חדשה הכוללת כל ערך מ-2 המחסניות. כל ערך במחסנית החדשה צריך להופיע פעם אחת. סדר האיברים במחסנית החדשה לא חשוב.

לדוגמה: עבור 2 מחסניות הבאות st1 ו-st2 הפעולה תחזיר מחסנית st3.

כותרת הפעולה:

Java:

```
public static Stack<Integer> commonValues(Stack<Integer> s1,
Stack<Integer> s2)
```

C#:

```
public static Stack<int> CommonValues(Stack<int> s1,
                    Stack<int> s2)
```

(7 נקודות)

ג. מהן הסיבוכיות של הפעולות שכתבתם בסעיפים א'-ב'? **הסבירו את תשובתכם.**

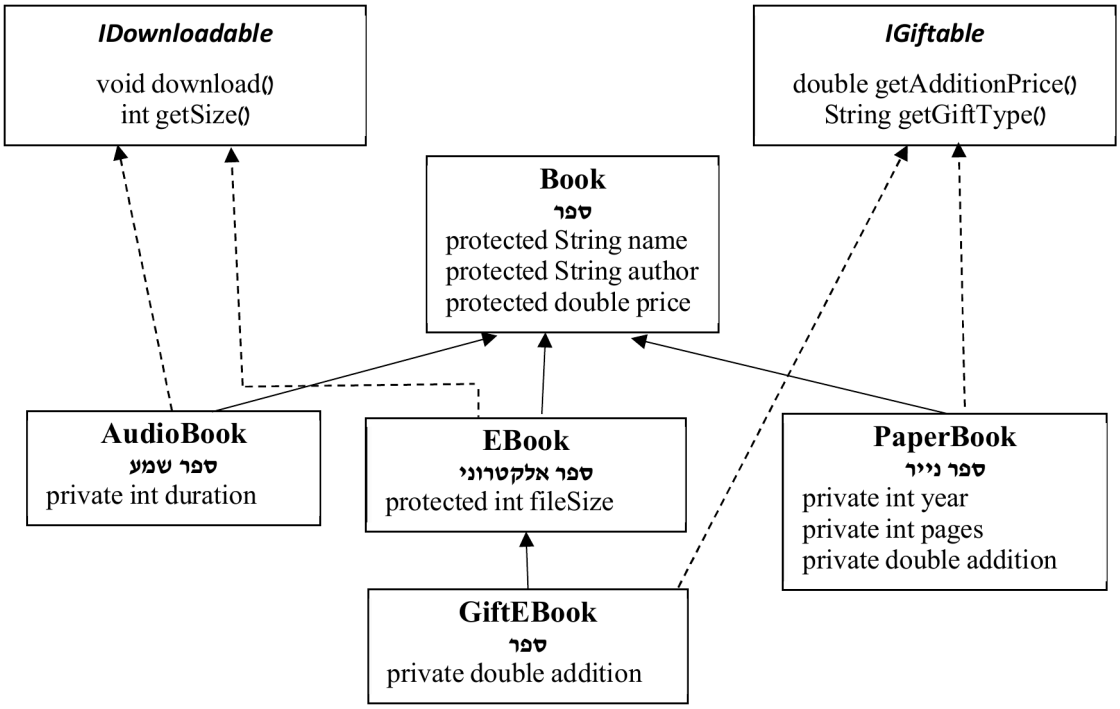
(4 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

7. החנות "ספר לכל" מוכרת סוגים שונים של ספרים: ספרים מודפסים, ספרי שמע, ספרים אלקטרוניים. חלק מהמוצרים ניתן לרכוש באריזת מתנה (בעבור תשלום נוסף).

לשם קידום מכירות ספרים, חנות "ספר לכל" פיתחה מערכת לניהול מלאי הספרים הכוללת את המחלקות הבאות Book, AudioBook, EBook, PaperBook, GiftEBook ו-2 ממשקים IGiftable ו-IDownloadable.

לפניכם תרשים UML המתאר קשרים בין המחלקות והממשקים:



א. כתבו עבור כל מחלקה, את כותרת המחלקה ואת הכותרות של כל הפעולות ש**חייבות** להיות במחלקה.

(5 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

ב. המחיר של כל ספר תלוי בסוג הספר. לשם כך הוחלט להגדיר 3 קבועים המציינים מקדמי מחיר עבור ספרים אלקטרוניים, ספרי שמע וספרים מודפסים:

Java:

```
final double EBOOK_FACTOR = 0.8;
final double AUDIOBOOK_FACTOR = 1.2;
final double PAPERBOOK_FACTOR = 1.0;
```

C#:

```
const double EBOOK_FACTOR = 0.8;
const double AUDIOBOOK_FACTOR = 1.2;
const double PAPERBOOK_FACTOR = 1.0;
```

באיזו מחלקה או באילו מחלקות יש להוסיף את הקבועים הנ"ל? הסבירו את תשובתכם.

(2 נקודות)

ג. כתבו פעולה או פעולות המחזירה את מחיר הספר (ללא אריזת מתנה), עבור כל סוג ספר, יש לציין באיזו מחלקה צריך להוסיף את הפעולה או את הפעולות.

(4 נקודות)

ד. כתבו פעולה חיצונית המקבלת מערך ספרים המתאר את מלאי החנות וסכום כסף שלקוח רוצה להשקיע. הפעולה תדפיס את פרטי הספרים שמחירם אינו עולה על הסכום שהלקוח רוצה להשקיע.

(3 נקודות)

ה. כתבו פעולה חיצונית המקבלת מערך ספרים המתאר את מלאי החנות. הפעולה תדפיס את מספר הספרים שאפשר לרכוש אותם באריזת מתנה.

(3 נקודות)

8. הגדרה: עץ "**שרשרת עולה**" הוא עץ בינארי של מספרים שלמים העונה על הכלל הבא:

- לכל צומת פרט לעלים יש בדיוק בן אחד.
- ערכים של כל הצמתים הם מספרים חיוביים זוגיים.
- הערך של צומת הבן גדול מהערך של צומת האב.

א. ציירו עץ "שרשרת עולה" הכולל לפחות 6 צמתים.

(4 נקודות)

ב. כתבו פעולה המקבלת הפנייה לשורש של עץ בינרי של מספרים שלמים. הפעולה תבדוק אם העץ הוא "שרשרת עולה". אם כן - הפעולה תחזיר ערך true, ולא - הפעולה תחזיר ערך false.

(14 נקודות)

9. הערה כללית לשאלה:

- אין להשתמש במבני נתונים נוספים.
- אפשר להניח שקיימת פעולה sizeQ המחזירה את מספר האיברים בתור. כותרת הפעולה:

Java:

```
public static int sizeQ(Queue<Integer> q)
```

C#:

```
public static int SizeQ(Queue<int> q)
```

א. כתבו פעולה המקבלת תור של מספרים שלמים וחיוביים q ומספר שלם חיובי pos. הפעולה תחזיר איבר אשר נמצא במקום pos בתור **בלי להוציא אותו מהתור**. אם אין מקום pos בתור, הפעולה תחזיר ערך (-1).

הערה: האיבר הנמצא בראש התור הוא איבר במקום 1, איבר אחריו - במקום 2 וכו'.

כותרת הפעולה:

Java:

```
public static int valueAt(Queue<Integer> q, int pos)
```

C#:

```
public static int ValueAt(Queue<int> q, int pos)
```

(7 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

ב. כתבו פעולה המקבלת 2 תורים של מספרים שלמים q1 ו - q2. הפעולה תיצור ותחזיר תור חדש לפי כלל הבא:

- קודם כול מכניסים איבר ראשון מהתור q1, ואחריו - איבר האחרון מהתור q2.
- איבר 2 מהתור q1, איבר לפני האחרון מהתור q2.
- וכך הלאה.

אפשר להניח שאורכם של 2 התורים שווה.

כותרת הפעולה:

Java:

```
public static Queue<Integer> merge (Queue<Integer>q1,
Queue<Integer>q2)
```

C#:

```
public static Queue<int> Merge (Queue<int>q1, Queue<int>q2)
```

(7 נקודות)

ג. מהן הסיבוכיות של הפעולות שכתבתם בסעיפים א' ו-ב'? **הסבירו את תשובתכם.**

(4 נקודות)

10. נתונות 2 המחלקות הבאות:

Java:

```
public class AA{
   public AA(){
    System.out.println("AA constructor");
   }
   public void one() {
      System.out.println("one in AA");
   }
   public void two() {
      one();
   }
}
public class BB extends AA{
   public BB(){
    System.out.println("BB constructor");
   }
   public void one(){
      System.out.println("one in BB");
   }
   public void two() {
      System.out.println("two in BB");
   }
   public void three() {
      super.two();
   }
}
```

C#:

```
public class AA{
   public AA(){
    Console.WriteLine("AA constructor");
   }
   public virtual void One() {
      Console.WriteLine("One in AA");
   }
   public virtual void Two() {
      One();
   }
}
public class BB : AA{
   public BB(){
```

```
    Console.WriteLine("BB constructor");
  }
  public override void One(){
    Console.WriteLine("One in BB");
  }
  public override void Two() {
    Console.WriteLine("Two in BB");
  }
  public void Three() {
    base.Two();
  }
}
```

א. נתון קטע הקוד הבא (מספרי שורות לא חלק של הקוד)

Java:

```
(1)  AA a = new AA();
(2)  BB b = new BB();
(3)  AA ab = new BB();
(4)  a.one();
(5)  ab.one();
(6)  b.one();
(7)  a.two();
(8)  ab.two();
(9)  b.two();
(10)  ((BB)(ab)).three();
(11)  b.three();
```

C#:

```
(1) AA a = new AA();
(2) BB b = new BB();
(3) AA ab = new BB();
(4) a.One();
(5) ab.One();
(6) b.One();
(7) a.Two();
(8) ab.Two();
(9) b.Two();
(10) ((BB)(ab)).Three();
(11) b.Three();
```

(1) עקבו אחרי ביצוע קטע הקוד ורשמו מה יהיה הפלט

(2) האם אפשר להוסיף לקטע הקוד את 2 הפקודות הבאות? **הסבירו את תשובתכם**

Java:

```
a.three();
((BB)a).three();
```

C#:

```
a.Three();
((BB)a).Three();
```

(14 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

ב. כתבו פעולה המקבלת מערך הפניות לעצמים מסוג Object. הפעולה תדפיס כמה עצמים הם מסוג BB, כמה עצמים הם מסוג AA ולא סוג BB וכמה עצמים הם לא מסוג AA.

(4 נקודות)

11. ארגון סטודנטים מתנדבים "עזרה לגיל זהב" מחזיק מאגר מידע של סטודנטים הרוצים להתנדב ולעזור לקשישים, בצורה ממוחשבת . עבור כל סטודנט נשמרים הנתונים הבאים:

- מספר תעודת הזהות (מחרוזת)
- שם (מחרוזת)
- יישוב מגורים (מחרוזת)
- אם יש לו רכב (ערך בוליאני, true - כן, false - לא)
- שפת האם (מחרוזת)
- שפה נוספת שסטודנט שולט בה ברמה גבוהה (מחרוזת)

א. כתבו את כותרת המחלקה ואת התכונות עבור כל אחת מהמחלקות הבאות:

- Student המתארת סטודנט.
- StudentsData המתארת מידע על כל הסטודנטים.

אפשר להניח שבמחלקה Student קיימים בנאים, פעולות get ו-set לכל תכונה.

לייצוג אוסף אפשר להשתמש במחלקות Stack, Queue, Node. אפשר להוסיף תכונות נוספות. חובה לתעד את התכונות.

(2 נקודות)

ב. בסיום שנת הלימודים יש לעדכן את המערכת ולהוציא מהמאגר (אוסף הנתונים) את הסטודנטים שסיימו ללמוד. כתבו פעולה במחלקה StudentsData המקבלת מספר תעודת זהות של מתנדב ומוציאה אותו מהמאגר. כותרת הפעולה:

Java:

```
public void eraseStudent(String id)
```

C#:

```
public void EraseStudent(string id)
```

(3 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

ג. מדי פעם מתקבלת בקשה לעזרה מאזרח וותיק אשר מדבר בשפה מסוימת והמתגורר ביישוב מסוים. כתבו במחלקה StudentsData את הפעולה:

Java:

```
public void print(String city, String lang)
```

C#:

```
public void Print(string city, string lang)
```

הפעולה תדפיס את שמות המתנדבים אשר מדברים בשפה lang ומתגוררים ביישוב city.

אם אין אף מתנדב שגר ביישוב city ומדבר בשפה lang, הפעולה תדפיס את דוברי השפה lang שהם בעלי רכב.

(4 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

ד. הנהלת הארגון רוצה לארגן כנס למתנדבים. הוחלט שהכנס יתקיים בעיר שמתגוררים בה הכי הרבה מתנדבים חברי הארגון.

(1) כתבו פעולה במחלקה StudentsData המקבלת שם של עיר ומחזירה את מספר המתנדבים המתגוררים בה. כותרת הפעולה:

Java:

```
public int numStudents(String city)
```

C#:

```
public int NumStudents(string city)
```

(2) כתבו פעולה במחלקה StudentsData המקבלת מערך של שמות ערים (מערך מסוג מחרוזת) ומחזירה את שם העיר שגרים בה הכי הרבה מתנדבים. כותרת הפעולה:

Java:

```
public String cityName(String[] cities)
```

C#:

```
public string CityName(string[] cities)
```

אפשר להניח שיש רק עיר אחת כזאת.

(7 נקודות)

12.

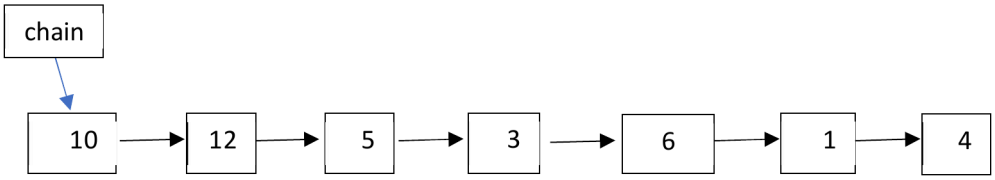
נתונה פעולה רקורסיבית הבאה:

Java:

```
public static void one(Node<Integer> n){
  if(n!=null)
  {
    if(n.getValue()% 2==0)
    {
      n.setNext(new Node<Integer> (0,n.getNext()));
      n = n.getNext();
    }
    one(n.getNext());
  }
}
```

C#:

```
public static void One(Node<int> n){
  if(n!=null)
  {
    if(n.GetValue()% 2==0)
    {
      n.SetNext(new Node<int> (0,n.GetNext()));
      n = n.GetNext();
    }
    One(n.GetNext());
  }
}
```



א. עקבו אחרי זימון הפעולה one(chain) ורשמו איך תיראה השרשרת אחרי ביצוע הפעולה.

(4 נקודות)

ב. מה מבצעת הפעולה one באופן כללי?

(2 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

ג. נתונה פעולה רקורסיבית two הבאה:

Java:

```
public static void two(Node<Integer> n){
  if(n.getNext()!=null)
  {
    if(n.getNext().getValue()%2 == 0)
     n.setNext(n.getNext().getNext());
    else
     n=n.getNext();
    two(n);
  }
}
```

C#:

```
public static void Two(Node<int> n){
  if(n.GetNext()!=null)
  {
    if(n.GetNext().GetValue()%2 == 0)
     n.SetNext(n.GetNext().GetNext());
    else
     n=n.GetNext();
    Two(n);
  }
}
```

עקבו אחרי זימון הפעולה two(chain) ורשמו איך תיראה השרשרת אחרי ביצוע הפעולה.

(4 נקודות)

ד. מה מבצעת הפעולה two באופן כללי?

(2 נקודות)

נכתב ונבדק על ידי צוותי GOOL בגרויות ו-GuyAI · www.guyai.co.il

ה. נתונה הפעולה three המשתמשת בפעולה two.

Java:

```
public static Node<Integer> three(Node<Integer>n){
  if(n == null) return null;
  if(n.getValue()%2 == 0)
   return three(n.getNext());
  else
  {
    two(n);
    return n;
  }
}
```

C#:

```
public static Node<int> Three(Node<int>n){
  if(n == null) return null;
  if(n.GetValue()%2 == 0)
   return Three(n.GetNext());
  else
  {
    Two(n);
    return n;
  }
}
```

מה יהיה התוכן של הרשימה chain אחרי ביצוע הפקודה הבאה.

Java:

```
chain = three(chain);
```

C#:

```
chain = Three(chain);
```

מה מבצעת הפעולה three באופן כללי?

(4 נקודות)

כל מבחני מה"ט עם פתרונות מלאים: www.guyai.co.il/mahat