

Guía para la cuarta sesión de laboratorio

Especificación, Validación y Testing. Curso 2015-2016

1. Sea la función `encuentraPrimeraRedonda` de la guía de la segunda sesión de laboratorio, escribe un conjunto de tests siguiendo la estrategia “boundary value analysis”. Se deben especificar y nombrar las clases de equivalencia e indicar, para cada test, de qué clase o clases es representante.
2. Sea la función `merge` que dados dos arrays de números enteros, ordenados en orden creciente, devuelve un nuevo array, también ordenado en orden creciente, que contiene todos los elementos de ambos arrays de entrada. Si alguno de los arrays es nulo se producirá un `NullPointerException`. Se pide lo siguiente:
 - a) Escribe un conjunto de tests siguiendo la estrategia “boundary value analysis”. Se deben especificar y nombrar las clases de equivalencia e indicar, para cada test, de qué clase o clases es representante.
 - b) Dibuja el CFG de bajo nivel incluyendo las aristas por posibles excepciones.
 - c) Implementa la función en Java de forma iterativa en orden lineal respecto a la suma de las longitudes de los dos arrays.
 - d) Escribe dos conjuntos de tests que satisfagan respectivamente los criterios MC/DC y path coverage (de bajo nivel y una iteración máximo).
3. Sea la clase `SortedListInt` que representa listas de números enteros enlazadas y ordenadas (disponible en el CV). Se pide lo siguiente:
 - a) Escribir dos conjuntos de tests que satisfagan respectivamente los criterios MC/DC y path coverage (de bajo nivel y una iteración máximo) para el método `merge`. Ten en cuenta los posibles caminos de excepción.
 - b) ¿Observas algún error en el método `merge` a partir de los tests generados? Prueba con listas vacías.