

## שאלות מתכונת ליחידת תכנות מונחה עצמים

ניתן להשתמש בחומרים לצורך הוראה בלבד  
לא ניתן לפרסם את החומרים או לעשות בהם כל שימוש מסחרי

### היררכית מחלקות, ירושה והכלה (גיטה קופרמן וטובי סטפ)

לפניך פרויקט המכיל ארבע מחלקות A , B , C , TestAbc :

```
public class A
{
    protected int x;

    public A (int x)
    {
        this.x = x;
    }
    public virtual void Change()
    {
        x= x*2;
    }
    public override string ToString()
    {
        Change();
        return "X :" + this.x;
    }
}
public class B : A
{
    protected int y;

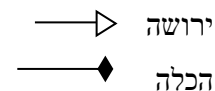
    public B(int x, int y)
        : base(x)
    {
        this.y = y;
    }
    public void Print()
    {
        Console.WriteLine("B-B-B");
    }
    public override string ToString()
    {
        return base.ToString() + " Y :" + this.y;
    }
}
public class C : B
```

```

{
    private int z;
    public C (int x, int y, int z)
        : base(x, y)
    {
        this.z = z;
    }
    public override void Change()
    {
        base.Change();
        z=z *2;
    }
    public override string ToString()
    {
        return base.ToString() + " Z :" + this.z;
    }
}
class TestAbc
{
    static void Main(string[] args)
    {
        A[] obj = new A[4];
        obj[0] = new A(5);
        obj[1] = new B(5, 10);
        obj[2] = new C(5, 10, 15);
        obj[3] = obj[0];
        for (int i = 0; i < obj.Length; i++)
            Console.WriteLine(obj[i]);
    }
}

```

א. סרטט את היררכית המחלקות המוגדרות בפרויקט. השתמש בסימנים אלה:



ב. ערוך מעקב אחרי הפעולה Main שבמחלקה TestAbc במעקב יש לכלול את מצב הזיכרון

בעבור כל עצם ולהציג את פלט התכנית.

ג. לפניך ארבע הוראות i – iv לכל אחת מהן, קבע אם אפשר להוסיף אותה לפעולה Main. אם כן

הסבר באמצעות עקרונות תכנות מונחה עצמים. אם לא הסבר מדוע.

i. Object sod = new A(20);

ii. B a = new A(5);

iii. B c = new C(2, 4, 6);

iv. A a = new C(1,2,3);

ד. האם אפשר לכתוב את שורות הקוד הבאות בתוך הפעולה Main מחלקה TestAbc

```
Object a = new A(30) ;
```

```
((A)a).change() ;
```

אם אפשר, הצג את הפלט. אחרת הסבר מדוע.

ה. האם אפשר לכתוב את שורות הקוד הבאות בתוך הפעולה Main במחלקה TestAbc

```
Object c = new C (11,22,33) ;
```

```
((B)c).print() ;
```

אם אפשר, הצג את הפלט. אחרת הסבר מדוע.

ו. האם אפשר לכתוב את שורות הקוד הבאות בתוך הפעולה Main במחלקה TestAbc Object

```
b = new B(11,12);
```

```
b.print() ;
```

אם אפשר, הצג את הפלט. אחרת הסבר מדוע.

## היררכית מחלקות (גיטה קופרמן)

באוניברסיטה העברית מוגדר מאגר סטודנטים באופן הבא:  
לכל סטודנט יש פרטים אישיים: שם `string name` ומספר זהות `string id` וכן סכום שכר הלימוד השנתי שעליו לשלם.  
סטודנט יכול להיות אחד מבין שלושה סוגים: סטודנט רגיל, סטודנט שמקבל מילגת לימודים וסטודנט שהוא בן משפחה של עובד אוניברסיטה.

יש להגדיר את המחלקות הבאות:

**Student** המחלקה מייצגת סטודנט רגיל.

תכונות המחלקה: `string id, string name`

פעולות: פעולה בונה המקבלת כפרמטר את ערכי כל התכונות

פעולה בונה מעתיקה

פעולה המחזירה את פרטי הסטודנט + סוג הסטודנט (**RegularStudent**)

פעולה בשם `Tuition` המחזירה את שכר הלימוד השנתי שהסטודנט משלם.

סטודנט רגיל משלם 17,000 ש"ח לשנה.

**ScholarshipStudent** המחלקה מייצגת סטודנט שמקבל מלגת לימודים.

תכונות המחלקה: `int sum, string id, string name` (sum הוא סכום המלגה שהסטודנט מקבל)

פעולות: פעולה בונה המקבלת כפרמטר את ערכי כל התכונות

פעולה בונה מעתיקה

פעולה המחזירה את פרטי הסטודנט + סוג הסטודנט (**ScholarshipStudent**)

פעולה בשם `Tuition` המחזירה את שכר הלימוד השנתי שהסטודנט משלם. מ

חושב לפי שכר לימוד שנתי רגיל פחות סכום המלגה.

**RelativeStudent** המחלקה מייצגת סטודנט בן משפחה של עובד.

תכונות המחלקה: `int degree, string id, string name` (degree יכול להיות 1,2 או 3 מצייין

את דרגת העובד)

פעולות:

פעולה בונה המקבלת כפרמטר את ערכי כל התכונות

פעולה בונה מעתיקה

פעולה המחזירה את פרטי הסטודנט + סוג הסטודנט (**RelativeStudent**)

פעולה בשם `Tuition` המחזירה את שכר הלימוד השנתי שהסטודנט משלם. בן של עובד מקבל

הנחה בשכר הלימוד בהתאם לדרגת העובד (ערך התכונה degree): 1 – מקבל 50% הנחה, 2- מקבל 75% הנחה ו-3 לא משלם כלל

- א. בנה תרשים מחלקות מתאים ותאר את היחסים שבין המחלקות.
- ב. ממש בסביבת עבודה את המחלקות. לכל מחלקה יש להגדיר את התכונות המתאימות ואת הפעולות הבאות(כמפורט לעיל): פעולה בונה, בונה מעתיקה, חישוב שכר לימוד ופעולה

`ToString`

ניתן להניח שלכל תכונה מוגדרות פעולות `Get/Set` ואין צורך לממש אותן. אפשר להשתמש בהן לפי הצורך.

- ג. עליך להגדיר מחלקה נוספת בשם `University` המנהלת את מאגר הסטודנטים. למחלקה זו שתי תכונות: מערך סטודנטים (יכול להכיל עד 500 סטודנטים) ו- `int numOfStudents` המציין את מספר הסטודנטים בפועל.

המחלקה תכיל את הפעולות הבאות:

פעולה בונה המייצרת מאגר סטודנטים ריק.

פעולה בשם `AddStudent` המקבלת הפניה לסטודנט ומוסיפה אותו למאגר אם מספר הזהות שלו אינו נמצא במאגר, אחרת אינה מבצעת דבר.

פעולה בשם `TuitionSum` המחזירה את הסכום הכולל שהתקבל משכר הלימוד של הסטודנטים במאגר.

פעולה בשם `StudentList` המדפיסה לכל סטודנט, את הפרטים שלו ואת גובה שכר הלימוד שמשלם.

## היררכית מחלקות (גיטה קופרמן ועינת לוי)

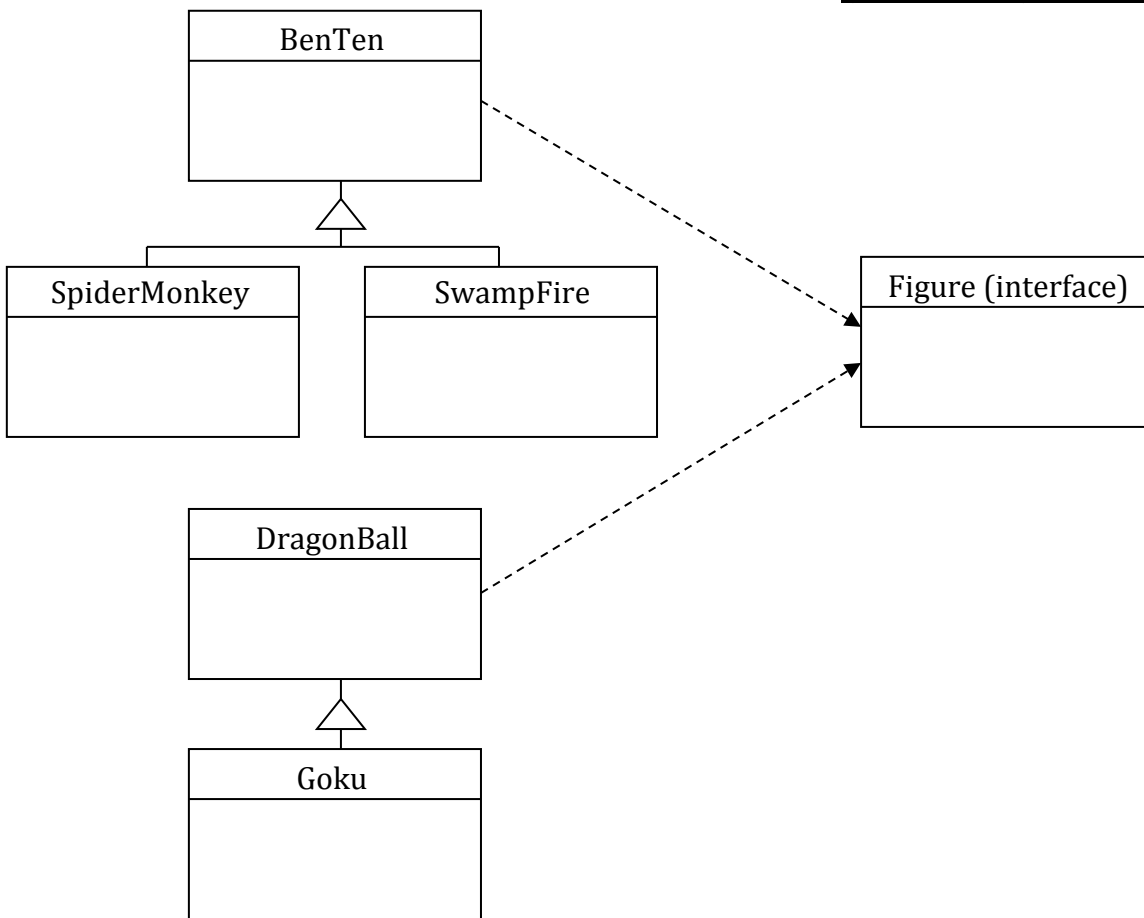
נתונים הטיפוסים הבאים:

```
public class AA
{
    private int x;
    private int y;
    public AA()
        { this.x=0;
          this.y=0; }
    public void SetX(int x)
        { this.x=x;}
    public void SetY(int y)
        {this.y=y;}
    public int GetX()
        { return this.x;}
    public int GetY()
        { return this.Y;}
    public int Sum()
        { return this.x+this.y;}
    public override string ToString()
        { return "X="+this.x+" Y="+this.y;}
}
public class BB
{
    private AA aa;
    private int z;
    int z, AA aa) public BB)
    { } // השלם 1 ??
    int x,int y,int z) public BB)
    } // השלם 2{// ???
    (public void SetZ (int z
    { ;this.z = z }
    (public void SetAa (AA aa
    {} // השלם 3// ??
    () public int GetZ
    { return this.z }
    public int() GetAa
    {return this.aa; }
    public int Sum ()
    { return this.aa.Sum()+this.z; }
    public override string ToString()
    { return this.aa.ToString() + " z="+this.z + " sum="+this.sum();}
```

}

- א. הוסף פעולה בונה נוספת למחלקה AA.
- ב. השלם את הפעולות החסרות במחלקה BB.
- הערה: יש להימנע כאשר זה אפשרי, מליצור הפניות רבות לעצם אחד.
- ג. בדוק את הפעולה GetAa, האם היא תקינה? אם כן הסבר מהי מבצעת ואם לא תקן אותה
- ד. מהו הקשר בין הטיפוסים? לווה את תשובתך בשרטוט מתאים.
- ה. בנה תכנית ראשית הבונה שני טיפוסים מסוג AA ושני טיפוסים מסוג BB התכנית מסכמת את אברי הטיפוסים ומציגה את התוצאה. יש להפעיל שני בנאים שונים בבניית כל אחד מהטיפוסים.

## היררכית מחלקות (גיטה קופרמן)



### לפניך תרשים UML

- א. רשום את כותרת כל אחת מהמחלקות בתרשים.
- ב. רשום קטע תוכנית היוצר מערך בגודל 5 בשם `figures` המאכלס מופע של כל אחת מהמחלקות הנ"ל. כלומר בכל תא במערך תהיה דמות שונה.  
הנח : כי לכל מחלקה יש פעולה בונה ברירת מחדל
- ג. בממשק **Figure** קיימת כותרת הפעולה `int MyPower()`, ציין באילו מחלקות יש לממש את הפעולה הנ"ל.
- ד. כתוב הוראה או הוראות בשפת **C#** המזמנות את הפעולה `myPower` על התא במערך שיצרת שמכיל את העצם **SpiderMonkey**
- ה. במחלקה **Goku** קיימת הפעולה `int GoldenBalls()`  
כתוב הוראה או הוראות בשפת **C#** המזמנות את הפעולה הנ"ל על התא במערך המכיל את העצם שיצרת מהמחלקה **Goku**
- ו. ליד כל אחת מההוראות הבאות ציין אם היא חוקית או לא, נמק את החלטתך.

1. `Figure f = new Figure();`



```
Figure f = new Goku() ; .2  
f.GoldenBalls() ;
```

```
BenTen f1 = new BenTen() ; .3  
f1.MyPower() ;
```

```
for(int i=0 ; i<this.figures.length ; i++) .4  
figure[i].MyPower() ;
```

## מעקב עם תרשים עצמים (אוי גרינולד, אלה לב)

לפניך פרויקט עם המחלקות **One**, **Two**, **MainOneTwo**

כתוב מעקב אחר הפעולה main במחלקה **MainOneTwo** וכתוב את הפלט.

במעקב יש לכתוב את ערכי המשתנים, ובעבור כל עצם, את ערכי התכונות שלו.

<pre>public class <b>One</b> {     private double x,y;     public One(double x, double y)     {         this.x=x;         this.y=y;     }     public One(One one)     {         this.x=one.x;         this.y=one.y;     }     public void setX(double x)     {         this.x=x;     }     public String toString()     {         return "&lt;" + x + "," + y + "&gt;";     } }</pre>	<pre>public class <b>Two</b> {     private One arr[];      public Two(One a[])     {         this.arr=a;     }     public Two(One a, One b)     {         this.arr=new One[4];         this.arr[0]=a;         this.arr[1]=new One(a);         this.arr[2]=b;         this.arr[3]=new One(b);     }     public String toString()     {         String str="";         for (int i=0; i&lt; this.arr.length; i++)             str= str + "arr[" + i + "]: " + this.arr[i].toString() + "\n";         return str;     } }</pre>
---	---

המחלקה הראשית – בעמוד הבא

```
public class MainOneTwo
{
    public static void main(String[] args)
    {
        One f1=new One(1.2, 3.4);
        One f2=new One(5.5, 7.8);
        One f3=f2;
        f2=f1;
        f1=f3;
        f1.setX(9.7);
        System.out.println(f1+" "+f2+" "+f3);

        One mar[]=new One[2];
        mar[0]=f1;
        mar[1]=f2;

        Two sec1=new Two(mar);
        System.out.println(sec1);
        Two sec2=new Two(f1,f2);
        System.out.println(sec2);

        f2.setX(20.5);
        System.out.println(sec1);
        System.out.println(sec2);
    }
}
```

## ירושה, דריסות, המרות, הפניות (אוי גרינולד)

קרא את קוד המחלקות שלפניך וענה על כל הסעיפים:

- א- שרטט תרשים המתאר את יחסי המחלקות. אין צורך לפרט את איברי המחלקות.
- ב- עבור כל אחת מהשורות הממוספרות בהמשך (5) - (1) כתוב אם היא תגרום לשגיאת הידור או לא. נמק תשובתך.
- ג- תקן את כל השגיאות שציינת בסעיף ב.
- ד- עקוב אחר הפעולה הראשית במחלקה CarShop ורשום את הפלט.
- יש ללוות את המעקב בתרשים עצמים. לכל עצם יש לרשום את ערכי התכונות שלו.
- ה- איך היה משתנה פלט התכנית אם במחלקה Car הגדרת התכונה של הצבע תהיה:

```
private static String color = "WHITE";
```

```
public class Car
{
    private String manufacturer;    // שם יצרן
    private int speed;              // מהירות
    private String color;           // צבע מכונית
    public static final int SPEED_UP = 5;    // מידת הגדלת מהירות

    public Car(String manufacturer, int speed)
    {
        this.manufacturer= manufacturer;
        this.speed= speed;
        this.color= "WHITE";
    }
    public void changeColor(String newColor)
    {
        this.color=newColor;
    }
    public void speedUp()
    {
        if( this.speed + SPEED_UP <=120 )
            this.speed += SPEED_UP;
    }
    public String getManufacturer()
    {
        return manufacturer;
    }
    public String getColor()
    {
        return color;
    }
}
```

```
public String toString()
{
    String car="Car: " + this.manufacturer + " , " + this.color + " , " + this.speed;
    return car;
}
}
```

```
public class TwoTruck extends Car
{
    private Car carToTow; // car to drawn with
    public TwoTruck(String manufacturer, int speed,Car car)
    {
        super(manufacturer, speed);
        this.carToTow = carToTow;
    }
    public String toString()
    {
        String twoTruck = "Two truck:";
        twoTruck += this.getManufacturer()+","; // (1)
        twoTruck += this.color + ","+this.speed; // (2)
        return twoTruck;
    }
}
```

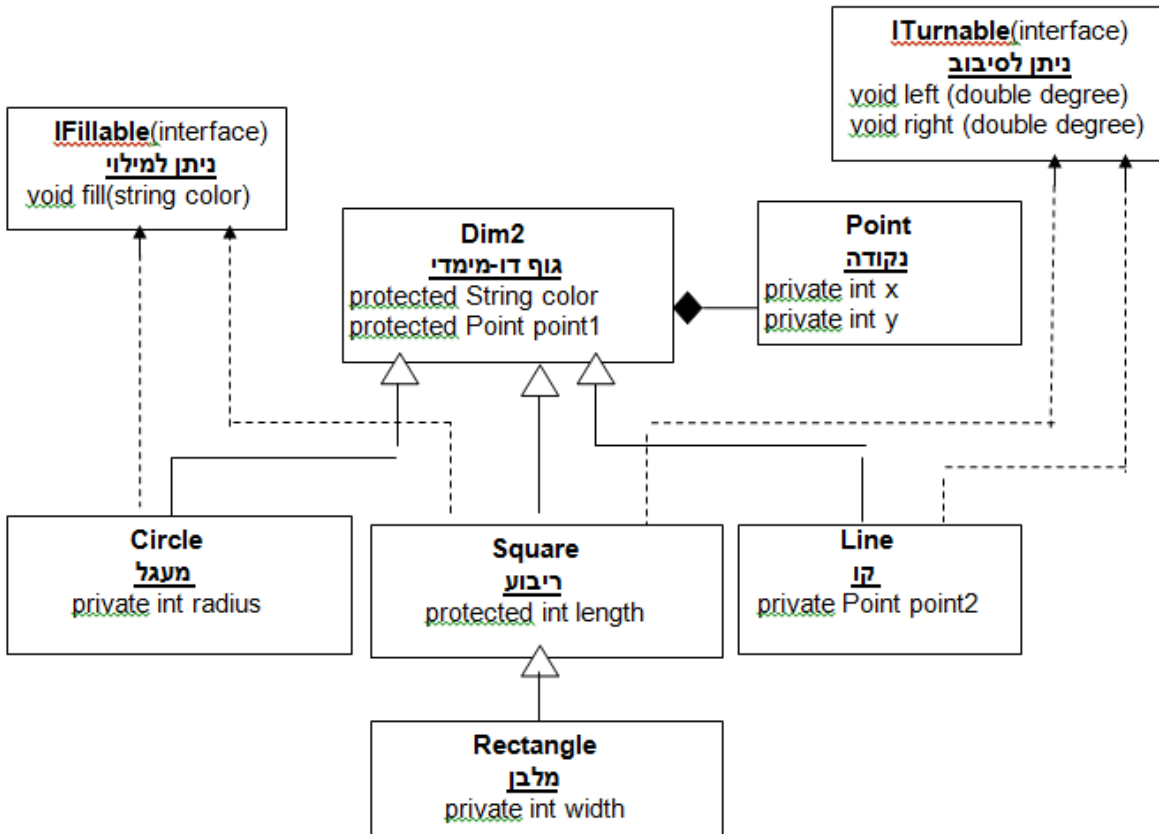
```
public class CarShop
{
    public static void main(String[] args)
    {
        Car[] ourCars = new Car[4];
        ourCars[0] = new Car("Ford",55);
        ourCars[1] = new SportCar("Ferari", 80, true);
        ourCars[2] = ourCars[1]; // (3)
        ourCars[3] = new TwoTruck("Mizubishi", 80, null);
        SportCar myCar = ourCars[1]; // (4)
        for (int c=0; c<ourCars.length; c++)
            ourCars[c].speedUp();
        ourCars[2].changeColor("BLUE"); // (5)
        for (int c=0; c<ourCars.length; c++)
            System.out.println(ourCars[c]);
    }
}
```

```
public class SportCar extends Car
{
    private boolean hasTurbo;
```

```
public SportCar(String manufacturer, int speed, boolean hasTurbo)
{
    super(manufacturer, speed);
    this.hasTurbo = hasTurbo;
}
public void speedUp()
{
    super.speedUp();
    super.speedUp();
    super.speedUp();
}
public String toString()
{
    String sportCar="Sport"+super.toString()+",Turbo:"+this.hasTurbo;
    return sportCar;
}
}
```

## ממשקים (אוי גרינולד)

חברת XYZ מפתחת תוכנה המאפשרת להגדיר גופים דו-ממדיים, ולבצע עליהן פעולות שונות. להלן תרשים UML חלקי של המערכת.



הנח כי לכל מחלקה יש פעולה בונה המקבלת כפרמטר את כל התכונות שלה וכי קיימות על הפעולות המאחזרות, get והקובעות, set לכל התכונות של המחלקה.

א. עבור כל אחד מחמש המחלקות שבתרשים כתוב את :

- i. כותרת המחלקה
- ii. חתימת כל הפעולות **שחייבות** להופיע במחלקה. הימנע מכתובת דברים מיותרים.

- ב. i. ציין לגבי כל אחת מעשרת השורות המסומנות בקוד שלפניך קבע האם ההוראה תקינה או שיש שגיאת תחביר או שגיאת ריצה. נמק תשובתך.
- ii. בכל שורה בה מתבצעת המרה ציין איזה סוג המרה מתבצעת.

```
public static void main(String[] args)
```

```
{
```

```
    Dim2 d2=new Square("red",new Point(50,50),10);
```

```
    Dim2 c=new Circle("blue",new Point(50,50),8);
```

```
Line line = new Line("green",new Point(100,100),new Point(200,200));
IFillable fillA, fillB, fillC;
ITurnable turnA;
fillA=new IFillable();           // (1)
fillB = d2;                       // (2)
turnA=line;                       // (3)
Rectangle r = new Rectangle("blue", new Point(2,10), 20, 15);
fillC = r;                         // (4)
turnA.setColor("green");          // (5)
fillC.fill("red");                // (6)
turnA.left (90);                  // (7)
((Square)c).left (Math.PI/4);     // (8)
Dim2 d=c;                          // (9)
d.fill("Yellow");                 // (10)
}
```