GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: אביב 2026 מועד ב'
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 כל שאלה 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!** בנספחי הבחינה מובאים ממשקים (interface) הכוללים פונקציות ומחלקות עזר, יש להקפיד ולהשתמש בהן בשאלות הרלוונטיות בבחינה כדי להימנע מאיבוד ניקוד. 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.** שני מספרים שלמים וחיוביים num1 ו-num2 נקראים "מצטלבים" (crossed) אם יש להם בדיוק מחלק משותף אחד (פרט ל-1).

**לדוגמה:**

המספרים 14 ו-21 הם מצטלבים כי יש להם מחלק משותף אחד בלבד (7).

המספרים 3 ו-27 הם מצטלבים כי יש להם מחלק משותף אחד בלבד (3).

המספרים 8 ו-9 אינם מצטלבים כי אין להם שום מחלק משותף.

המספרים 12 ו-36 אינם מצטלבים כי יש להם יותר ממחלק משותף אחד (2, 3, 6, 12).

כתבו פעולה המקבלת שני מספרים שלמים חיוביים. הפעולה תבדוק אם הם "מצטלבים". אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

כותרת הפעולה:

**Java:**

```
public static boolean crossed(int num1, int num2)
```

**C#:**

```
public static bool Crossed(int num1, int num2)
```

(6 נקודות)

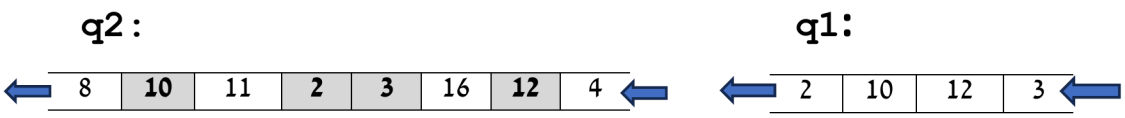
**ב.** כתבו פעולה crossedStacks המקבלת שתי מחסניות של מספרים שלמים חיוביים st1 ו-st2. הפעולה תבדוק שלכל איבר במחסנית st1 יש איבר מצטלב במחסנית st2. אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

(8 נקודות)

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

(2 נקודות)

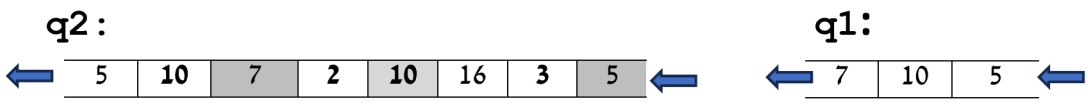
**2.**




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

**לדוגמה:** עבור התורים q1 ו-q2 הבאים הפעולה תחזיר ערך true (ראו איור לשאלה).

(7 נקודות)




**ב.** כתבו פעולה המקבלת שני תורים של מספרים שלמים q1 ו-q2. הפעולה תבדוק אם כל הערכים של התור q1 נמצאים בתור q2 **באותו הסדר**. אם כן, הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.

**לדוגמה:** עבור התורים q1 ו-q2 הבאים הפעולה תחזיר ערך true (ראו איור לשאלה).

(7 נקודות)

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

(2 נקודות)


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

**3.** חברת Cloud-Tech מספקת תשתיות מחשוב. בתחילה, החברה סיפקה רק רכיבי חומרה, אך עם הזמן התרחבה גם למכירת רישיונות תוכנה. לפניכם המחלקות המגדירות את רכיבי המערכת:

- המחלקה Hardware עבור רכיבי חומרה פיזיים;
- המחלקה Server עבור שרתי מחשוב;
- המחלקה Switch עבור מתגי תקשורת;
- המחלקה SoftwareLicense עבור רישיונות תוכנה (כגון Windows או Oracle).

**Java:**

```
public class Hardware {
   private String serialNumber; // מספר סידורי
   private double price;       // מחיר יחידה
   public double getPrice() { return this.price; }
}
public class Server extends Hardware {
   private int ramGB;     // זיכרון בגיגה בייט
   private int cpuCores;  // מספר ליבות מעבד
}
public class Switch extends Hardware {
   private int portsCount; // מספר פורטים
   private boolean managed; // האם מנוהל
   public int getPortsCount() { return this.portsCount; }
   public boolean isManaged(){return this.managed;}
}
public class SoftwareLicense {
   private String softwareName; // שם התוכנה
   private int months; // משך הרישיון בחודשים
   private double monthlyRate;  // עלות חודשית
   public double getPrice(){
     return (this.months * this.monthlyRate);
   }
}
```

**C#:**

```
public class Hardware {
   private string serialNumber; // מספר סידורי
   private double price;       // מחיר יחידה
   public double GetPrice() { return this.price; }
}
public class Server : Hardware {
   private int ramGB;     // זיכרון בגיגה בייט
   private int cpuCores;  // מספר ליבות מעבד
}
```

```
public class Switch : Hardware {
  private int portsCount; // מספר פורטים
  private bool managed; // האם מנוהל
  public int GetPortsCount() { return this.portsCount; }
  public bool IsManaged(){return this.managed;}
}
public class SoftwareLicense {
  private string softwareName; // שם התוכנה
  private int months; // משך הרישיון בחודשים
  private double monthlyRate;  // עלות חודשית
  public double GetPrice(){
    return (this.months * this.monthlyRate);
  }
}
```

**א.** בתחילת דרכה, המחלקה ProjectOrder ניהלה רק רכיבי חומרה המסופקים ללקוח.
למחלקה התכונות האלה:
- `String clientName` – שם הלקוח; (`C#: `string clientName`)
- `Hardware[] items` – מערך של פריטי חומרה בלבד.

כתבו במחלקה ProjectOrder את הפעולה countLargeSwitches. הפעולה תחזיר כמות מתגי תקשורת הניתנים לניהול שיש להם יותר מ-16 פורטים.

(6 נקודות)

**ב.** החברה החליטה לאפשר ללקוחות להוסיף להזמנה גם רישיונות תוכנה (SoftwareLicense).
כעת המחלקה ProjectOrder צריכה להחזיק במערך אחד את **כל** סוגי המוצרים (גם חומרה וגם רישיונות).

**הערה:** אין להוסיף מחלקות או לשנות את יחסי הירושה הקיימים.

(1) אילו שינויים יש לבצע בהגדרת התכונות של המחלקה ProjectOrder כדי לתמוך בדרישה החדשה?

(2) כתבו פעולה פנימית `public double calculateTotal()` (C#: `public double CalculateTotal()`).
הפעולה תחזיר את העלות הכוללת של כל הפריטים במערך החדש.

(10 נקודות)


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

**4.** נתון פרויקט שהוגדרו בו ארבע מחלקות ושני ממשקים:

**Java:**

```
interface IConnectable {void connect(String network);}
interface IRemoteControllable { void performAction(); }

public class EDevice {
  private String model;

  public EDevice(String model) { this.model = model; }

  public void printInfo() {
    System.out.println("Device: " + model);
  }
}

public class SmartLamp extends EDevice
          implements IRemoteControllable {
  public SmartLamp(String model) { super(model); }
  public void performAction() {
      System.out.println("Lamp: Lighting up");
  }
}

public class SmartTV extends EDevice
      implements IConnectable, IRemoteControllable {
  public SmartTV(String model) { super(model); }
  public void connect(String net) {
      System.out.println("TV connected to " + net);
  }
  public void performAction() {
      System.out.println("TV: Playing movie");
  }
}

public class RobotVacuum extends EDevice
        implements IConnectable {
  public RobotVacuum(String model) { super(model); }
  public void connect(String net) {
      System.out.println("Vacuum ready on " + net);
  }
}
```

**C#:**

```
interface IConnectable {void Connect(string network);}
interface IRemoteControllable { void PerformAction(); }

public class EDevice {
  private string model;

  public EDevice(string model) { this.model = model; }

  public void PrintInfo() {
    Console.WriteLine("Device: " + model);
  }
}

public class SmartLamp : EDevice, IRemoteControllable {
  public SmartLamp(string model) : base(model) {
  }
   public void PerformAction() {
     Console.WriteLine("Lamp: Lighting up");
  }
}

public class SmartTV : EDevice, IConnectable,
            IRemoteControllable {
   public SmartTV(string model) : base(model) {
   }
   public void Connect(string net) {
     Console.WriteLine("TV connected to " + net);
   }
   public void PerformAction() {
     Console.WriteLine("TV: Playing movie");
   }
}

public class RobotVacuum : EDevice, IConnectable {
   public RobotVacuum(string model) : base(model) {
   }
   public void Connect(string net) {
     Console.WriteLine("Vacuum ready on " + net);
   }
}
```

**א.** נתון קטע קוד:

**Java:**

```
EDevice tv = new SmartTV("Sony-4K");
EDevice lamp = new SmartLamp("Lux");
IConnectable vacuum = new RobotVacuum("iRobot");
// .... //
```

**C#:**

```
EDevice tv = new SmartTV("Sony-4K");
EDevice lamp = new SmartLamp("Lux");
IConnectable vacuum = new RobotVacuum("iRobot");
// .... //
```

**עבור כל אחד מהקטעים הבאים (1-6), כתבו אם הוא תקין או גורם לשגיאה כאשר מוסיפים אותו במקום המסומן ב-.**

**אם הקוד תקין - הסבירו מה יהיה הפלט. אם לא - הסבירו מדוע.**

**Java:**

```
// 1
tv.printInfo();
tv.connect("Home_Net");

// 2
IRemoteControllable r1 = new SmartLamp("LED");
r1.printInfo();

// 3
if (vacuum instanceof EDevice) {
  ((EDevice)vacuum).printInfo();
}

// 4
SmartTV tv2 = (SmartTV)(new EDevice("Generic"));
tv2.printInfo();

// 5
IConnectable c1 = (IConnectable)lamp;
c1.connect("Guest_WiFi");

// 6
```



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

```
IRemoteControllable r2 = vacuum;
r2.performAction();
```

### C#:

```
// 1
tv.PrintInfo();
tv.Connect("Home_Net");

// 2
IRemoteControllable r1 = new SmartLamp("LED");
r1.PrintInfo();

// 3
if (vacuum is EDevice) {
   ((EDevice)vacuum).PrintInfo();
}

// 4
SmartTV tv2 = (SmartTV)(new EDevice("Generic"));
tv2.PrintInfo();

// 5
IConnectable c1 = (IConnectable)lamp;
c1.Connect("Guest_WiFi");

// 6
IRemoteControllable r2 = vacuum;
r2.PerformAction();
```

(12 נקודות)

**ב.** כתבו פעולה המקבלת מערך של מכשירי חשמל (מערך עצמים מסוג EDevice) ושם רשת.

הפעולה צריכה:

1. להדפיס את המידע של כל מכשיר;
2. אם מכשיר ניתן לחיבור - לחבר אותו לרשת;
3. אם מכשיר ניתן לשליטה מרחוק - להפעיל אותו.

כותרת הפעולה:

**Java:**

```
public static void setupHome(EDevice[] devices, String network)
```

**C#:**

```
public static void SetupHome(EDevice[] devices, string network)
```

(4 נקודות)

**5.**

**א. נתונה הפעולה secret (C#: Secret):**

**Java:**

```
public static boolean secret(Stack<Integer> s) {
  if (s.isEmpty()) return true;
  boolean res = true;
  while (!s.isEmpty()) {
    int x = s.pop();
    if (!s.isEmpty()) {
      if (x * s.top() >= 0) {
        res = false;
      }
    }
  }
  return res;
}
```

**C#:**

```
public static bool Secret(Stack<int> s) {
  if (s.IsEmpty()) return true;
  bool res = true;
  while (!s.IsEmpty()) {
    int x = s.Pop();
    if (!s.IsEmpty()) {
      if (x * s.Top() >= 0) {
        res = false;
      }
    }
  }
  return res;
}
```

(1) תנו דוגמה למחסנית s הכוללת לפחות שישה ערכים שלמים (גם חיוביים וגם שליליים), שעבורה זימון הפעולה secret(s) יחזיר ערך true. אם אין מחסנית כזאת - הסבירו את הסיבה לכך.

(2) מה מבצעת הפעולה secret באופן כללי עבור מחסנית של מספרים שלמים?

(6 נקודות)



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

**ב.** נתונה הפעולה mystery (C#: Mystery):

**Java:**

```
public static boolean mystery(Queue<Integer> q) {
  if (q.isEmpty()) return true;
  boolean res = true;
  while (!q.isEmpty()) {
    int first = q.remove();
    if (!q.isEmpty()) {
      int second = q.remove();
      if (first * second <= 0) {
        res = false;
      }
    }
  }
  return res;
}
```

**C#:**

```
public static bool Mystery(Queue<int> q) {
  if (q.IsEmpty()) return true;
  bool res = true;
  while (!q.IsEmpty()) {
    int first = q.Remove();
    if (!q.IsEmpty()) {
      int second = q.Remove();
      if (first * second <= 0) {
        res = false;
      }
    }
  }
  return res;
}
```

(1) תנו דוגמה לתור q הכולל לפחות שישה ערכים שלמים (גם חיוביים וגם שליליים), שעבורו זימון הפעולה mystery(q) יחזיר ערך true. אם אין תור כזה - הסבירו את הסיבה לכך.

(2) מה מבצעת הפעולה mystery באופן כללי עבור תור של מספרים שלמים?

(6 נקודות)



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

ג. כתבו פעולה בשם check (C#: Check) המקבלת מחסנית של מספרים שלמים חיוביים ומספר שלם חיובי k. הפעולה תבדוק אם כל איברי המחסנית הם כפולות של k (מתחלקים ב-k ללא שארית). אם כן - הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.
בסיום הפעולה, **על המחסנית לחזור למצב המקורי**.
(4 נקודות)

**6.** בנמל אשדוד עומדת מעבורת מטען המיועדת להובלת משאיות לאירופה. המשאיות עולות למעבורת בזו אחר זו ומסתדרות בטור. המשאית שנכנסה ראשונה למעבורת חונה קרוב לפתח היציאה, ולכן תהיה הראשונה לצאת מהאונייה בנמל היעד.
המחלקה Truck מייצגת משאית:

**Java:**

```
class Truck {
  private String plateNumber;  // מספר רישוי
  private int weight;          // משקל המשאית (בטונות)
  private int cargoValue;      // ערך המטען של המשאית (בשקלים)
}
```

**C#:**

```
class Truck {
  private string plateNumber;  // מספר רישוי
  private int weight;          // משקל המשאית (בטונות)
  private int cargoValue;      // ערך המטען של המשאית (בשקלים)
}
```

אפשר להניח שבמחלקה הוגדרו בנאי ופעולות get (C#: Get) לכל התכונות.

א. הגדירו מחלקה בשם Ferry (מעבורת) המתארת את ריכוז המשאיות שעליה. למחלקה שתי תכונות:
- תור של משאיות (`Queue<Truck> trucks`);
- משקל מקסימלי שהמעבורת יכולה לשאת (maxCapacity).

כתבו פעולה בונה (המקבלת כפרמטר את המשקל המרבי) ומאתחלת מעבורת ריקה (ללא משאיות).
(3 נקודות)


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

**ב.** מטעמי בטיחות וניהול סיכונים בלב ים, המעבורת תיחשב "**מעבורת תקינית**" אם היא עומדת בשני התנאים הבאים:

1. **מגבלת משקל כוללת:** המשקל המצטבר של **כל** המשאיות שעל המעבורת אינו עולה על המשקל המקסימלי המותר (maxCapacity).
2. **סדר ערך המטען:** מטעמי ביטוח, המשאית הראשונה שתצא מהאונייה (זו שנמצאת ראש התור) חייבת להיות בעלת המטען **הזול ביותר**, והערך עולה ככל שמתקרבים לסוף התור. לכן, ערך המטען (cargoValue) של כל משאית בתור חייב להיות **נמוך או שווה** לערך המטען של המשאית שנמצאת אחריה.

כתבו פעולה פנימית בתוך המחלקה Ferry הבודקת האם המעבורת היא "מעבורת תקינה" לפי שני התנאים לעיל. **שימו לב:** בסיום הפעולה מצב התור (סדר וערך האיברים) חייב לחזור לקדמותו.

כותרת הפעולה

**Java:**

```
public boolean isValid()
```

**C#:**

```
public bool IsValid()
```

(8 נקודות)

**ג.** כתבו פעולה המקבלת משאית (אובייקט מסוג Truck). על הפעולה לנסות להוסיף את המשאית ל**סוף התור** של המעבורת. ההוספה תתבצע רק אם המעבורת תישאר "מעבורת תקינה" לאחר ההוספה.

- אם ההוספה בוצעה - הפעולה מוסיפה את המשאית ומחזירה true.
- אם הוספת המשאית תהפוך את המעבורת ל"לא תקינה" - **אין להוסיף** את המשאית והפעולה תחזיר false.

כותרת הפעולה

**Java:**

```
public boolean addTruck(Truck t)
```

**C#:**

```
public bool AddTruck(Truck t)
```

(5 נקודות)


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

## 7. נתונות הפעולות הבאות:

### Java:

```
public static int what(BinNode<Character> bt) {
  if (bt == null) return 0;
  return 1 + Math.max(what(bt.getLeft()),
              what(bt.getRight()));
}

public static void printTree(BinNode<Character> bt) {
  if (bt == null) return;

  printTree(bt.getLeft());

  int leftH = what(bt.getLeft());
  int rightH = what(bt.getRight());
  int count = leftH + rightH + 1;

  for (int i = 0; i < count; i++) {
    System.out.print(bt.getValue());
  }

  printTree (bt.getRight());
}
```

### C#:

```
public static int What(BinNode<char> bt) {
  if (bt == null) return 0;
  return 1 + Math.Max(What(bt.GetLeft()),
              What(bt.GetRight()));
}

public static void PrintTree(BinNode<char> bt) {
  if (bt == null) return;

  PrintTree(bt.GetLeft());

  int leftH = What(bt.GetLeft());
  int rightH = What(bt.GetRight());
  int count = leftH + rightH + 1;

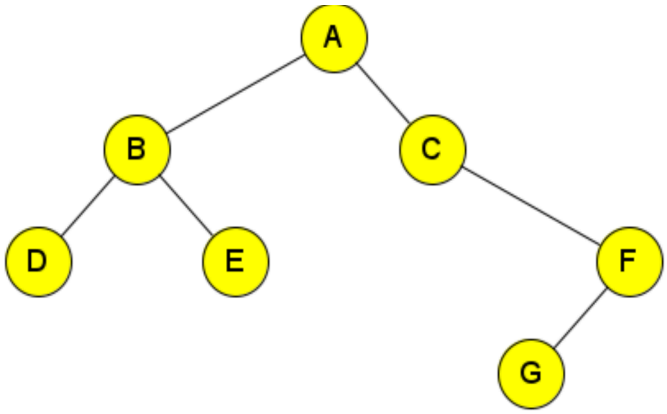
  for (int i = 0; i < count; i++) {
    Console.Write(bt.GetValue());
```


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

```
    }

    PrintTree (bt.GetRight());
}
```

נתון עץ בינרי bt:




**א.** מה תהיה תוצאת הזימון what(bt)?

(3 נקודות)

**ב.** עבור הצומת B, כמה פעמים תודפס האות 'B'?

(3 נקודות)

**ג.** מה תהיה תוצאת הזימון printTree(bt)? **יש להראות מעקב אחרי זימון הפעולה.**

(8 נקודות)

**ד.** ציירו עץ בינרי שעבורו פלט של הפעולה printTree יהיה EDDCCCBBBBAAAAA.

(4 נקודות)



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

**8.** ב"מכללת אלונים" לומדים חמישה ימים בשבוע. בכל יום לימודים מתקיימים עשרה שיעורים, ובכל שיעור לומדת בחדר קבוצת לימוד אחת. לכל קבוצת לימוד יש מספר ייחודי (מספר שלם חיובי).
לכל חדר קיימת מערכת שיעורים משלו: מערך דו-ממדי (arr) הכולל עבור כל יום וכל שיעור את מספר הקבוצה הלומדת בו (המספר 0 מציין שהחדר פנוי). גודל המערך 10X5.
במכללה יש שלושה סוגים של חדרי לימוד: חדר רגיל, מעבדת מחשבים וחדרי מולטימדיה.
צוות המכללה מעוניין לפתח פרויקט לשיבוץ חדרים. לשם כך נאסף מידע לגבי כל אחד מחדרי הלימוד:

**עבור חדר לימוד רגיל (ClassRoom):**

- מספר החדר (roomNum)
- מספר מקומות הישיבה בחדר (seats)
- מערך של מערכת שיעורים (arr)

**עבור מעבדת מחשבים (CompRoom):**

- מספר החדר (roomNum)
- מספר מקומות הישיבה בחדר (seats)
- מערך של מערכת שיעורים (arr)
- מספר מחשבים (comps)

**עבור חדר מולטימדיה (MediaRoom):**

- מספר החדר (roomNum)
- מספר מקומות הישיבה בחדר (seats)
- מערך של מערכת שיעורים (arr)
- מספר מחשבים (comps)
- מערך שמות של סוגי ציוד המצויים בחדר (equips)

**א.** שרטטו תרשים UML המתאר את הקשר בין המחלקות MediaRoom, CompRoom ו-ClassRoom באופן **המתאים ביותר לעקרונות של תכנות מונחה עצמים**.
לכל אחת מהמחלקות ClassRoom, CompRoom, MediaRoom כתבו:

- כותרת המחלקה;
- תכונות;
- פעולה בונה - הפעולה הבונה של כל מחלקה מקבלת את כל הפרמטרים הנדרשים.

**הערה**: הניחו שהפעולות get ו-set מוגדרות עבור כל תכונה בכל אחת מהמחלקות. גם פעולות toString() הוגדרו בכל אחת מהמחלקות.
(4 נקודות)

ב. • כתבו פעולה המקבלת מחרוזת המייצגת סוג ציוד ובודקת אם הציוד קיים בחדר. אם כן - הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false. כותרת הפעולה:

**Java:**

```
boolean isEquipmentExist(String eq)
```

**C#:**

```
bool IsEquipmentExist(string eq)
```

• כתבו פעולה המקבלת מספר המייצג יום בשבוע (מ-1 עבור יום ראשון ועד 5 עבור יום חמישי) ומספר שיעור לימוד (בין 1 ל-10). הפעולה תבדוק אם החדר פנוי בשיעור זה. אם כן - הפעולה תחזיר ערך true, ואם לא - הפעולה תחזיר ערך false.
ניתן להניח כי הפרמטרים של הפעולה תקינים. כותרת הפעולה:

**Java:**

```
boolean isFree(int day, int lesson)
```

**C#:**

```
bool IsFree(int day, int lesson)
```

**עבור כל פעולה יש לציין באיזו מחלקה צריך לממש את הפעולה.**

(6 נקודות)


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

**ג.** הוגדרה המחלקה College לניהול חדרי לימוד במכללה. למחלקה תכונה אחת.

```
public class College {
    private _____ rooms;
}
```

- השלימו את החסר בהגדרת התכונה.
- כתבו במחלקה College פעולה המקבלת יום day ומספר שיעור lesson ומדפיסה את הפרטים של כל **מעבדות מחשב** הפנויות באותו היום ובאותו השיעור. כותרת הפעולה:

**Java:**

```
void printFreeCompRooms(int day, int lesson)
```

**C#:**

```
void PrintFreeCompRooms(int day, int lesson)
```

אפשר להניח שהפרמטרים תקינים.

- כתבו במחלקה College פעולה בשם collectEquipmentExist (C#: CollectEquipmentExist) המקבלת מחרוזת המייצגת סוג ציוד **ומחזירה** את אוסף חדרי מולטימדיה שבהם קיים סוג הציוד הזה.

**הערה:**

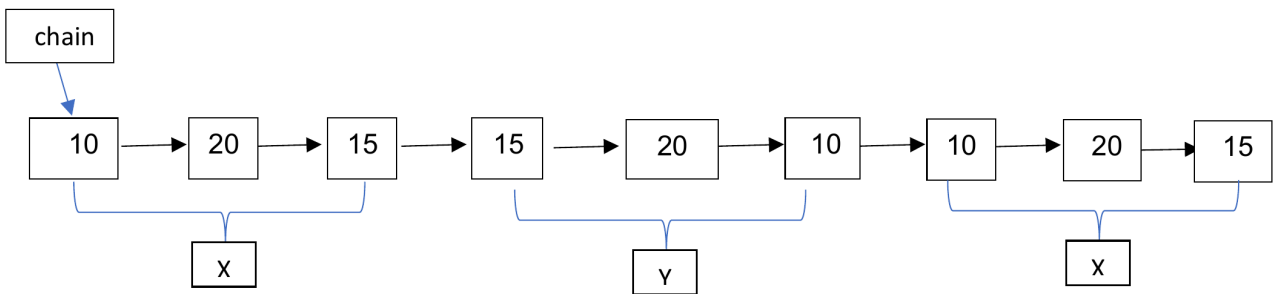
אוסף יכול להיות מערך, תור, מחסנית או שרשרת חוליות. אם החלטתם להחזיר מערך, הוא צריך לכלול **רק** חדרים שבהם הציוד קיים, ללא מקומות ריקים.

(8 נקודות)

**9.** נתונה הגדרה הבאה:

שרשרת חוליות נקראת "שרשרת XYX" אם היא מורכבת משלושה חלקים כך שחלק הראשון זהה לחלק האחרון, והחלק האמצעי הפוך לחלק הראשון (וגם לחלק האחרון).

לדוגמה, השרשרת שלפניכם היא "שרשרת XYX":





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

**א.** כתבו פעולה המקבלת הפניה לחוליה הראשונה של שרשרת חוליות של מספרים שלמים.
הפעולה תבדוק אם שרשרת היא "שרשרת XYX". אם כן - הפעולה תחזיר ערך true, ואם לא - ערך false.
כותרת הפעולה:

**Java:**

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

**C#:**

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

(16 נקודות)

**ב.** מהי סיבוכיות הפעולה isXYX שכתבתם בסעיף א'? **הסבירו את תשובתכם.**
(2 נקודות)

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