

Familia Profesional Informática y Telecomunicaciones		Nombre del Ciclo Formativo Título de Técnico Superior en Desarrollo de Aplicaciones Web				
Centro Educativo IES Campanillas (sede PTA)		Módulo Profesional Programación Código: 0485 N.º de créditos ECTS: 14			Profesor Juan Antonio Jiménez Morales	
Curso lectivo 2018 / 2019	Grupo 1º DAW	Tipo de documento Examen	Trimestre Segundo – Control 3	Modelo Único	Fecha 16/01/2019	Pág. 1/4

INSTRUCCIONES

- ➔ El alumno debe entregar una carpeta con las soluciones al examen cuyo nombre debe estar formado por "Ex" seguido del número de lista, seguido de las iniciales. Por ejemplo, Facundo Romuedo Piladro que es el número 8 de la lista entregará 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 - si procede - el turno. También debe indicar una breve descripción de lo que hace el programa.
- ➔ Únicamente se necesita entregar el código fuente en java, **no se deben entregar los archivos con la extensión .class**.

EJERCICIOS

- [5 puntos] Implemente el juego de las Torres de Hanói. El juego consiste en tres postes verticales. En uno de los postes se apila un número de discos perforados por su centro (este dato, la cantidad de discos con los que quiera jugar el usuario, será solicitado al usuario). Los discos se apilan sobre uno de los postes en tamaño decreciente de abajo a arriba. No hay dos discos iguales, y todos ellos están apilados de mayor a menor radio -desde la base del poste hacia arriba- en uno de los postes, quedando los otros dos postes vacíos. El juego consiste en pasar todos los discos desde el poste ocupado (es decir, el que posee la torre inicial) al tercer poste (que inicialmente está vacío). Para realizar este objetivo, es necesario seguir tres simples reglas:

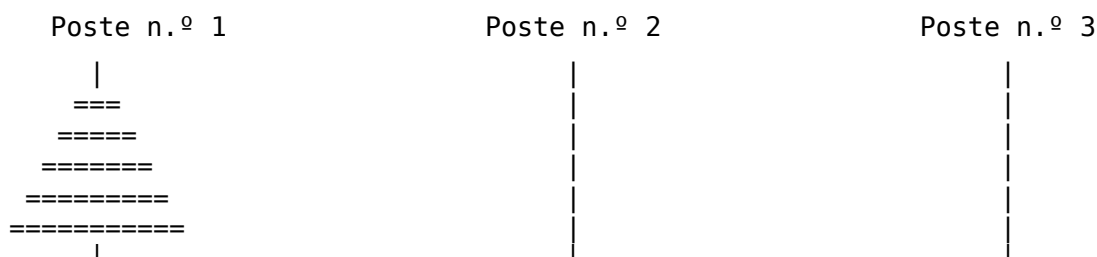
Solo se puede mover un disco cada vez.

Un disco de mayor tamaño no puede estar sobre uno más pequeño que él mismo.

Solo se puede desplazar el disco que se encuentre arriba en cada poste.

Existen diversas formas de llegar a la solución final, todas ellas siguiendo estrategias diversas.

Nosotros vamos a representar en pantalla los discos y los postes de la siguiente manera (suponiendo que partimos de una cantidad inicial de 5 discos):



Nuestro programa va a servir de "tablero de juego" para que un usuario pueda jugar. Debe, por tanto, capturar movimientos que quiera hacer el usuario, comprobar que se puede hacer, y reflejarlo en el tablero. El programa terminará cuando haya conseguido traspasar todos los discos al poste número 3, o cuando el usuario indique como número de poste -1.

Para representar en memoria el estado de cada poste, vamos a utilizar un array bidimensional de enteros, con 3 filas (una por cada poste) y con una cantidad de enteros por cada fila igual a la cantidad de discos con los que quiera jugar el usuario + 1. En la primera posición de cada fila, se almacenaría el n.º de discos que hay actualmente en cada poste, y desde la posición 1 en adelante, el "radio" de cada disco (a mayor número, mayor radio y, por tanto, disco mayor). Así, inicialmente, el estado de los arrays correspondientes a los postes de arriba sería el siguiente:

Posición->	0	1	2	3	4	5
Poste 1:	5	5	4	3	2	1
Poste 2:	0	0	0	0	0	0
Poste 3:	0	0	0	0	0	0

Familia Profesional Informática y Telecomunicaciones		Nombre del Ciclo Formativo Título de Técnico Superior en Desarrollo de Aplicaciones Web				
Centro Educativo IES Campanillas (sede PTA)		Módulo Profesional Programación Código: 0485 N.º de créditos ECTS: 14			Profesor Juan Antonio Jiménez Morales	
Curso lectivo 2018 / 2019	Grupo 1º DAW	Tipo de documento Examen	Trimestre Segundo – Control 3	Modelo Único	Fecha 16/01/2019	Pág. 2/4

La cantidad de discos con la que desea jugar el usuario se solicitará al inicio del juego. Esta cantidad oscilará entre 2 y 10, ambos incluidos.

Puede utilizar las siguientes funciones para implementar este juego:			
Función	Argumentos	Qué hace	Devuelve
inicializaPostes	<ul style="list-style-type: none"> Número de discos 	Reserva memoria para un array bidimensional de enteros, con 3 filas (una por cada poste) y tantas columnas como número de discos indicado como argumento + 1. Inicializa la primera fila (poste) con todos los discos, y el resto de filas sin discos.	Array bidimensional de enteros
muestraTorres	<ul style="list-style-type: none"> Array bidimensional de enteros 	Presenta por pantalla el estado en el que se encuentran las torres	Nada
intentaMover	<ul style="list-style-type: none"> Array bidimensional de enteros, número de poste desde el que se desea mover un disco, número de poste hacia el que se desea mover el disco 	Mueve, si se puede, el disco situado en la cima del poste origen a la cima del poste destino	Verdadero si ha podido realizar el movimiento; falso en caso contrario. Además, si ha podido realizar el movimiento, dicho movimiento queda reflejado en el array bidimensional, ya que pasa por referencia.
checkHanoi	<ul style="list-style-type: none"> Array bidimensional de enteros 	Comprueba si todos los discos están situados en el último poste, con lo que el juego acabaría	Verdadero si se han colocado todos los discos en el último poste; falso en caso contrario

- El alumno puede optar por no implementar la función muestraTorres, y utilizar el código intermedio facilitado, haciendo uso de paquetes en Java. En este caso, el ejercicio tendrá una puntuación menor. Para utilizar el código intermedio, debe saber que dicha función está implementada dentro de una clase denominada `accesorioHanoi`, la cual forma parte de un paquete denominado `libreriaHanoi`, que debe estar ubicado en la carpeta `libreriaHanoi`.

Familia Profesional Informática y Telecomunicaciones		Nombre del Ciclo Formativo Título de Técnico Superior en Desarrollo de Aplicaciones Web				
Centro Educativo IES Campanillas (sede PTA)		Módulo Profesional Programación Código: 0485 N.º de créditos ECTS: 14			Profesor Juan Antonio Jiménez Morales	
Curso lectivo 2018 / 2019	Grupo 1º DAW	Tipo de documento Examen	Trimestre Segundo – Control 3	Modelo Único	Fecha 16/01/2019	Pág. 3/4

2. [2,5 puntos] Elija una de las opciones:

OPCIÓN A: Escribe un programa que, a partir de 2 enteros introducidos por el usuario, diga cuántos dígitos del primer número están en el segundo número. Se recomienda usar `long` en lugar de `int` ya que el primero admite números más largos.

Ejemplo 1:

Por favor, introduzca el primer número entero positivo: 416783

Por favor, introduzca el segundo número entero positivo: 356

En el número 416783 hay 2 dígitos que están en el número 356

Observe que los dígitos del primer número que se repiten en el segundo son los subrayados: 416783

Ejemplo 2:

Por favor, introduzca un número entero positivo: 3177842

Por favor, introduzca el segundo número entero positivo: 7353

En el número 3177842 hay 3 dígitos que están en el número 7353

Observe que los dígitos del primer número que se repiten en el segundo son los subrayados: 3177842

→ Para la implementación de este ejercicio, debe apoyarse en el uso de funciones “que tengan sentido” y que ayuden a simplificar la codificación y el entendimiento del problema. De no hacerlo, la puntuación será inferior.

Como puede comprobar, los dígitos que se repitan en el primer número se contabilizan tantas veces como aparezcan (y, claro está, siempre que también lo tenga al menos una vez el segundo número). Por eso, en el segundo ejemplo, se contabilizan 3 dígitos (una vez el 3 y dos veces el 7).

OPCIÓN B. Crea la función de manejo de arrays que tenga la siguiente cabecera y que haga lo que se especifica en los comentarios. Dicha función debe pertenecer a un paquete con una librería que debes definir:

```
public int[] filtraSin7(int x[])
//Devuelve un array con todos los números que NO contienen el dígito 7
//(por ej. 8, 26, 360, 982...) que se encuentren en otro array
//que se pasa como parámetro.
```

Obviamente el tamaño del array que se devuelve será menor o igual al que se pasa como parámetro. Utiliza esta función en un programa para comprobar que funcionan bien. Para que el ejercicio resulte más fácil, las repeticiones de números que NO contienen “sietes” se conservan; es decir, si en el array x el número 865 se repite 3 veces, en el array devuelto también estará repetido 3 veces. Si no existe ningún número que NO contiene “sietes” en el array x, se devuelve un array con el número -1 como único elemento.

Familia Profesional Informática y Telecomunicaciones		Nombre del Ciclo Formativo Título de Técnico Superior en Desarrollo de Aplicaciones Web				
Centro Educativo IES Campanillas (sede PTA)		Módulo Profesional Programación Código: 0485 N.º de créditos ECTS: 14			Profesor Juan Antonio Jiménez Morales	
Curso lectivo 2018 / 2019	Grupo 1º DAW	Tipo de documento Examen	Trimestre Segundo – Control 3	Modelo Único	Fecha 16/01/2019	Pág. 4/4

3. [2,5 puntos] Genera el precio (bien desglosado) de un pedido a un almacén en función de las siguientes condiciones:

- Se pregunta el precio del artículo, el número de unidades y el código de descuento.
- El precio inicial del pedido se calcula multiplicando el precio unitario por la cantidad.
- El código 685 hace que se resten 15 euros.
- El código 778 aplica un 5% de descuento a las 20 primeras unidades.
- El código 219 hace que las primeras 5 unidades salgan gratis (válido solo si se piden más de 5 unidades).
- Al final se suma el 21% de IVA.

Cualquier código diferente a los anteriormente mencionados no tiene ningún efecto sobre el precio.

Ejemplo 1:

Introduzca el precio del artículo (en Euros): 35,50

Indique el número de unidades: 7

Indique el código de descuento: 685

A continuación, se le muestra el precio desglosado del pedido:

```

-----
Precio del artículo:                35,50 €
Número de unidades:                7
Precio total unidades:             248,50 €
Descuento aplicado: Código 685:    15,00 €
Precio con descuento:              233,50 €
IVA (21%):                        49,04 €
-----
PRECIO FINAL:                      282,54 €

```

Ejemplo 2:

Introduzca el precio del artículo (en Euros): 35,50

Indique el número de unidades: 8

Indique el código de descuento: 238

A continuación, se le muestra el precio desglosado del pedido:

```

-----
Precio del artículo:                35,50 €
Número de unidades:                8
Precio total unidades:             284,00 €
Descuento aplicado: Código no válido 0,00 €
Precio con descuento:              284,00 €
IVA (21%):                        59,64 €
-----
PRECIO FINAL:                      343,64 €

```

Ejemplo 3:

Introduzca el precio del artículo (en Euros): 35,50

Indique el número de unidades: 8

Indique el código de descuento: 219

A continuación, se le muestra el precio desglosado del pedido:

```

-----
Precio del artículo:                35,50 €
Número de unidades:                8
Precio total unidades:             284,00 €
Descuento aplicado: Código 219:    177,50 €
Precio con descuento:              106,50 €
IVA (21%):                        22,37 €
-----
PRECIO FINAL:                      128,87 €

```

→ Para la implementación de este ejercicio, debe apoyarse en el uso de funciones “que tengan sentido” y que ayuden a simplificar la codificación y el entendimiento del problema. De no hacerlo, la puntuación será inferior.