

INSTRUCCIONES

- El alumno debe entregar una carpeta con las soluciones al examen cuyo nombre debe estar formado por el número de lista seguido de las iniciales. Por ejemplo, Facundo Romuedo Piladro que es el número 8 de la lista entregaría una carpeta con nombre **Ex08frp**.
- Los ficheros o carpetas correspondientes a las soluciones se deben nombrar igual que la carpeta junto con el número del ejercicio, por ejemplo **Ex08frp1.java**, **Ex08frp2.java**, etc.
- En los comentarios de cada programa se debe indicar el nombre completo, la fecha y el turno.

EJERCICIOS

1. Realiza la función **sumatorioDeDigitos** y pruébala en un programa. Esta función calcula la suma de todos los dígitos que contiene un número. Por ejemplo, **sumatorioDeDigitos(203)** da como resultado 5, **sumatorioDeDigitos(9310722)** da 24, **sumatorioDeDigitos(0)** da 0, etc. A continuación se muestra la cabecera de la función:

```
public static int sumatorioDeDigitos(long x)
```

2. Crea la clase **Potencia**. Cada potencia tiene una base y un exponente. Para simplificar, consideramos únicamente las potencias con exponente entero (la base sí podrá tener decimales). Cuando se multiplican dos potencias con la misma base, se conserva la base y se suman los exponentes. Cuando se dividen dos potencias con la misma base, se conserva la base y se restan los exponentes. Define convenientemente el método **toString** y los métodos **multiplicaPor()** y **divideEntre()** de tal forma que el siguiente programa muestre la salida que se indica. En caso de que el usuario intente multiplicar o dividir potencias de distinta base, se debe mostrar un mensaje de error y devolver la potencia **cero elevado a cero**.

```
public class Ejercicio02 {  
    public static void main(String[] args) {  
        Potencia p1 = new Potencia(5, 3);  
        Potencia p2 = new Potencia(2, 0);  
        Potencia p3 = new Potencia(5, 2);  
        System.out.println(p1 + " " + p2 + " " + p3);  
        System.out.println(p1.multiplicaPor(p2));  
        System.out.println(p1.multiplicaPor(p3));  
        System.out.println(p1.divideEntre(p3));  
    }  
}
```

5^3 2^0 5^2

Lo siento, no puedo multiplicar potencias de distinta base.

0^0

5^5

5^1

3. Sea A una matriz con m filas y n columnas. La matriz transpuesta, denotada con A^t está formada de tal manera que el elemento a_{ji} de la matriz original A se convertirá en el elemento a_{ij} de la matriz transpuesta A^t .

Por ejemplo, si A es:

1 2

3 4

5 6

La transpuesta de A sería:

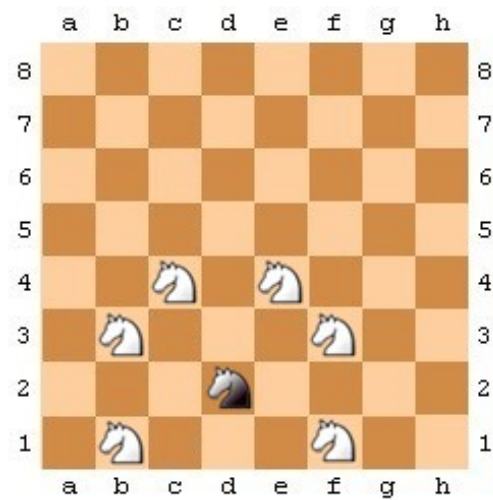
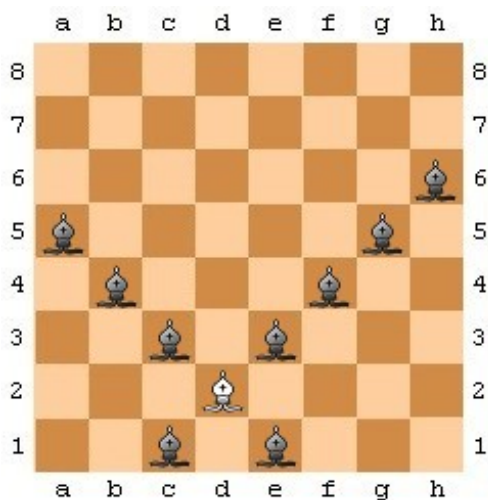
1 3 5

2 4 6

Realiza una función que devuelva la matriz transpuesta a una dada. La cabecera de la función es la siguiente:

```
public static int[][] transpuesta(int[ ][ ] a)
```

4. Crea la clase abstracta **Figura** con los atributos **color**, **fila** y **columna**. Crea, a su vez, las clases **Alfil** y **Caballo** como subclases de **Figura**. Implementa el método **puedeMatar** tanto en la clase **Alfil** como en la clase **Caballo**. El método **puedeMatar** devolverá verdadero si la figura a la que se le aplica el método puede matar a la figura que se pasa como parámetro, teniendo en cuenta las coordenadas de ambas. A continuación se muestra cómo pueden moverse el alfil y el caballo.



NOTA: Por tratarse de una recuperación, este examen se calificará como APTO o NO APTO.