GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: אביב 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 שני 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: יש לענות על **שלוש** שאלות בלבד. ערך כל שאלה 16 נקודות. **חלק ב' – 36 נקודות** שאלות 6-10: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה 18 נקודות. **חלק ג' – 16 נקודות** שאלות 11-12: יש לענות על **אחת** בלבד. ערך שאלה 16 נקודות. נקודה אחת תינתן על הערכה. **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!** לשאלון מצורפים 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 וחיוביים נקראת "מחסנית אחידה" אם כל הערכים שנמצאים בה הם מספרים זוגיים או כל הערכים שנמצאים בה הם מספרים אי-זוגיים.

כתבו פעולה המקבלת מחסנית של מספרים שלמים ובודקת אם היא "מחסנית אחידה".

כותרת הפעולה:

Java:

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

C#:

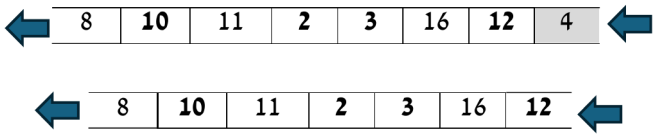
```
public static bool IsUniform(Stack<int> st)
```

(12 נקודות)

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

(4 נקודות)

2.



א. כתבו פעולה המקבלת תור של מספרים שלמים. הפעולה תחזיר את האיבר הנמצא בסוף התור ותוציא אותו מהתור. כל שאר איברי התור יישארו ללא שינוי.

לדוגמה:

עבור התור q הבא הפעולה תחזיר ערך 4

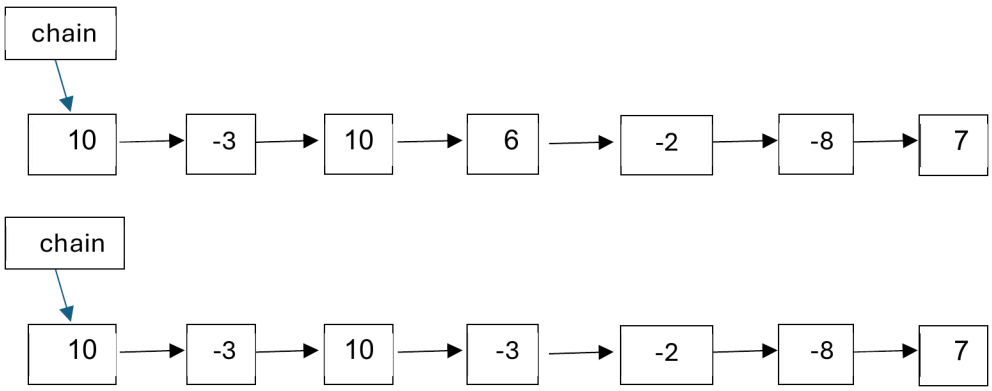
התור q אחרי זימון הפעולה:

(12 נקודות)

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

(4 נקודות)

3.



א. כתבו פעולה המקבלת הפניה chain לחוליה הראשונה של שרשרת חוליות של מספרים שלמים.

הפעולה תבדוק אם **אין חזרות של מספרים שליליים** (כלומר, כל ערך שלילי מופיע פעם אחת בלבד).

אם כן - הפעולה תחזיר ערך true, ולא - ערך false.

כותרת הפעולה:

Java:

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

C#:

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

לדוגמה: עבור שרשרת הבאה הפעולה תחזיר ערך true

עבור שרשרת הבאה הפעולה תחזיר ערך false כי ערך 3- מופיע פעמיים

(12 נקודות)

ב. מהי סיבוכיות הפעולה uniqNeg שכתבתם בסעיף א'? **הסבירו את תשובתכם.**

(4 נקודות)

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

4. נתונות שתי המחלקות Child ו- Driver הבאות:

Java:

```
public class Child extends Parent {
 private int age;
 public Child( String fName, String lName, char gender,int age){
    super(fName, lName, gender);
    this.age = age;
 }

 public Child(){
  super("David","Cohen");
  this.age = 13;
  this.gender = 'm';
 }

 public int getBirthYear(){
   return Parent.year - this.age;
 }

 public void printDetails() {
   if(this.gender == 'f')System.out.print("Daughter ");
   else System.out.print("Son ");
   System.out.print(getName()+" was born at " + getBirthYear());
 }
}// end of class Child

public class Driver {
  public static void main(String[] args) {

       Parent.setYear(2020);

       Parent[] family = new Parent[5];
       family[0] = new Parent("Anna", "Blum");
       family[1] = new Parent("Moshe","Coehn",'m');
       family[2] = new Child();
       family[3] = new Child("Dorit","Levi",'f',24);
       family[4] = new Child("Eli","Coehn",'m',30);
       family[3].updateCity("Tel-Aviv");
       family[4].updateCity("Jerusalem");

       for(int i=0; i<family.length; i++)
       {
          family[i].printDetails();
          family[i].printCity();
       }
```


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

```
  }
}// end of class Driver
```

C#:

```
public class Child : Parent {
 private int age;
 public Child( String fName, String lName, char gender,int age):
          base(fName, lName, gender){
   this.age = age;
 }

 public Child() : base("David","Cohen"){
  this.age = 13;
  this.gender = 'm';
 }

 public int GetBirthYear(){
   return Parent.year - this.age;
 }

 public override void PrintDetails() {
   if(this.gender == 'f')Console.Write("Daughter ");
   else Console.Write ("Son ");
   Console.Write (GetName()+" was born at " + GetBirthYear());
 }
}// end of class Child

public class Driver {
  public static void Main(string[] args) {

      Parent.SetYear(2020);

      Parent[] family = new Parent[5];
      family[0] = new Parent("Anna", "Blum");
      family[1] = new Parent("Moshe","Coehn",'m');
      family[2] = new Child();
      family[3] = new Child("Dorit","Levi",'f',24);
      family[4] = new Child("Eli","Coehn",'m',30);
      family[3].UpdateCity("Tel-Aviv");
      family[4].UpdateCity("Jerusalem");

      for(int i=0; i<family.Length; i++)
      {
        family[i].PrintDetails();
```

```
            family[i].PrintCity();
        }
    }
}// end of class Driver
```

עקב ביצוע הפעולה הראשית של המחלקה Driver התקבל פלט הבא:

```
Mother Anna Blum lives in Beer-Sheva
Father Moshe Coehn lives in Beer-Sheva
Son David Cohen was born at 2007 lives in Beer-Sheva
Daughter Dorit Levi was born at 1996 lives in Tel-Aviv
Son Eli Coehn was born at 1990 lives in Jerusalem
```

כתבו את המחלקה המורישה Parent (מחלקת העל, super-class) של המחלקה Child.
מחלקת העל צריכה להכיל את כל התכונות והפעולות המתחייבות ממחלקת Child.

יש לממש את הפעולות של מחלקת העל בהתאם לתוצאות הפלט שהתקבל בעקבות ביצוע הפעולה הראשית.

אין לשנות את מחלקת Child.

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

5. נתונה הפעולה הבאה:

Java:

```
public static void what(Stack<Integer> s)
{
   Stack<Integer> temp1 = new Stack<Integer>();
   Stack<Integer> temp2 = new Stack<Integer>();
   while(!s.isEmpty())
   {
      int x = s.pop();
      boolean found = false;
      while(!s.isEmpty())
      {
         int y =s.pop();
         if(x < y)
            found = true;
         temp1.push(y);
      }
      while(!temp1.isEmpty())
         s.push(temp1.pop());

      if(!found)
         temp2.push(x);
   }
   while(!temp2.isEmpty())
      s.push(temp2.pop());
 }
}
```

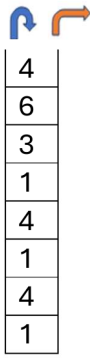
C#:

```
public static void What(Stack<int> s)
{
   Stack<int> temp1 = new Stack<int>();
   Stack<int> temp2 = new Stack<int>();
   while(!s.IsEmpty())
   {
      int x = s.Pop();
      bool found = false;
      while(!s.IsEmpty())
      {
         int y =s.Pop();
         if(x < y)
            found = true;
```


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

```
        temp1.Push(y);
      }
      while(!temp1.IsEmpty())
        s.Push(temp1.Pop());

      if(!found)
        temp2.Push(x);
    }
    while(!temp2.IsEmpty())
      s.Push(temp2.Pop());
  }
}
```



א. נתונה מחסנית s

מה יהיה התוכן של המחסנית s אחרי זימון הפעולה what(s) עבור המחסנית s

יש להראות מעקב אחרי ביצוע הפעולה what.

(10 נקודות)

ב. תנו דוגמה למחסנית s המכילה לפחות חמישה ערכים כך שתוכן המחסנית לא ישתנה אחרי זימון הפעולה what(s)

(3 נקודות)

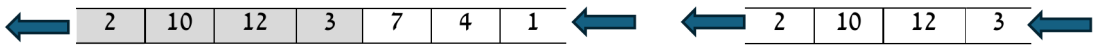
ג. מה מבצעת הפעולה what(s) עבור מחסנית של מספרים שלמים s?

(3 נקודות)

6. הערה כללית לשאלה:

- אין להשתמש במבני נתונים נוספים פרט לתור או תורים נוספים.
- אפשר להניח שקיימת פעולה `public static int size(Queue<Integer> q)` (ב-C#: `public static int Size(Queue<int> q)`) המחזירה מספר איברים בתור q.

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



א. הגדרה: תור q1 נקרא "תור התחלתי" של תור q2 אם כל האיברים של q1 נמצאים בתחילת התור q2 באותו הסדר.

לדוגמה:

התור q1

הוא תור התחלתי של התור q2

כתבו פעולה המקבלת שני תורים של מספרים שלמים q1 ו- q2. הפעולה תבדוק אם q1 הוא תור התחלתי של q2. אם כן - הפעולה תחזיר ערך true, ולא - הפעולה תחזיר ערך false.

כותרת הפעולה:

Java:

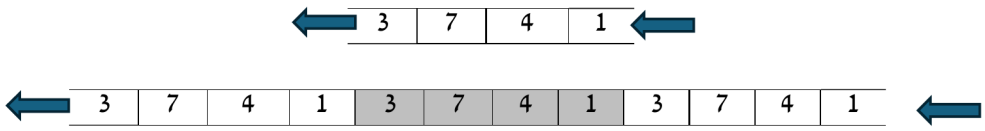
```
public static boolean startWith(Queue<Integer> q1,
                                Queue<Integer> q2)
```

C#:

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

(7 נקודות)

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



ב. **הגדרה**: תור q2 נקרא "שכפול" של תור q1 אם כל האיברים של q1 נמצאים בתור q2 באותו הסדר כמה פעמים (התור q2 מורכב ממספר של תורים q1).

לדוגמה:

עבור התור q1

התור q2 הוא "שכפול" של התור q1

כתבו פעולה המקבלת שני תורים של מספרים שלמים q1 ו- q2. הפעולה תבדוק אם q2 הוא "שכפול" של q1. אם כן - הפעולה תחזיר ערך true, ולא - הפעולה תחזיר ערך false.

כותרת הפעולה:

Java:

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

C#:

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

(7 נקודות)

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

(4 נקודות)

7. סדרת מספרים נקראת "עולה - יורדת" אם עד איבר מסוים המספרים הולכים וגדלים ומאיבר זה ואילך, הם הולכים וקטנים.

לדוגמה:

הסדרה הבאה היא סדרה "עולה - יורדת"

2, 5, 13, 19, **21**, 18, 13, 6, 3

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

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

כותרת הפעולה:

Java:

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

C#:

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

(14 נקודות)

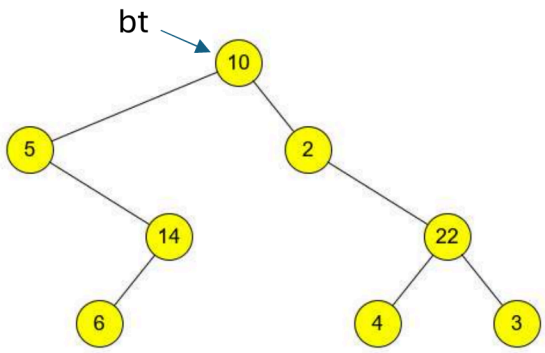
ב. מהי סיבוכיות של הפעולות שכתבתם בסעיף א'? **הסבירו את תשובתכם.**

(4 נקודות)

8. כתבו פעולה המקבלת הפנייה לשורש של עץ בינרי של מספרים שלמים. הפעולה תדפיס את כל הצמתים שהערך שלהם **קטן מהממוצע** של כל הצמתים.

לדוגמה:

עבור העץ הבא



הפעולה תדפיס 5, 2, 6, 4, 3

הערה: סדר ההדפסה לא חשוב.

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

9. נתונות שלוש המחלקות הבאות:

Java:

```
public class One {
   protected int n1;
   public One ()
   {
      this.n1 = 1;
   }
   public One (int val)
   {
      this.n1 = val;
   }
   public void update(int k)
   {
      this.n1 = this.n1 ˆ k;
   }
   public String toString()
   {
      return "One: n1=" + this.n1;
   }
}

public class Two extends One {
   protected int n2;
   public Two(int val)
   {
      this.n2 = val;
   }
   public Two (int val1,int val2)
   {
      super(val1);
      this.n2 = val2;
   }
   public void update(int k)
   {
      super.update(k);
      this.n2 = this.n2 ˆ k;
   }

   public String toString()
   {
      return super.toString() + " Two: n2=" + this.n2;
   }
}
```


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

```
public class Test {
   public   static void main(String[] args){
      One[] arr = new One [6];
      arr [0] = new One (2);
      arr [1] = new Two(4);
      arr [2] = new Two(3, 6);
      arr [3] = arr [0];
      arr [4] = new Two(4);
      arr [5] = arr[1];

      for (int i = 0; i < arr.length; i++)
         System.out.println(arr[i]);

(˙)    for(int i=0;i< arr.length;i++)
         arr [i].update(i+3);

(˙˙)   for (int i = 0; i < arr.length; i++)
         System.out.println(arr[i]);
   }
}
```

C#:

```
public class One {
   protected int n1;
   public One ()
   {
      this.n1 = 1;
   }
   public One (int val)
   {
      this.n1 = val;
   }
   public virtual void Update(int k)
   {
      this.n1 = this.n1 ˙ k;
   }
   public override string ToString()
   {
      return "One: n1=" + this.n1;
   }
}

public class Two : One {
   protected int n2;
```

```
  public Two(int val):base()
  {
    this.n2 = val;
  }
  public Two (int val1,int val2):base(val1)
  {
    this.n2 = val2;
  }
  public override void Update(int k)
  {
    base.Update(k);
    this.n2 = this.n2 * k;
  }

  public override string ToString()
  {
    return base.ToString() + " Two: n2=" + this.n2;
  }
}

public class Test {
  public   static void Main(string[] args){
    One[] arr = new One [6];
    arr [0] = new One (2);
    arr [1] = new Two(4);
    arr [2] = new Two(3, 6);
    arr [3] = arr [0];
    arr [4] = new Two(4);
    arr [5] = arr[1];

    for (int i = 0; i < arr.Length; i++)
      Console.WriteLine(arr[i]);

(*)   for(int i=0;i< arr.Length;i++)
      arr [i].Update(i+3);

(**)  for (int i = 0; i < arr.Length; i++)
      Console.WriteLine (arr[i]);
  }
}
```

א. עקבו אחרי ביצוע פעולה הראשית (main) של המחלקה Test ורשמו מה יהיה הפלט של הפעולה הראשית. **חובה להציג עבור כל עצם שנוצר בפעולה את ערכי התכונות שלו.**
(12 נקודות)

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

ב. שתי הלולאות (*) ו-() הוחלפו ללולאה הבא:

Java:

```
for(int i=0;i< arr.length;i++) {
  arr [i].update(i+3);
  System.out.println(arr[i]);
}
```

C#:

```
for(int i=0;i< arr.Length;i++) {
  arr [i].Update(i+3);
  Console.WriteLine(arr[i]);
}
```

האם פלט של הפעולה הראשית השתנה? **הסבירו את תשובתכם**

(6 נקודות)

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

10. לפניכם שתי הפעולות הרקורסיביות what ו-why. הפעולה what מקבלת הפניה לאחת מהחוליות בשרשרת חוליות של מספרים שלמים, והפעולה why מקבלת הפניה לחוליה ראשונה בשרשרת החוליות.

Java:

```
public static Node<Integer> what (Node<Integer> p1)
{
   if(p1.getNext() ==null)
      return p1;
   Node<Integer> p2 = what(p1.getNext());
   if(p1.getValue() < p2.getValue())
      return p1;
   return p2;
}


public static void why(Node<Integer> chain)
{
   if(chain.getNext()!=null)
   {
      Node<Integer>p = what(chain);
      int temp=p.getValue();
      p.setValue(chain.getValue());
      chain.setValue(temp);
      why(chain.getNext());
   }
}
```

C#:

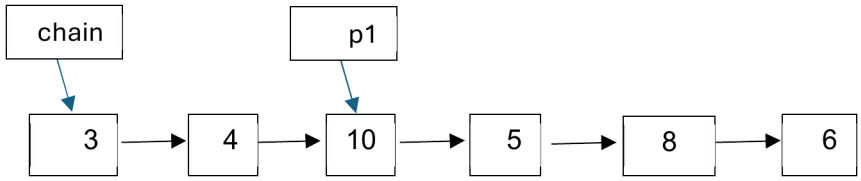
```
public static Node<int> What (Node<int> p1)
{
   if(p1.GetNext() ==null)
      return p1;
   Node<int> p2 = What(p1.GetNext());
   if(p1.GetValue() < p2.GetValue())
      return p1;
   return p2;
}


public static void Why(Node<int> chain)
{
   if(chain.GetNext()!=null)
```


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

```
    {
        Node<int>p = What(chain);
        int temp=p.GetValue();
        p.SetValue(chain.GetValue());
        chain.SetValue(temp);
        Why(chain.GetNext());
    }
}
```

נתונה שרשרת חוליות הבאה:



א. מה יחזיר זימון הפעולה what(p1)? **יש להראות מעקב אחר ביצוע הפעולה.**

(6 נקודות)

ב. מה מבצעת הפעולה what באופן כללי?

(3 נקודות)

ג. עקבו אחרי זימון הפעולה why(chain) ורשמו איך תיראה השרשרת אחרי זימון.

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

(6 נקודות)

ד. מה מבצע הפעולה why באופן כללי?

(3 נקודות)

11. במאגר הציונים של סטודנטים באוניברסיטה נשמרים עבור כל סטודנט הנתונים הבאים:
מספר תעודת הזהות ורשימת כל ציונים שלו. לכל ציון נשמר קוד מקצוע ושני ציונים (ציון של סמסטר א' וציון של סמסטר ב')

א. הגדירו מחלקה Grade המתארת ציון במקצוע. תכונות של המחלקה:

- קוד מקצוע.
- ציון סמסטר א'.
- ציון סמסטר ב'.

(2 נקודות)

ב. הגדירו מחלקה Student המתארת סטודנט באוניברסיטה. תכונות המחלקה:

- מספר תעודת הזהות של הסטודנט.
- רשימת ציונים (שרשרת חוליות מסוג Grade).

(2 נקודות)

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

ג. כתבו פעולה המקבלת מאגר (אוסף נתונים) של סטודנטים ומחזירה מספר תעודת הזהות של סטודנט בעל ממוצע (של כל הציונים בכל המקצועות שהוא למד) הגבוה ביותר.
אפשר להניח שקיים רק סטודנט אחד עם ממוצע הגבוה ביותר.
אפשר להשתמש לייצוג האוסף בכל מבנה הנתונים הנלמד (תור, מחסנית, שרשרת חוליות)
הערה: אפשר להניח שבמחלקות Grade ו-Student קיימות פעולות get לכל התכונות.
(12 נקודות)

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

12. נתונות שתי הפעולות what ו- where המקבלות הפניה לחוליה הראשונה של שרשרת חוליות:

Java:

```
public static Node<Integer> what(Node<Integer> chain, int x){
    while(chain!=null && chain.getValue()% x == 0){
      chain = chain.getNext();
    }
    if(chain == null)
        return null;

    Node<Integer> p = chain;
    Node<Integer> p1 = p.getNext();
    while(p1 != null){
      if(p1.getValue()% x == 0)
      {
        p.setNext(p1.getNext());
        p1.setNext(null);
        p1 = p.getNext();
      }
      else
      {
        p = p1;
        p1 = p1.getNext();
      }
    }
    return chain;
  }
public static void where(Node<Integer> chain){
    Node<Integer> p = chain;
    while(p != null) {
      p.setNext(what(p.getNext(),p.getValue()));
      p = p.getNext();
    }
  }
```

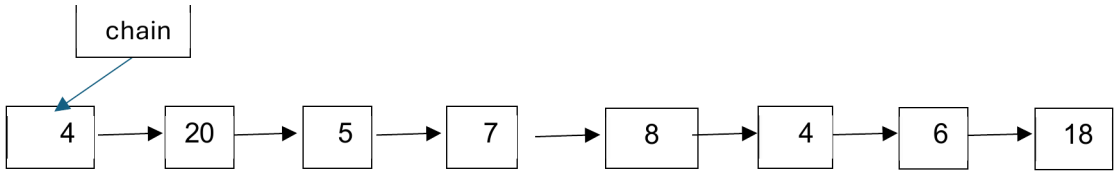
C#:

```
public static Node<int> What(Node<int> chain, int x){
    while(chain!=null && chain.GetValue()% x == 0){
      chain = chain.GetNext();
    }
    if(chain == null)
        return null;
```


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

```
    Node<int> p = chain;
    Node<int> p1 = p.GetNext();
    while(p1 != null){
      if(p1.GetValue()% x == 0)
      {
        p.SetNext(p1.GetNext());
        p1.SetNext(null);
        p1 = p.GetNext();
      }
      else
      {
        p = p1;
        p1 = p1.GetNext();
      }
    }
    return chain;
  }
public static void Where(Node<int> chain){
    Node<int> p = chain;
    while(p != null) {
      p.SetNext(What(p.GetNext(),p.GetValue()));
      p = p.GetNext();
    }
  }
```

נתונה שרשרת חוליות הבא:



א. עקבו אחרי זימון הפעולה what(chain, 4) ורשמו מה תחזיר הפעולה.

(5 נקודות)

ב. מה מבצעת הפעולה what באופן כללי?

(2 נקודות)

ג. עקבו אחרי זימון הפעולה where(chain) ורשמו איך תיראה השרשרת אחרי ביצוע. אין צורך במעקב אחרי זימון הפעולה what.

(5 נקודות)

ד. האם אפשר לשנות סדר חוליות בשרשרת chain כך ששרשרת לא תשתנה אחרי זימון הפעולה where? אם כן - רשמו את השרשרת אחרי שינוי, אם לא - הסבירו למה אי אפשר לבצע שינוי כזה.

(2 נקודות)

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

ה. מה מבצעת הפעולה where באופן כללי?

(2 נקודות)

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