

## Exercicis Pas Paràmetres - IIP Primer Parcial

1. 2 punts Es disposa de la classe `Point` que defineix un punt en l'espai bidimensional real (amb dos atributs que representen la seua abscissa i la seua ordenada), amb la funcionalitat que es mostra en part, a continuació, en la seua documentació:

Constructors	
Constructor and Description	
<code>Point(double px, double py)</code> Crea un <code>Point</code> amb abscissa <code>px</code> i ordenada <code>py</code> .	

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method and Description	
double	<code>getX()</code>	Torna l'abscissa del <code>Point</code> <code>this</code> .
double	<code>getY()</code>	Torna l'ordenada del <code>Point</code> <code>this</code> .
void	<code>setX(double px)</code>	Actualitza l'abscissa del <code>Point</code> <code>this</code> a <code>px</code> .
void	<code>setY(double py)</code>	Actualitza l'ordenada del <code>Point</code> <code>this</code> a <code>py</code> .
java.lang.String	<code>toString()</code>	Torna un <code>String</code> que representa el <code>Point</code> <code>this</code> en el format típic matemàtic, i.e., (abscissa,ordenada).

Donat el següent programa Java:

```
public class Exercici1 {
    private Exercici1() { }

    public static void main(String[] args) {
        Point p = new Point(1.0, -1.0);
        double x = p.getX();
        double y = p.getY();

        System.out.print("Abans d'invocar a canviaCoords: ");
        System.out.println("x = " + x + ", y = " + y + ", p = " + p.toString());

        canviaCoords(x, y, p);
        System.out.print("Després de la primera crida a canviaCoords: ");
        System.out.println("x = " + x + ", y = " + y + ", p = " + p.toString());

        x = p.getY();
        y = p.getX();
        canviaCoords(x, y, p);
        System.out.print("Després de la segona crida a canviaCoords: ");
        System.out.println("x = " + x + ", y = " + y + ", p = " + p.toString());
    }

    public static void canviaCoords(double x, double y, Point p) {
        double z = x; x = y; y = z;
        p.setX(x);
        p.setY(y);
    }
}
```

**Es demana:** Completar què es mostra per pantalla després de executar-lo.

Abans d'invocar a `canviaCoords`: x = \_\_\_\_\_, y = \_\_\_\_\_, p = (\_\_\_\_\_, \_\_\_\_\_)

Després de la primera crida a `canviaCoords`: x = \_\_\_\_\_, y = \_\_\_\_\_, p = (\_\_\_\_\_, \_\_\_\_\_)

Després de la segona crida a `canviaCoords`: x = \_\_\_\_\_, y = \_\_\_\_\_, p = (\_\_\_\_\_, \_\_\_\_\_)

2. 2 punts Es disposa de la classe **Punt** que defineix un punt en un espai bidimensional real (amb dos atributs representant la seua abscissa i la seua ordenada), amb la funcionalitat que es mostra en part, a continuació, en la seua documentació:

## Constructor Summary

Constructors
Constructor and Description
<b>Punt</b> (double abs, double ord) Crea un Punt amb coordenades (abs, ord).

## Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method and Description	
void	<b>setX</b> (double nou) Actualitza a nou el valor de l'abscissa del Punt en curs.	
void	<b>setY</b> (double nou) Actualitza a nou el valor de l'ordenada del Punt en curs.	
java.lang.String	<b>toString</b> () Torna un String amb les dades del Punt en curs en el format (x, y).	

Donada la següent classe programa:

```
public class Exercici2 {
    private Exercici2() { }

    public static void main(String[] args) {
        Punt p = new Punt(1.0, -1.0);
        double x = 1.0, y = -1.0;
        System.out.println("Abans de canviarCoord: x = " + x + ", y = " + y + ", p = " + p);

        canviarCoord(x, y, p);
        System.out.println("Després de canviarCoord una vegada: x = " + x + ", y = " + y + ", p = " + p);

        canviarCoord(x, y, p);
        System.out.println("Després de canviarCoord 2 vegades: x = " + x + ", y = " + y + ", p = " + p);
    }

    public static void canviarCoord(double x, double y, Punt p) {
        p.setX(y);
        p.setY(x);
    }
}
```

**Es demana:** Completar què es mostra per pantalla després de la seua execució.

Abans de canviarCoord: x = \_\_\_\_\_, y = \_\_\_\_\_, p = (\_\_\_\_\_, \_\_\_\_\_)

Després de canviarCoord una vegada: x = \_\_\_\_\_, y = \_\_\_\_\_, p = (\_\_\_\_\_, \_\_\_\_\_)

Després de canviarCoord 2 vegades: x = \_\_\_\_\_, y = \_\_\_\_\_, p = (\_\_\_\_\_, \_\_\_\_\_)