GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: קיץ 2025 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- C# (עמוד 2) מבחן ב- JAVA (עמוד 14) בכל מבחן עשר שאלות. **חלק א' – 32 נקודות** שאלות 1–3: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה 16 נקודות. **חלק ב' – 32 נקודות** שאלות 4–6: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה 16 נקודות. **חלק ג' – 36 נקודות** שאלות 7–9: יש לענות על **שתי** שאלות בלבד. ערך השאלה 18 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 המבחן, ולוודא שהן מובנות. 2. את התשובות יש לכתוב בצורה מסודרת, בכתב יד ברור ונקי (גם בכך תלויה הערכת המבחן). 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!**

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

**בהצלחה!**

# השאלות

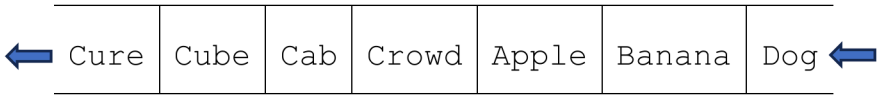
## 1. הגדרה:

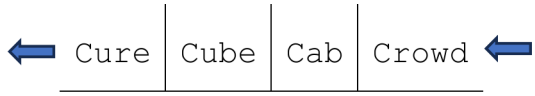
תור מחרוזות "תקין" הוא תור שבו כל המחרוזות המתחילות באותה האות נמצאות רק בתחילת התור.

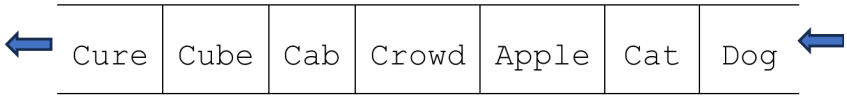
לדוגמה, התור שלפניכם הוא "תור תקין" (איור 1 לשאלה).

גם התור הזה הוא "תור תקין" (איור 2 לשאלה).

אבל התור הזה אינו "תור תקין" (איור 3 לשאלה).




**א.** כתבו פעולה המקבלת תור של מחרוזות "**תקין**" q ומחרוזת st. הפעולה תוסיף את המחרוזת לתור כך שהוא יישאר תקין.

כותרת הפעולה:

```
public static void addToProperQueue(Queue<String> q, String s)
```

```
public static void AddToProperQueue(Queue<string> q, string s)
```

(3 נקודות)

**ב.** כתבו פעולה המקבלת תור של מחרוזות. הפעולה תבדוק אם תור הוא "תור תקין". אם כן - הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

כותרת הפעולה:

```
public static boolean isProperQueue(Queue<String> q)
```

```
public static bool IsProperQueue(Queue<string> q)
```

(5 נקודות)



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

**ג.** כתבו פעולה המקבלת תור של מחרוזות. הפעולה תבדוק אם התור הוא "תור תקין". אם כן - הפעולה לא תבצע דבר, ואם לא - הפעולה תשנה את סדר המחרוזות בתור כך שהוא יהיה "**תקין**".

כותרת הפעולה:

```
public static void fixIt(Queue<String> q)
```

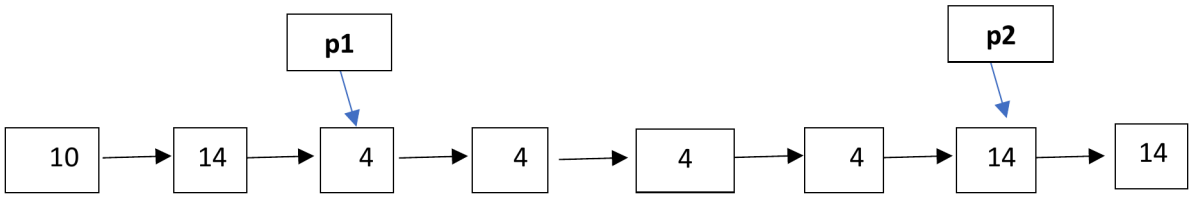
```
public static void FixIt(Queue<string> q)
```

(5 נקודות)

**ד.** מהי סיבוכיות הפעולות שכתבתם בסעיפים א'-ג'? **הסבירו את תשובתכם.**

(3 נקודות)

**2.** נגדיר "רצף K" כרצף של K חוליות של מספרים שלמים שהערך של כל אחת מהן הוא K.




**א.** כתבו פעולה המקבלת שתי הפניות p1 ו-p2 לחוליות שונות בשרשרת חוליות של מספרים שלמים. הפעולה תבדוק אם החוליות בין p1 (כולל) ל-p2 (לא כולל) מהוות את "רצף K". אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

לדוגמה, עבור השרשרת שלפניכם (איור לסעיף א) הפעולה תחזיר ערך true.

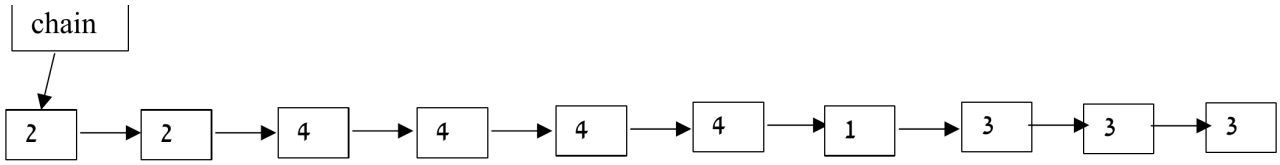
כותרת הפעולה:

```
public static boolean isSequenceK(Node<Integer> p1,
                                  Node<Integer> p2)
```

```
public static bool IsSequenceK(Node<int> p1, Node<int> p2)
```

(6 נקודות)


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




**ב.** שרשרת חוליות נקראת "שרשרת רצפים" אם היא מורכבת מכמה "רצפי K".
כתבו פעולה המקבלת הפניה לחוליה הראשונה של שרשרת חוליות של מספרים שלמים.
הפעולה תבדוק אם השרשרת היא "שרשרת רצפים". אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.
לדוגמה, השרשרת שלפניכם היא "שרשרת רצפים" (איור לסעיף ב).
כותרת הפעולה:

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

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

(8 נקודות)

**ג.** מהי סיבוכיות הפעולות שכתבתם בסעיפים א'-ב'? **הסבירו את תשובתכם.**
(2 נקודות)

**3.** בשאלה זו שלושה סעיפים. אין קשר בין הסעיפים.

**א.** נתונה פעולה ראשית במחלקה Test.

```
public class Test {
   public static void main(String[] args){
      S s1 = new R();
      P p1 = (P)(new R());
      S s2 = new P();
      P p2 = new Q();
//     (*****)
   }
}
```

```
public class Test {
   public static void Main(string[] args){
      S s1 = new R();
      P p1 = (P)(new R());
      S s2 = new P();
      P p2 = new Q();
//     (*****)
   }
}
```

(1) שרטטו את **עצי הירושה עבור שתי** אפשרויות היררכיה בין המחלקות כך שהפעולה הראשית תהיה תקינה והתוכנית תרוץ ללא שגיאות.

(2) רוצים להוסיף בסוף התוכנית את שורת הקוד שלפניכם:

```
Q q1 =(Q) p1;
```

האם התוכנית הראשית תהיה תקינה עבור שתי האפשרויות שציינתם? אם לא - מהו סוג השגיאה שתתרחש?

(4 נקודות)



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

**ב. לפניכם המחלקות First, Second ו-Program. במחלקות First ו-Second חסר מימוש בנאים ופעולות toString.**

```
public class First {
 private static int c = 0;
 protected int num;
 public First(int num){…}
 public String toString(){…}
}

public class Second extends First{
 private char ch;
 public Second(int num, char ch){…}
 public String toString(){…}
}
```

```
public class First {
 private static int c = 0;
 protected int num;
 public First(int num){…}
 public override string
       ToString(){…}
}

public class Second : First{
 private char ch;
 public Second(int num, char ch):
     base(num){…}
 public override string
     ToString(){…}
}
```

**השלימו את הפעולות במחלקות First ו-Second כך שאחרי הרצת הפעולה הראשית של המחלקה Program יתקבל הפלט הזה:**

```
public class Program{
 public static void main(String[] args) {
  First f1 = new First(10);
  System.out.println("f1 is "+f1);
  First f2 = new Second(3,'A');
  System.out.println("f2 is "+f2);
  First f3 = new Second(4,'Z');
  System.out.println("f3 is "+f3);
  First f4 = new First(7);
  System.out.println("f4 is "+f4);
```


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

```
 }
}
```

```
public class Program{
 public static void Main(string[] args) {
  First f1 = new First(10);
  Console.WriteLine("f1 is "+f1);
  First f2 = new Second(3,'A');
  Console.WriteLine("f2 is "+f2);
  First f3 = new Second(4,'Z');
  Console.WriteLine("f3 is "+f3);
  First f4 = new First(7);
  Console.WriteLine("f4 is "+f4);
 }
}
```

## הפלט שהתקבל אחרי הרצת התוכנית:

```
f1 is 1-10
f2 is 2-6-A
f3 is 3-12-Z
f4 is 4-28
```

(6 נקודות)



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

**ג.** לפניכם קוד המחלקה Two. המחלקה יורשת מהמחלקה One.

```
public class Two extends One
{
  private int z;
  public Two()
  {
    this.z = 11;
  }

  public Two(String s1, int z)
  {
    super(s1);
    this.z = z;
  }

  public double p1(int a)
  {
    return a * super.p1(a);
  }

  public int p2()
  {
    return One.w + this.m();
  }

  public void print()
  {
    One.d();
    System.out.println(s+"SON");
  }
}
```

```
public class Two : One
{
  private int z;
  public Two()
  {
    this.z = 11;
  }

  public Two(string s1, int z): base(s1)
  {
    this.z = z;
  }
```



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

```
  public override double P1(int a)
  {
    return a * base.P1(a);
  }

  public int P2()
  {
    return One.w + this.M();
  }

  public void Print()
  {
    One.D();
    Console.WriteLine(s+" SON");
  }
}
```

כתבו במחלקה One את התכונות וה**כותרות** של הפעולות הנדרשות.

יש לציין הרשאת הגישה (public, protected, private) לכל תכונה ולכל פעולה.

- **אין צורך לממש את הפעולות.**
- **אין לשנות את הקוד של המחלקה Two.**

(6 נקודות)



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

**4.** החברה "סוד זה הצלחה" החליטה שכל הישיבות והכנסים שלה יתקיימו באופן וירטואלי דרך מערכת מפגשים וירטואליים Boom שפיתחו מתכנתים שלה. במפגשי Boom ניתן לחלק משתתפים בין "חדרים וירטואליים". לשם ניהול המפגשים הווירטואליים פותחו מחלקות Meeting, VRoom, Participant.

- כל משתתף (Participant) מאופיין על ידי שם (name) ומספר היחידה שבה הוא עובד (dep, מספר שלם בין 1 ל-10).
- כל חדר (VRoom) מאופיין על ידי מספר חדר (num, מספר שלם וחיובי) ותור משתתפים (parts).
- למחלקה Meeting תכונה אחת - רשימת חדרים וירטואליים (rooms - שרשרת חוליות מסוג VRoom).

```
public class User {
 private String name;
 private int dep;
 public User(String n, int d){
  this.name = n;
  this.dep = d;
 }
} //אפשר להניח שהוגדרו הפעולות set/get

public class Meeting {
 private Node<VRoom> rooms;
 public Meeting() {
  this.rooms = null;
 }
}

public class Vroom {
 private int num;
 Queue<Participant> parts;
 public VRoom(int n){... }
}
```

```
public class User {
 private string name;
 private int dep;
 public User(string n, int d){
  this.name = n;
  this.dep = d;
 }
} //אפשר להניח שהוגדרו Set/Get

public class Meeting {
 private Node<VRoom> rooms;
```


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

```
 public Meeting() {
  this.rooms = null;
 }
}

public class Vroom {
 private int num;
 Queue<Participant> parts;
 public VRoom(int n){
  ...
 }
}
```

**א.** כתבו במחלקה VRoom פעולה בונה המקבלת מספר חדר ובונה חדר ריק ממשתתפים.

(2 נקודות)

**ב.** כתבו במחלקה VRoom פעולה add המקבלת משתתף ומוסיפה אותו לחדר.

(2 נקודות)

**ג.** לפני כל ישיבה, הנהלת החברה מחליטה על מספר המשתתפים שיהיו בחדר וירטואלי.
כתבו במחלקה Meeting פעולה המקבלת מספר שלם וחיובי number. הפעולה תקלוט את פרטי המשתתפים (שם ומספר יחידה). הפעולה תיצור חדר שמספרו אחד ואליו יצטרפו number משתתפים ראשונים, לאחריו חדר מספר שתיים ואליו יצטרפו number משתתפים הבאים וכך הלאה. הקלט יסתיים כאשר ייקלט 0 כמספר המחלקה.
כותרת הפעולה:

```
public void divide (int number)
```

```
public void Divide (int number)
```

(6 נקודות)

**ד.** כאשר מטרת הישיבה היא סיעור מוחות (brainstorming), חשוב שבכל חדר יהיו נציגים מכל עשר יחידות החברה.
כתבו במחלקה Meeting פעולה אשר מדפיסה עבור כל חדר את מספרי יחידות החברה שנציגיהן לא נמצאים בחדר.
כותרת הפעולה:

```
public void printMissingUnits()
```

```
public void PrintMissingUnits()
```

(6 נקודות)


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

**5. א.** כתבו פעולה **getLast** המקבלת הפניה לחוליה הראשונה של שרשרת של מספרים שלמים.
הפעולה תחזיר הפנייה לחוליה האחרונה של השרשרת.

(2 נקודות)

נתונה הפעולה **what** המקבלת שני מספרים שלמים $k < n$ . הפעולה משתמשת בפעולה getLast.

```
public static Node<Integer> what (int n, int k){
  if (n == k)
    return new Node<Integer>(n);
  else
  {
    if (k % 2 == 0)
      return new Node<Integer>(k, what(n, k+1));
    else
    {
      Node<Integer> p = new Node<Integer>(k);
      Node<Integer> chain = what (n, k+1);
      Node<Integer> last = getLast (chain);
      last.setNext (p);
      return chain;
    }
  }
}
```

```
public static Node<int> What (int n, int k){
  if (n == k)
    return new Node<int>(n);
  else
  {
    if (k % 2 == 0)
      return new Node<int>(k, What(n, k+1));
    else
    {
      Node<int> p = new Node<int>(k);
      Node<int> chain = What (n, k+1);
      Node<int> last = GetLast (chain);
      last.SetNext (p);
      return chain;
    }
  }
}
```

**ב.** עקבו אחרי זימון הפעולה what(10,5) ורשמו מה תחזיר הפעולה.

**יש להראות את תוכן השרשרת בחזרה מכל קריאה רקורסיבית.**

(6 נקודות)


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

**נתונה הפעולה secret:**

```
public static Node<Integer> secret (int n, int k) {
  if (n == k)
    return new Node<Integer>(n);
  else
  {
    if (n % 2 == 0)
      return new Node<Integer>(n, secret(n-1, k));
    else
    {
      Node<Integer> chain = secret (n-1, k);
      getLast (chain).setNext (new Node<Integer>(n));
      return chain;
    }
  }
}
```

```
public static Node<int> Secret (int n, int k) {
  if (n == k)
    return new Node<int>(n);
  else
  {
    if (n % 2 == 0)
      return new Node<int>(n, Secret(n-1, k));
    else
    {
      Node<int> chain = Secret (n-1, k);
      GetLast (chain).SetNext (new Node<int>(n));
      return chain;
    }
  }
}
```

**ג.** מה תהיה תוצאת הזימון secret(6, 2)?

(3 נקודות)

**ד.** האם קיימים שני מספרים שלמים וחיוביים $k < n$ כך שתוצאות הזימונים secret(n, k) ו-what(n, k) יהיו זהות? אם כן - תנו דוגמה לזוג מספרים שכזה, ואם לא - הסבירו למה הדבר אינו אפשרי.

(2 נקודות)

**ה.** כתבו את הפעולה what בצורה הלא רקורסיבית.

(3 נקודות)



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

**6.** **א.** כתבו פעולה sumLeaves שמקבלת עץ בינארי של מספרים שלמים ומחזירה את סכום הערכים של **כל העלים** בעץ.

(4 נקודות)

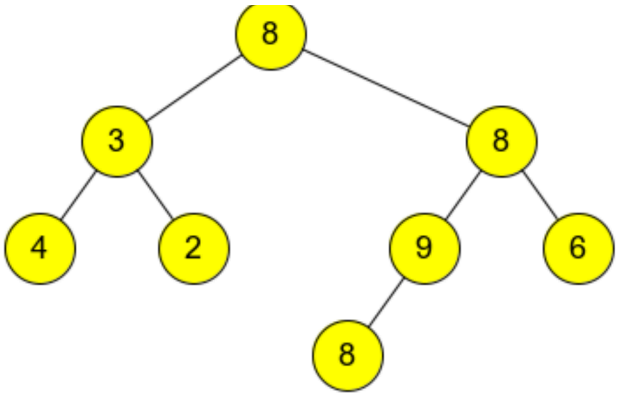
נתונה המחלקה NodeInfo שמכילה שתי תכונות: מספר שלם num, ומחרוזת info.

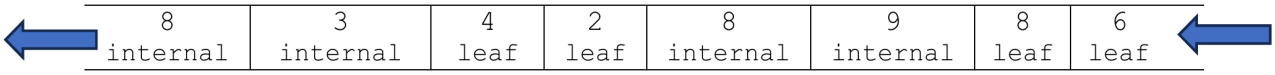
במחלקה הוגדרה פעולה בונה:

```
public NodeInfo(int num, String info)
```

```
public NodeInfo(int num, string info)
```

במחלקה הוגדרו פעולות get ו-set במחלקה NodeInfo.



| 8<br>internal | 3<br>internal | 4<br>leaf | 2<br>leaf | 8<br>internal | 9<br>internal | 8<br>leaf | 6<br>leaf |
|---|---|---|---|---|---|---|---|

**ב.** כתבו פעולה createQueue המקבלת עץ בינארי של מספרים שלמים ומחזירה תור מטיפוס NodeInfo שבו:

- ערך ה-num בכל איבר זהה לערך הצומת בעץ המתקבל כפרמטר.
- ערך ה-info מכיל את המחרוזת "*leaf*" אם הצומת הוא עלה, ואת המחרוזת "*internal*" אם הצומת הוא צומת פנימי.

**סדר האיברים בתור לא חשוב!**

**לדוגמה**, עבור העץ שלפניכם (איור 1 לסעיף ב) הפעולה תחזיר את התור הזה (איור 2 לסעיף ב).

כותרת הפעולה:

```
public static Queue<NodeInfo> createQueue (BinNode<Integer> bt)
```

```
public static Queue<NodeInfo> CreateQueue (BinNode<int> bt)
```

(6 נקודות)

**ג.** כתבו פעולה maxLeaveValue המקבלת עץ בינארי של מספרים שלמים ומחזירה את **הערך הגדול ביותר מבין העלים של העץ.**

(6 נקודות)



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

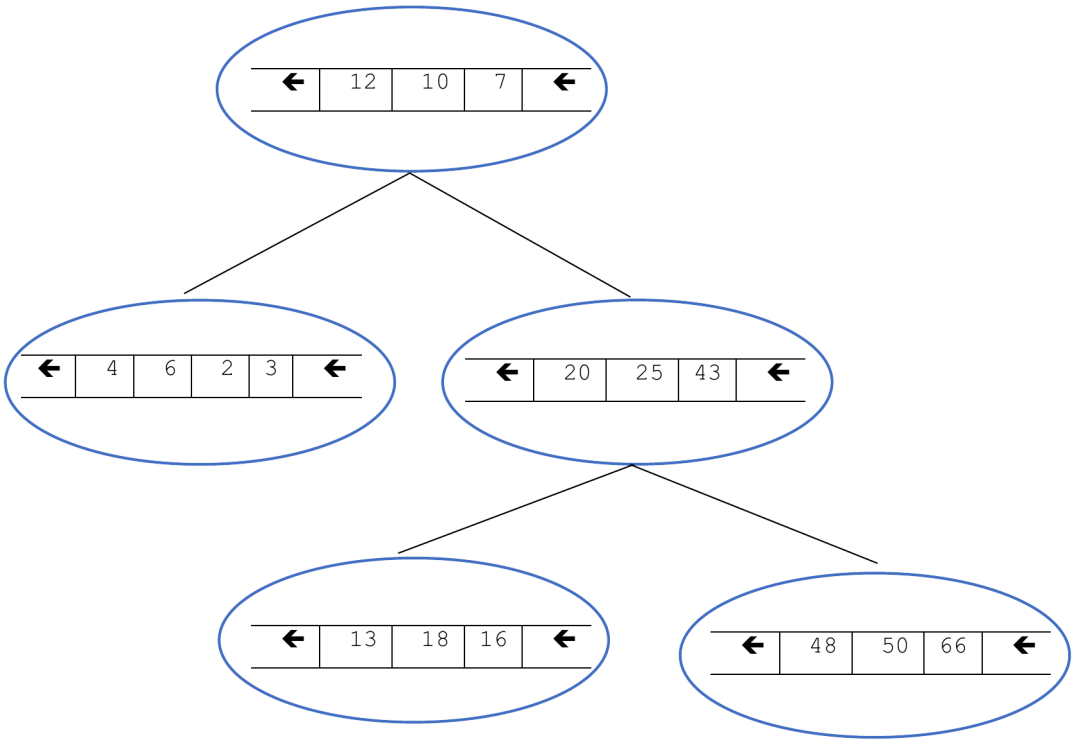
**7. א.** כתבו פעולה המקבלת שני תורים של מספרים שלמים q1 ו-q2. הפעולה תבדוק אם כל הערכים של תור q1 קטנים מכל הערכים של q2. אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

כותרת הפעולה:

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

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

(6 נקודות)




**ב.** נתון עץ שבו כל צומת מכיל תור של מספרים שלמים. עץ נקרא "עץ מעולה" אם הוא עונה על שני התנאים האלה:

- לכל צומת שאינו עלה יש שני בנים.
- כל הערכים הנמצאים בצומת האב גדולים מכל הערכים הנמצאים בבן השמאלי וקטנים מכל הערכים הנמצאים בבן הימני.

**לדוגמה**, העץ שלפניכם הוא "עץ מעולה" (איור לסעיף ב).

כתבו פעולה המקבלת עץ תורים של מספרים שלמים ובודקת אם הוא "עץ מעולה". אם כן - הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

(6 נקודות)

**ג.** נתונה טענה: ערך **הכי קטן** ב"**עץ מעולה**" נמצא בעלה **הכי שמאלי**. האם הטענה נכונה? **הסבירו את תשובתכם.**

(3 נקודות)


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

**ד.** מהי הסיבוכיות של הפעולות שכתבתם? **הסבירו את תשובתכם.**

(3 נקודות)


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

## 8. נתונות המחלקות האלה:

```
public abstract class A
{
 protected int x;

 public A() {
  this.x = 1;
 }
 public abstract int f(int v);
 public void f(A a) {
   x = a.x;
 }
 public String toString(){
  return "x =" + this.x;
 }
} //end of class A

public class B extends A {
 public B() {
  super();
 }
 public B(int val) {
  this.x = f(val);
 }
 public int f(int val) {
  return this.x + val;
 }
 public void f(B b) {
  this.x = this.x * b.x;
 }
 public String toString(){
 return "B: " +
        super.toString();
 }
} //end of class B

public class D extends B {
 public D(int val)  {
     this.x = val - this.x;
 }

 public void f(A a)  {
  this.x = this.x + a.x + 1;
 }
```



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

```
 public void f(B b)  {
  this.x = this.x * b.x;
 }

 public void f(D d)  {
  this.x = d.x - 1;
 }
 public String toString(){
   return "D: " +
       super.toString();
 }

} //end of class D

public class Driver
{
 public static void main
       (String [] args){
  A a;
  B b;
  int m;
//    ............
 }
}
```

```
public abstract class A {
 public int x;

 public A() {
  this.x = 1;
 }
 public abstract int F(int v);

 public virtual void F(A a) {
  x = a.x;
 }

 public override
       string ToString(){
  return "x =" + this.x;
 }
} //end of class A

public class B : A {
 public B():base() {
 }
```



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

```
 public B(int val):base(){
  this.x = F(val);
 }
 public override int F(int val){
  return this.x + val;
 }
 public virtual void F(B b){
  this.x = this.x * b.x;
 }
 public override
     string ToString(){
 return "B: " + base.ToString();
 }
} //end of class B

public class D : B {
 public D(int val):base(){
     this.x = val - this.x;
 }

 public override void F(A a){
  this.x = this.x + a.x + 1;
 }
 public override void F(B b){
  this.x = this.x * b.x;
 }
 public void F(D d)  {
  this.x = d.x - 1;
 }
 public override string ToString(){return "D: " + base.ToString();}

} //end of class D

public class Driver {
 public static void Main (string [] args){
  A a;
  B b;
  int m;
//    ..............
 }
}
```

**בכל אחד מסעיפים הבאים השורה (*) תוחלף לקטע קוד רלוונטי.**



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

**א. מהו הפלט של קטע הקוד שלפניכם:**

```
m = 11;
a = new B(m);
System.out.println(a);
a = new D(m);
System.out.println(a);
```

```
m = 11;
a = new B(m);
Console.WriteLine(a);
a = new D(m);
Console.WriteLine(a);
```

(4 נקודות)

**ב. מהו הפלט של קטע הקוד שלפניכם:**

```
b = new B(3);
m = b.f(4);
System.out.println(m);
m = b.f(m);
System.out.println(m);
```

```
b = new B(3);
m = b.F(4);
Console.WriteLine(m);
m = b.F(m);
Console.WriteLine(m);
```

(4 נקודות)

**ג. מהו הפלט של קטע הקוד שלפניכם:**

```
a = new B(5);
int m = a.f(a.f(2));
System.out.println(m);
```

```
a = new B(5);
m = a.F(a.F(2));
Console.WriteLine(m);
```

(3 נקודות)

**ד. מהו הפלט של קטע הקוד שלפניכם:**

```
a = new D(10);
b = new B(20);
a.f(b);
System.out.println(a);
```

```
a = new D(10);
b = new B(20);
a.F(b);
Console.WriteLine(a);
```

(3 נקודות)

**ה. אחרי ביצוע קטע הקוד הבא, הערך של התכונה x של העצם d יהיה 5.**
**מהו הערך של המשתנה m?**

```
D d = new D(m);
b = new B();
((A)d).f(b);
```

```
D d = new D(m);
b = new B();
((A)d).F(b);
```

(4 נקודות)


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

### 9. נתונות שתי הפעולות isExist1 ו-isExist2

```
public static boolean isExist1(Stack<Integer> st, int x){
  while(!st.isEmpty()){
    if(st.pop()==x) return true;
  }
  return false;
}
public static boolean isExist2(Stack<Integer> st, int x){
  Stack<Integer> temp = new Stack<Integer>();
  boolean f = false;
  while(!st.isEmpty()){
    if(st.top()==x) f = true;
    temp.push(st.pop());
  }
  while(!temp.isEmpty()) {
    st.push(temp.pop());
  }
  return f;
}
```

```
public static bool IsExist1(Stack<int> st, int x){
  while(!st.IsEmpty()){
    if(st.Pop()==x) return true;
  }
  return false;
}
public static bool IsExist2(Stack<int> st, int x){
  Stack<int> temp = new Stack<int>();
  bool f = false;
  while(!st.IsEmpty()){
    if(st.Top()==x) f = true;
    temp.Push(st.Pop());
  }
  while(!temp.IsEmpty()) {
    st.Push(temp.Pop());
  }
  return f;
}
```

**א. הסבירו מה מבצעת כל אחת מהפעולות, מה הבדל ביניהן ומהי הסיבוכיות של כל אחת מהן.**

(3 נקודות)



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

| st1 | 2 | 3 | 10 | 7 |
|---|---|---|---|---|

| st2 | 3 | 14 | 2 | 2 | 3 | 7 | 10 | 2 | 10 |
|---|---|---|---|---|---|---|---|---|---|

**ב.** כתבו פעולה המקבלת שתי מחסניות של מספרים שלמים st1 ו-st2. הפעולה תבדוק אם כל האיברים של המחסנית st1 נמצאים במחסנית st2. אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

**לדוגמה**, עבור זוג המחסניות שלפניכם הפעולה תחזיר true (איור לסעיף ב).

(4 נקודות)

| st2 | 3 | 14 | 2 | 2 | 3 | 7 | 10 | 2 | 7 |
|---|---|---|---|---|---|---|---|---|---|

**ג.** כתבו פעולה המקבלת שתי מחסניות של מספרים שלמים st1 ו-st2. הפעולה תבדוק אם כל האיברים של המחסנית st1 נמצאים במחסנית st2 **באותו הסדר**. אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

**לדוגמה**, עבור זוג המחסניות מסעיף ב' הפעולה תחזיר false. אבל אם המחסנית st2 תהיה כך (איור לסעיף ג) הפעולה תחזיר ערך **true**.

(4 נקודות)

| st2 | 3 | 14 | 2 | 2 | 3 | 10 | 7 | 2 | 7 |
|---|---|---|---|---|---|---|---|---|---|

**ד.** כתבו פעולה המקבלת שתי מחסניות של מספרים שלמים st1 ו-st2. הפעולה תבדוק אם המחסנית st1 היא תת-מחסנית של st2 (כלומר, אם כל האיברים של המחסנית st1 נמצאים במחסנית st2 **ברצף ובאותו הסדר**). אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

**לדוגמה**, עבור זוג המחסניות מסעיף ב' או ג' הפעולה תחזיר false. אבל אם המחסנית st2 תהיה כך (איור לסעיף ד) הפעולה תחזיר ערך true.

(4 נקודות)

**ה.** מהי הסיבוכיות של כל אחת מהפעולות שכתבתם? הסבירו את תשובתכם.

(3 נקודות)

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


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