

תאריך: 31.10.25

מבוא למדעי המחשב – כיתה יא-1

דף עבודה בנושא: שימוש חוזר בפונקציות וכתובת תכנית כמו שתכנתים כותבים

מהות התרגיל:

להבין שימוש חוזר בפונקציות, (re-use) לעבוד עם שלושה קבצים באותו פרויקט, ולחבר מספר פונקציות פשוטות לתוכנית שלמה הפועלת יחד.

מטרת התרגיל

בשיעור הזה אנחנו נתחיל לעבוד עם **תוכנית אמיתית** שמורכבת ממספר פונקציות שונות, כל אחת מבצעת משימה קטנה, וכולן יחד מרכיבות מערכת שלמה. המטרה היא שתבינו מה זה אומר "לעבוד עם פונקציות", ומה ההבדל בין פונקציה **שאתם כותבים** – לבין פונקציה שכבר נכתבה ואתם רק צריכים להשתמש בה. נלמד גם איך להשתמש בקבצים שונים באותה תוכנית – בדיוק כמו שעושים מתכנתים אמיתיים: קובץ אחד לפונקציות כלליות (Utils.cs) קובץ שני וגם שלישי לפונקציות שאתם כותבים (LabFuncs.cs) וקובץ ראשי שבו נמצאת Main() כלומר הפונקציה הראשית (Program_V2.cs) (הערה האות V היא קיצור של המילה "גרסה" באנגלית: Version).

למה זה חשוב

- כדי להבין מה זה שימוש חוזר בקוד (reuse).
- כדי להפסיק לשכפל קוד ולהתחיל לחשוב במונחים של פונקציות.
- כדי להבין איך באמת נראית תוכנית אמיתית – לא רק תרגיל קטן.
- ובעיקר – כדי להתחיל לחשוב כתכנתים. זאת קפיצה מעבר ל- "פתרון שאלות".
- וכדי שיהיה לנו מעניין ומהנה!

מבנה הפרויקט

```
ArrayGame_V2/  
├── Program_V2.cs // Main הקובץ הראשי שמכיל את הפונקציה  
├── Utils.cs // פונקציות עזר מוכנות (לא לשינוי)  
├── Lab.V1_Funcs.cs // (V1 - פונקציות בסיסיות) חלק ראשון של המחלקה  
└── Lab.V2_Funcs.cs // חלק שני ArrayGame_V1/
```

הוראות עבודה

1. פתחו פרויקט חדש ב- Visual Studio או ב- VS Code (Console App).
2. הוסיפו לתוך הפרויקט את הקבצים שהוזכרו למעלה (מבנה הפרוייקט).
הקבצים האלו נמצאים במודל או ב GDRIVE. תוכלו להגיע אליהם בקלות דרך הקישור הבא:
[/https://tzvimelamed.com/lab](https://tzvimelamed.com/lab)
3. הריצו את התוכנית. אתם אמורים לראות פלט שמודיע אילו חלקים עדיין לא ממומשים.

החומר נועד לשימוש אישי של תלמידי תיכון קריית שרת בלבד.
אין להעתיק, לשכפל או להפיץ ללא רשות 1

4. עברו על ההערות בקבצים Lab.V1_Funcs.cs and Lab.V2_Funcs.cs. כל הערה מתארת פונקציה שעליכם לממש.
5. אל תשנו את הקובץ Utils.cs. הוא מדמה "ספרייה קיימת".

חלק א – חימום (V1)

- קלט מהמשתמש: גודל מערך, תחום מינימום/מקסימום. (Utils.GetUserInput)
- יצירת מערך אקראי (CreateRandomArray) בעזרת Utils.RandInRange.
- הדפסה רגילה של המערך. (PrintArraySimple)

חלק ב – הרחבה וחשיבה (V2)

- ממשו את הפונקציות הבאות בקובץ StudentFuncs.cs. מומלץ לפי הסדר.
1. SumArray(int[] arr) מחזירה את סכום איברי המערך.
 2. double AverageArray(int[] arr) – משתמשת ב־ SumArray כדי להחזיר ממוצע.
 3. CountEven(int[] arr) – סופרת כמה איברים זוגיים במערך.
 4. FindMax(int[] arr) – מחזירה את הערך המקסימלי במערך.
 5. FindMin(int[] arr) – מחזירה את הערך המינימלי במערך.
 6. PrintGreaterThan(int[] arr, double limit) – מדפיסה איברים הגדולים מערך limit כלשהו (למשל ממצע).

חלק ג – אתגר (אופציונלי)

ColorizeArrayByChoice(int[] arr):

שאלו את המשתמש מצב צביעה:

- 1 - זוגיים, 2 - ראשוניים (Green), 3 - מתחלקים ב־ 3 (Yellow)
- הדפסו את המערך בעזרת Utils.PrintColored. מותר להשתמש ב־ Utils.IsPrime שנתונה לכם

תזכורות טכניות

- כל הפונקציות static.
- לא משכפלים קוד: מזהים חזרות והופכים אותן לפונקציות.
- שמרו את כל הקבצים באותה תיקייה של הפרויקט.
- פלט לקונסול – באנגלית בלבד.