

תאריך:

כיתה יא-1

שם התלמיד/ה: \_\_\_\_\_

שם המורה: צבי מלמד

## מבחן במדעי המחשב כיתות יא - בחינת אימון

### משך הבחינה (ללא תוספת זמן) X דקות

#### נושאי המבחן:

- רשימות  $\text{Node}\langle T \rangle$
- רקורסיות - כתיבה
- רקורסיות - מעקב

#### שאלה מספר 1 – רשימה פלינדרומית (25 נקודות)

נתונה רשימה מקושרת חד-כיוונית של מספרים שלמים.

נאמר כי רשימה היא פלינדרומית אם סדר האיברים בה זהה מההתחלה לסוף ומהסוף להתחלה. כלומר, האיבר הראשון שווה לאחרון, השני שווה לאיבר לפני האחרון וכן הלאה.

לדוגמה: , הרשימה הבאה היא רשימה פלינדרומית

$3 \rightarrow 7 \rightarrow 5 \rightarrow 7 \rightarrow 3 \rightarrow \text{null}$

הרשימה הבאה אינה רשימה פלינדרומית

$4 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow \text{null}$

#### סעיף א'

כתוב בשפה פשוטה וברורה את האלגוריתם לבדיקת האם רשימה נתונה היא פלינדרומית. יש לפרט את שלבי הפעולה, ללא כתיבת קוד.

#### סעיף ב'

כתוב פעולה חיזונית (סטטית) בשפת C# בשם `IsPalindrome` המקבלת הפניה לראש רשימה מקושרת של מספרים שלמים, ומחזירה:

`true` ← אם הרשימה היא פלינדרומית, ו- `false` אם היא איננה כזאת.

#### מגבלות:

- יש לשמור על הרשימה המקורית ללא שינוי בסיום הפעולה
- אין להפוך את הרשימה למערך
- אין להשתמש במבני נתונים נוספים מלבד רשימות מקושרות ו/או מחסנית
- מותר להשתמש ברקורסיה



## שאלה 2

נתון הקוד הבא:

```
static public int Calc(int[] arr, int index)
{
    if (index < 0)
        return 0;

    int val = Transform(arr[index]);
    return val + Calc(arr, index - 2);
}

static public int Transform(int num)
{
    int sum = 0;

    while (num > 0)
    {
        sum += num % 10;
        num /= 10;
    }

    if (sum % 2 == 0)
        return sum;

    return 0;
}

//=====
static void Main(string[] args)
{
    int[] arr = { 123, 45, 78, 9, 246 };

    Console.WriteLine(Calc(arr, 4)); // סעיף א
    Console.WriteLine(Calc(arr, 3)); // סעיף ב
}
```



### סעיף א'

יש לבצע מעקב מלא אחר הקריאה  $\text{Calc(arr, 4)}$ .

- יש להציג תרשים עץ קריאות או טבלת מעקב.
- יש לציין את הערך המוחזר מכל קריאה רקורסיבית.
- אין צורך לעקוב אחר הלולאה בפעולה Transform, אך יש לציין את הערך המוחזר ממנה.

מהו הפלט המודפס בסעיף א'?

### סעיף ב'

מהו הפלט המודפס כתוצאה מהקריאה  $\text{Calc(arr, 3)}$ ?

אין חובה לבצע מעקב מלא, אך יש לנמק את התשובה.

### סעיף ג'

מה מבצעת הפעולה Calc?

יש לנסח תשובה קצרה וברורה, במילים בלבד.



### שאלה 3

נתון הקוד הבא:

```
static int Process(Node<int> head)
{
    if (head == null)
        return 0;

    int value = head.GetValue();

    if (value % 2 == 0)
        return value + Process(head.GetNext());

    return Process(head.GetNext()) - value;
}

//=====
static void Main(string[] args)
{
    Node<int> lst =
        new Node<int>(3,
            new Node<int>(4,
                new Node<int>(1,
                    new Node<int>(6,
                        new Node<int>(2, null)))));

    Console.WriteLine(Process(lst));           // סעיף א
    Console.WriteLine(Process(lst.GetNext())); // סעיף ב
}
```

### סעיף א'

יש לבצע מעקב מלא אחר הקריאה `Process(lst)`.  
יש להציג **טבלת מעקב**  
בכל שורה יש לציין:

- ערך החוליה הנוכחית
- הערך המוחזר מהקריאה הרקורסיבית
- הערך המוחזר מהקריאה הנוכחית

מהו הערך המודפס?

### סעיף ב'

מהו הערך המודפס בקריאה  
`Process(lst.GetNext())`?

אין חובה לבצע טבלת מעקב מלאה, אך יש לנמק את התשובה.

### סעיף ג'

מה מבצעת הפעולה `Process`?  
יש לנסח תשובה קצרה וברורה, במילים בלבד.



#### שאלה 4

נתונה רשימה מקושרת חד-כיוונית של מספרים שלמים .  
נאמר כי מתבצע **פיצול** של הרשימה בכל מקום שבו משתנה **כיוון הסדרה** :  
מעבר מסדרה עולה לסדרה יורדת  
או מעבר מסדרה יורדת לסדרה עולה  
כל תת-רשימה חייבת להיות רציפה ולשמור על סדר האיברים המקורי .

#### דוגמה

#### **בהינתן הרשימה**

1 → 3 → 5 → 4 → 2 → 6 → 8 → 7 → null

יתקבל מערך של תת-רשימות :

arr[0]: 1 → 3 → 5 → null

arr[1]: 4 → 2 → null

arr[2]: 6 → 8 → null

arr[3]: 7 → null

כתוב פעולה חיזונית (סטטית) בשפת C#, המקבלת הפניה לראש רשימה מקושרת של מספרים שלמים, ומחזירה מערך של רשימות מקושרות, בהתאם להגדרת הפיצול שלעיל.

#### הנחיות ומגבלות

- מותר לשנות ("לקלקל") את הרשימה המקורית
- אין צורך לשכפל איברים
- אין להפוך את הרשימה למערך
- אין להשתמש במבני נתונים נוספים מלבד רשימות מקושרות ומערך ההחזרה
- סדר האיברים בכל תת-רשימה חייב להישמר



## שאלה 5

נתונה רשימה מקושרת מטיפוס כלשהו ע"י  $\langle T \rangle$  Node. כתוב פעולה חיזונית (סטטית) בשפת C#, המקבלת הפניה לראש הרשימה המקושרת הזאת, ומתשנה נוסף שנקרא לו split – (פיצול). הפעולה מחזירה מערך של רשימות מקושרות שפוצלו מהרשימה המקורית בכל פעם שנמצא הערך split.

### דוגמה

#### **בהינתן הרשימה**

"red" → "blue" → "stop" → "green" → "yellow" → "stop" → "black" → null

והמשתנה split ערכו הוא "stop" אז יוחזר המערך הבא :

```
arr[0]: "red" → "blue" → null
arr[1]: "green" → "yellow" → null
arr[2]: "black" → null
```

או, ערך המשתנה split הוא 0, והרשימה היא :

→ 1 → 8 → 0 → 5 → 0 → 7 → 3 null

אזי יוחזר המערך :

```
arr[0]: 3 → 7 → null
arr[1]: 5 → null
arr[2]: 8 → 1 → null
```

### הנחיות ומגבלות

- מותר לשנות ("לקלקל") את הרשימה המקורית
- אין צורך לשכפל איברים
- אין להפוך את הרשימה למערך (פרט למערך שהיא צריכה להחזיר)
- אין להשתמש במבני נתונים נוספים מלבד :
  - רשימות מקושרות
  - מערך ההחזרה
- סדר האיברים בכל תת-רשימה חייב להישמר
- להשוואת ערכים יש להשתמש בפעולה : Equals

### שאלה 6 – מאתגרת

פתרו שוב את שאלה #4 בהבדל הבא : אין לשנות או לקלקל את הרשימה המקורית. כלומר הרשימות החדשות (המפוצלות) שאתם יוצרים צריכות להיות העתק של הרשימה המקורית.