

# מבחן בנושא WinForms + מערכים + פונקציות

- המבחן עם חומר פתוח
- ניתן להשתמש באינטרנט
- המבחן מול מחשב (אסור שימוש באפליקציות כלשהן לשיתוף מידע!)
- אין אפשרות לשימוש במכשירים סלולריים

## חלק א' - פונקציות

1. תקן את קטע הקוד הבא (מבלי לשנות את הערך המוחזר של פונקציית Power2):

```
static void Main(string[] args)
{
    Console.WriteLine("Enter a number:");
    int x = Convert.ToInt32(Console.ReadLine());
    Power2(x);
    Console.WriteLine($"power 2 of your number is {x}");
}

private static void Power2(int i)
{
    i = i * 2;
}
```

2. בפונקציה הבאה, האם ישתנה ערך האיבר הראשון במערך a ב main? **מדוע כן/לא?**

```
static void Update(int[] a)
{
    a[0] = 1;
}

static void Main(string[] args)
{
    int[] a = { 5, 4, -5 };
    Update(a);
}
```

3. בפונקציה הבאה, האם ישתנה ערך האיבר הראשון במערך a ב main? **מדוע כן/לא?**

```
static void Update(int[] a)
{
    a = new int[3];
    a[0] = 1;
}

Static void Main(string[] args)
{
    int[] a = { 5, 4, -5 };
    Update(a);
}
```

4. תקן את קטע הקוד הבא

(מבלי לשנות את השורות המושחרות בתוך פונקציה InitializeArrays)

```
static void Main(string[] args)
{
    int[] a, b;
    InitializeArrays(a, b);
    for (int i = 0; i < a.Length; i++)
    {
        Console.WriteLine(a[i]);
    }
    for (int i = 0; i < b.Length; i++)
    {
        Console.WriteLine(b[i]);
    }
}

private static void InitializeArrays(int[] a , int[] b)
{
    Random r = new Random();
    a = new int[10];
    b = new int[10];
    for (int i = 0; i < 10; i++)
    {
        a[i] = r.Next(101);
        b[i] = r.Next(101);
    }
}
```

5. הסבר מתי נשתמש בפרמטר בפונקציה המוגדר עם params? (הדגם בקטע קוד)  
**\*אתגר:** מה ההבדל/יתרון בין param לבין שימוש במערך רגיל בפונקציה

6. תקן את קטע הקוד הבא:

(InitializeArrays , PrintArray את לשנות)

```
namespace Simon
{
    public partial class Form1 : Form
    {
        private void Form1_Load(object sender, EventArgs e)
        {
            int[] a;
            a = new int[10];
            InitializeArrays();
            PrintArray();
        }

        private void PrintArray()
        {
            for (int i = 0; i < a.Length; i++)
            {
                Console.WriteLine(a[i]);
            }
        }
    }
}
```

```
private void InitializeArrays()
{
    Random r = new Random();
    a = new int[10];
    for (int i = 0; i < 10; i++)
    {
        a[i] = r.Next(101);
    }
}
}
```

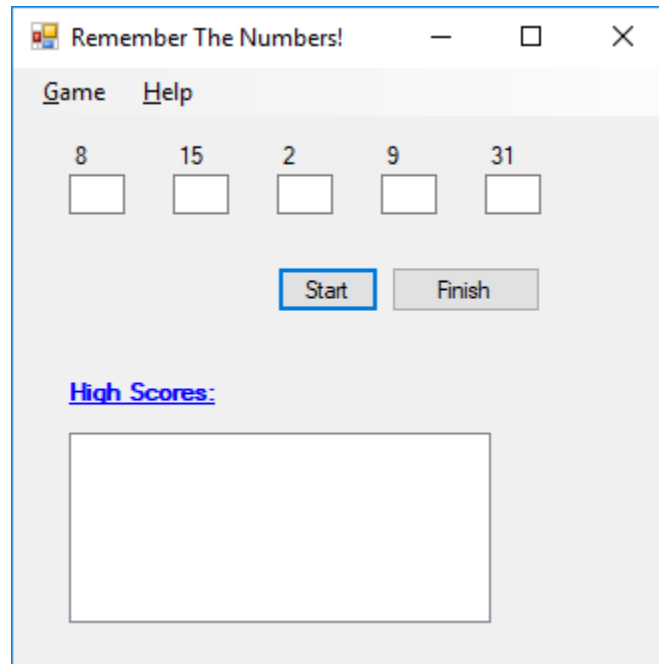
חלק ב' – מערכים (Console Application). ענה לפחות על שאלה אחת מהבאות:

1. כתוב מתודה `int Measure(int[] a, int[] b)`

המתודה מודדת שני מערכים ומחזירה 1 אם המערך הראשון "שוקל" יותר מהשני, מחזירה 0 אם שווים במשקלם ומחזירה -1 אם המערך השני "שוקל" יותר מהראשון.  
כיצד נמדד משקל המערך?  
בודקים איבר איבר, מי יותר גדול. אם לדוגמא התקבלו שני מערכים  $a, b$   
איברי מערך  $a$  הם:  $[8, 7, -1, 4]$   
איברי מערך  $b$  הם:  $[3, 10]$   
עבור איבר  $0: 8 > 3$  ולכן יש ל  $a$  איבר אחד הגדול יותר מ  $b$   
עבור איבר  $1: 7 < 10$  ולכן יש ל  $b$  איבר אחד הגדול יותר מ  $a$   
ל  $a$  יש רק שני איברים ולכן יתר האיברים לא נמדדים.  
בסופו של דבר יצא שהמערכים "שוקלים" משקל זהה מכיוון ש: ל  $a$  יש 1 ול  $b$  יש 1  
כתוב תוכנית המייצרת שני מערכים בגדלים אקראיים `Random`. ומלא את המערכים במספרים אקראיים `Random`. קרא למתודה שכתבת ושלח לה את שני המערכים. לאחר מכן הדפס מי המערך ששוקל יותר (השתדל לחלק את הקוד שכתבת לכמה שיותר מתודות [בתחום ההגיון])

2. כתוב מתודה `int WinX0(int[,] board)`

המתודה מקבלת מטריצה דו-מימדית של מספרים שלמים.  
ראשית בדוק אם גודל המטריצה הוא  $3 \times 3$ . אם לא החזר -1 וצא מהמתודה אח"כ, בדוק אם כל ערכי המטריצה הם בין 0-1, אם לא החזר -2 וצא מהמתודה המטריצה מייצגת מצב מלוח איקס עיגול. כאשר 1 מייצג  $X$  ו-0 מייצג עיגול  
מטרת המתודה היא לבדוק אם אחד השחקנים ניצח (כלומר אם יש שורת  $X$  או טור  $X$  או אלכסון  $X$ ) וכנ"ל לגבי עיגול  
אם קיבלת מצב ששני השחקנים ניצחו (לדוגמא שורה של  $X$  וגם שורה של 0) החזר -3  
אם קיבלת מצב ש  $X$  ניצח החזר 1  
אם קיבלת מצב שעיגול ניצח החזר 2  
אם אף אחד לא ניצח החזר 0  
כעת ב `Main` ייצר מטריצה מאוכלסת במספרים 0 ו-1 ושלח לפונקציה. הדפס למסך את תוצאת המתודה במילים (השתמש ב `switch-case`). לדוגמא אם חזר -1 הדפס: "Matrix size is illegal". אם חזר 1 הדפס: "X Won!" וכך הלאה עבור כל אחד מהאפשרויות של הערכים החוזרים מהמתודה  
**\*אתגר:** ייצר ב `Main` שש מטריצות (אחת לכל מצב) ושלח אותם למתודה שכתבת



כתוב אפליקציית WinForm אשר מציגה למשתמש חמישה Labels. בתוך כל Label הצב מספר אקראי (השתמש ב- Random)

הצב במסמך TextBox מתחת לכל Label (סה"כ חמישה TextBox)

הצב כפתור "Start"

הצב כפתור "Finish"

בעת לחיצה על הכפתור "Start" נעלם הטקסט בכל חמשת ה- Label אפשרויות להעלים את הטקסט (בחר אחת):

1. שנה את המחרוזת שתהיה שווה ל: ""
2. שנה את צבע ה- Label כך שצבע ה- ForeColor יהיה זהה ל Background
3. \*אתגר: שנה את שדה ה- Visible ל False

כעת, המשתמש צריך מזיכורו לשחזר את המספרים שהיו ב- Label לתוך תיבות הטקסט בהתאמה (כלומר המשתמש צריך לכתוב בכל TextBox את המספר שהופיע ב- Label שמעליו) בעת לחיצה על הכפתור "Finish" נבדקים המספרים הכתובים ב- TextBox. אם כולם זהים בדיוק למה שהיה קודם לכן ב- Label שהיה מעליהם אז מופיע MessageBox המכריז על המשתמש כמנצח אחרת מוצג לו שטעה

הוסף תפריט עם כותרת: Game ובתוכו יהיו אופציות:

- New Game. כאשר המשתמש יבחר באופציה זו, יותחל משחק חדש (כלומר ה- Label יחזור, תיבות הטקסט ינוקו, ויוגרלו מספרים חדשים).
- Exit. כאשר המשתמש יבחר באופציה זו, האפליקציה תיסגר (לא לישכוח ליבדוק אם הוא בטוח, לפני הסגירה)

הוסף ListBox המכילה את תוצאות המשחקים. בכל פעם שהמשתמש ילחץ Finish יקבל ניקוד. אם ענה נכון על מספר 1 יקבל נקודה אם על שני מספרים יקבל שתי נקודות (מקסימום 5 נקודות). כלומר כל לחיצה על Finish תוסיף את מספר הנקודות ל- ListBox

**\*אתגר:** אחרי לחיצה על Finish כל מספר שגוי בתיבת הטקסט ייצבע באדום  
**\*אתגר:** נסה לפתור את התרגיל ללא שימוש בכפתור Finish, כלומר שבכל הקלדה נבדק אם המשתמש הקליד את כל המספרים נכון וברגע שכן, מודפסת לו מיד ההודעה (כל עוד לא הזין את כל המספרים או שהזין אבל הם לא 100% נכונים לא מוצגת שום הודעה)  
**\*אתגר:** בדוק כמה זמן עבר מרגע לחיצה על Start ועד ללחיצה על Finish (או לחילופין אם עשית את האתגר, אז עד לרגע השלמת כל המספרים הנכונים). הדפס כמה זמן לקח. אם לקח פחות מ 10 שניות הוסף להודעה כי המשתמש ניצח, אם לא- הצג כי סיים בהצלחה ושינסה שוב כדי לשפר את הזמנים

(2) מדוע באפליקציית WinForm נשתמש ב Debug.Trace?

(3) **\*אתגר:** מדוע במתודות בהם היינו צריכים להדפיס מספר רכיבים **בהשהייה** הוספנו: new Thread

(4) **\*אתגר:**

- בראיון עבודה מספרים לך כי החברה עובדת בדוט נט 2.0.  
אילו מהתשובות עשוי להיות נכון (יכול להיות יותר מ- 1. ויכול גם שלא)
1. החברה עובדת ב WinForms וכידוע לא הוסיפו שיפורים אחרי גרסא 2.0
  2. דוט נט 2.0 היא גרסא חזקה ורוב החברות עדיין משתמשים בה
  3. החברה רוצה להטמיע את מוצריה על לינוקס וכידוע Mono יכול לקמפל רק עד גרסא 2.0
  4. החברה עובדת על ווינדוס 7 ולכן אין אפשרות להשתמש בגרסאת דוט נט מעבר ל 2.0

בהצלחה!