

■ Problema #1

Escriba la clase Grafico que extienda de Canvas y que soporte graficar funciones en \mathbb{R}^2 . Para ello, suponga que tiene discretizados los valores de la funcion en los puntos $\{(x_k, y_k)\}_{k=0}^{n-1} \subseteq \mathbb{R}^2$, y con ellos traze lineas rectas entre puntos adyacentes. Se le sugiere la siguiente estructura de clase y que redefina el metodo **paint(Graphics g)**

```
class Grafico extends Canvas{
    // variables de instancia...
    public grafica(int[] x, int[], xmin, xmax, ymin, ymax){...}
    public void zoom(double factor){... propuesto ...}
    public void paint(Graphics g){...}
}
```

■ Problema #2

Las piezas del ajedrez pueden ser modeladas por una clase abstracta, que tiene todas las operaciones básicas de las piezas en general, y por subclases que especifican exactamente que comportamiento particular tiene la pieza.

Ejemplo	Descripcion	Encabezamiento
Pieza p = new Peon(2,5);	Crea un peón en la fila 2 columna 5	Peon(int x, int y)
Pieza t = new Torre(2,5);	Crea una torre en la fila 2 columna 5	Torre(int x, int y)
Pieza a = new Alfil(2,5);	Crea un alfil en la fila 2 columna 5	Alfil(int x, int y)
Pieza c = new Caballo(2,5);	Crea un caballo en la fila 2 columna 5	Caballo(int x, int y)
int f = p.getFila();	Retorna la fila donde se encuentra la pieza p	int getFila()
int c = a.getCol();	Retorna la columna donde se encuentra la pieza a	int getCol()
if(c.amenaza(2,4)){...}	Determina si la pieza c amenaza la posicion del tablero	boolean amenaza(int x, int y)

Se le pide implementar las Clases Pieza, Torre y Caballo.

■ Problema #3

Sea una matriz mágica o semi-mágica $M \in M_{n,n}(\mathbb{R})$, por ejemplo:

$$M_1 = \begin{pmatrix} 16 & 3 & 2 & 13 \\ 5 & 10 & 11 & 8 \\ 9 & 6 & 7 & 12 \\ 4 & 15 & 14 & 1 \end{pmatrix} \quad M_2 = \begin{pmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{pmatrix} \quad M_3 = \begin{pmatrix} -2 & 1 & 1 \\ 1 & -2 & 1 \\ 1 & 1 & -2 \end{pmatrix}$$

Descubra su propiedad e implemente una función que diga si una matriz es mágica o semi-mágica.

Propuesto: Escriba un programa que entregue una matriz mágica $M_{n,n} \in (\mathbb{N})$