

Práctica Individual 2 – Ejercicios recursivos

A resolver en clases de prácticas por el profesor (**NO hay que incluirlos en la entrega**):

1. Dada una lista de enteros, obtener su inversa.
2. Diseñe un algoritmo que dada una matriz de $n \times n$ enteros (con $n=2^m$; siendo m un número entero mayor que 0), devuelva cierto si el valor en la casilla superior izquierda es menor al de la casilla inferior derecha, y cada una de las 4 submatrices cumplen también dicha propiedad, y así sucesivamente.
3. Diseñar un algoritmo recursivo, con y sin memoria, y posteriormente encontrar un algoritmo iterativo para la siguiente definición:

$$g(a, b) = \begin{cases} a + b^2, & a < 2 \wedge b < 2 \\ a^2 + b, & a < 2 \vee b < 2 \\ g\left(\frac{a}{2}, b-1\right) + g\left(\frac{a}{3}, b-2\right) + g\left(a-2, \frac{b}{4}\right), & \text{en otro caso} \end{cases}$$

siendo a y b números enteros positivos.

A resolver por los alumnos (**SÍ hay que incluirlos en la entrega**):

1. Dada la siguiente definición recursiva de la función f (que toma como entrada 3 números enteros positivos y devuelve una cadena):

$$f(a, b, c) = \begin{cases} "(" + toString(a * b * c) + ")", & a < 3 \wedge b < 3 \wedge c < 3 \\ "(" + toString(a + b + c) + ")", & a < 5 \vee b < 5 \vee c < 5 \\ toString(a * b * c) + f(a/2, b-2, c/2), & a \text{ es par} \wedge b \text{ es par} \wedge c \text{ es par} \\ toString(a + b + c) + f(a/3, b-3, c/3), & \text{en otro caso} \end{cases}$$

siendo $+$ un operador que representa la concatenación de cadenas, y $toString(i)$ un método que devuelve una cadena a partir de un entero. Al llevar a cabo la implementación, para el tratamiento de cadenas se recomienda hacer uso de *String.format*.

2. Dada una matriz de $n \times n$ cadena de caracteres (con $n=2^m$; siendo m un número entero mayor que 0), devolver una lista de cadenas de caracteres que incluya las cadenas que se forman uniendo las 4 cadenas de las 4 esquinas de la matriz principal, y de cada una de sus 4 submatrices, y así sucesivamente hasta llegar a una matriz de 2×2 . El orden en el que se unen las cadenas de las esquinas es: superior izquierda, superior derecha, inferior izquierda, e inferior derecha.
3. Dada una lista de enteros ordenada de mayor a menor, diseñar un algoritmo que devuelva un conjunto que incluya los elementos de dicha lista cuyo valor se encuentre en un rango $[a, b)$ dado (siendo a y b de tipo entero). Para trabajar con conjuntos, haga uso del tipo *IntegerSet* del repositorio.

4. Diseñar un algoritmo recursivo, con y sin memoria, y posteriormente encontrar un algoritmo iterativo que calcule los valores de la recurrencia $f_n = 2*f_{n-1} + 4*f_{n-2} + 6*f_{n-3}$, $f_2 = 6$, $f_1 = 4$, $f_0 = 2$.

5. Diseñar un algoritmo recursivo, con y sin memoria, y posteriormente encontrar un algoritmo iterativo para la siguiente definición:

$$g(a, b, c) = \begin{cases} a + b^2 + 2 * c, & a < 3 \vee b < 3 \vee c < 3 \\ g(a - 1, b/2, c/2) + g(a - 3, b/3, c/3), & a \text{ es múltiplo de } b \\ g\left(\frac{a}{3}, b - 3, c - 3\right) + g\left(\frac{a}{2}, b - 2, c - 2\right), & \text{en otro caso} \end{cases}$$

siendo a, b y c números enteros positivos.

Tenga en cuenta que:

- Para cada ejercicio debe leer los datos de entrada de un fichero, y mostrar la salida por pantalla. Dicha lectura debe ser independiente del algoritmo concreto que resuelva el ejercicio.
- La solución tiene que ser acorde al material de la asignatura proporcionado.

SE PIDE resolver de forma eficiente:

- Ejercicio 1: Proporcione una solución iterativa usando while, una recursiva no final, una recursiva final, y una en notación funcional.
- Ejercicios 2 y 3: proporcione una solución recursiva.
- Ejercicio 4 y 5: proporcione una solución recursiva sin memoria, otra recursiva con memoria, y otra iterativa.

DEBE REALIZAR SU ENTREGA EN 2 PARTES:

- Proyecto en eclipse con las soluciones en Java.
- Memoria de la práctica en un único archivo PDF, que debe contener:
 - Código realizado
 - Volcado de pantalla con los resultados obtenidos para las pruebas realizadas, incluyendo al menos los resultados obtenidos para los tests proporcionados.