

תאריך: 26.2.2026

## מבוא למדעי המחשב – כיתה יא-1

### תכנות מונחה עצמים – ריבוי צורות (Polymorphism)

#### תירגול #3 – חלק ב' – משימת מעבדה / תירגול

#### תירגול - מעבדה #3: פולימורפיזם (פרויקט "עולם הצורות")

**הערה:** לקורס מצורפים קבצי בסיס הנמצאים ב-GDRIVE- יש לעבוד על בסיס הקבצים שניתנו. אין לשנות שמות מחלקות או חתימות של פעולות קיימות אלא אם נאמר במפורש.

#### מטרות המעבדה

1. מימוש ירושה רב-שכבתית (Base → Derived → SubDerived).
2. שימוש ב-virtual ו-override למימוש פולימורפיזם.
3. עבודה עם מערך ותור של טיפוס בסיס.
4. שימוש במחלקת Canvas להצגה טקסטואלית.

#### תחנה 1 – ירושה בסיסית והדפסת אובייקטים

##### מטרה:

לבנות היררכיה בסיסית תקינה של צורות ולהתרגל לשימוש ב-base ו- ToString.

בקבצי הבסיס קיימת מחלקה בשם Shape.

עליך להשלים את המחלקות הבאות:

1. כתוב מחלקה בשם Rectangle היורשת מ-Shape.
  - הוסף שדות פרטיים: height, width.
  - כתוב בנאי המקבל name, width, height.
  - הקפד לקרוא ל- base(name).
  - דרוס את הפעולה ToString כך שתציג את שם הצורה ואת המידות שלה.
2. כתוב מחלקה בשם Triangle היורשת מ-Shape.
  - הוסף שדות פרטיים: height, baseLength.
  - כתוב בנאי מתאים.
  - דרוס את הפעולה ToString כך שתציג את הנתונים של המשולש.

לאחר מימוש המחלקות:

- כתוב פונקציה בשם Main\_1.
- בתוך הפונקציה צור לפחות אובייקט אחד מסוג Rectangle ואובייקט אחד מסוג Triangle.

- הדפס כל אובייקט באמצעות Console.WriteLine.

ודא שההדפסה משתמשת ב- ToString שדרסת ולא במימוש של מחלקת הבסיס.

החומר נועד לשימוש אישי של תלמידי תיכון קריית שרת בלבד.

אין להעתיק, לשכפל או להפיץ ללא רשות 1

## תחנה 2 – המהפך הפולימורפי: חישוב שטח

מטרה:

לממש פולימורפיזם באמצעות `virtual` ו-`override` ולהבין את משמעות הבחירה **בזמן ריצה**.  
עליך לבצע את המשימות הבאות:

1. פתח את מחלקת `Shape` והוסף את הפעולה הבאה:

```
public virtual double GetArea()  
{  
    return 0;  
}
```

2. במחלקה `Rectangle` דרוס את הפעולה `GetArea` כך שתחזיר את שטח המלבן.

3. במחלקה `Triangle` דרוס את הפעולה `GetArea` כך שתחזיר את שטח המשולש.

לאחר מכן:

4. כתוב פונקציה בשם `Main_2`.

5. בתוך הפונקציה:

- צור מערך מסוג `Shape[]` בגודל 3.
- הכנס למערך אובייקטים מסוגים שונים (לפחות מלבן ומשולש).
- עבור על המערך בלולאה.
- עבור כל איבר הדפס:

▪ את `ToString` של האובייקט

▪ את השטח שלו באמצעות קריאה ל-`GetArea()`

ודא שהקוד אינו בודק את סוג האובייקט באמצעות `if` (או `is`).

שאלת חשיבה:

מה יקרה אם תסיר את המילה `virtual` מהפעולה `GetArea` במחלקת `Shape`? נמק.

מה התפקיד של המילה הזאת?

## תחנה 3 – ירושה רב-שכבתית : `Square`

מטרה:

להבין ירושה מדרגה שנייה ולחזק את עקרון `IS-A` בהיררכיה רב-שכבתית.

עליך לבצע את המשימות הבאות:

1. כתוב מחלקה בשם `Square` היורשת מ-`Rectangle`.

2. למחלקה `Square` יהיה בנאי המקבל:

○ `name`

○ `side`

3. בתוך הבנאי השתמש ב-`base(name, side, side)` כך שאורך הצלע ישמש גם כרוחב וגם כגובה.

החומר נועד לשימוש אישי של תלמידי תיכון קריית שרת בלבד.

4. (רשות) דרוס את הפעולה ToString כך שיופיע במפורש שמדובר ב-Square.

לאחר מכן:

5. פתח את הפונקציה Main\_2 או כתוב פונקציה בשם Main\_3.

6. הוסף למערך מסוג Shape[] גם אובייקט מסוג Square.

7. הרץ את התוכנית. וודא:

○ שהאובייקט מסוג Square מודפס כראוי.

○ שהשטח מחושב נכון.

○ שלא נדרש שינוי בלולאה הקיימת.

שאלת חשיבה:

מדוע אין צורך לשנות את הקוד בלולאה כאשר מוסיפים מחלקה חדשה היורשת מ?Shape-

להלן דוגמא לפלט אפשרי של התכנית:

```
=== World Of Shapes ===

---- Main_1 : Basic Inheritance ----
Rectangle: Rect1 Width=4 Height=5
Triangle: Tri1 Base=6 Height=3

---- Main_2 : Polymorphism ----
Rectangle: Rect2 Width=3 Height=7
Area = 21

Triangle: Tri2 Base=8 Height=4
Area = 16

Rectangle: Rect3 Width=2 Height=9
Area = 18

---- Main_3 : Square (Multi-level Inheritance) ----
Rectangle: Rect4 Width=2 Height=6
Area = 12

Triangle: Tri3 Base=5 Height=4
Area = 10

Square: Square1 Side=5
Area = 25

Program finished.
```

#### תחנה 4 – עבודה עם Canvas

(יתווסף בהמשך, בלי נדר... צבי 27.2.26)

מטרה:

להשתמש במחלקה קיימת ולשלב צורות בהקשר ויזואלי, תוך חשיבה על שימוש חוזר בקוד.

בקבצי הבסיס מצורפת מחלקה בשם Canvas.

עליך לבצע את המשימות הבאות:

1. פתח את פונקציית Main.
2. צור אובייקט מסוג Canvas בגודל 20 X 20.
3. צור אובייקט מסוג Rectangle והוסף אותו ללוח באמצעות הפעולה AddRectangle.
4. קרא לפעולה DrawCanvas והרץ את התוכנית כדי לראות את התוצאה.

לאחר מכן:

5. הוסף למחלקת Canvas פעולה חדשה בשם:

AddSquare(int row, int col, Square s)

6. בתוך הפעולה החדשה, בדוק האם ניתן להשתמש בקוד שכבר קיים עבור מלבן במקום לכתוב קוד חדש לחלוטין.

7. בדוק שהוספת ריבוע ללוח פועלת בצורה תקינה.

שאלת חשיבה:

כאשר Square יורש מ Rectangle- האם ניתן להתייחס אליו כאל מלבן במחלקת Canvas? נמק.