

## Diseño de pruebas de la clase ArrayListCountry

### Configuración de los Escenarios

Nombre	Clase	Escenario
setup1	ArrayListCountry	Crea un objeto de la clase ArrayListCountry, con un ArrayList con Long n posiciones añadiendo los elementos de forma aleatoria. alc.add(n)1 alc.add(n)2 alc.add(n)3 alc.add(n)4 alc.add(n)5

### Diseño de Casos de Prueba

<b>Objetivo de la Prueba:</b> Verificar que el método searchIterative de la clase ArrayListCountry funcione correctamente, según el ArrayList del setUp.				
Clase	Método	Escenario	Valores de Entrada	Resultado
ArrayListCountry	searchIterative	setUp1	n=1	El método retorna un boolean con valor true
ArrayListCountry	searchIterative	setUp1	n=2	El método retorna un boolean con valor true
ArrayListCountry	searchIterative	setUp1	n=3	El método retorna un boolean con valor true
ArrayListCountry	searchIterative	setUp1	n=4	El método retorna un boolean con valor true
ArrayListCountry	searchIterative	setUp1	n=5	El método retorna un boolean con valor true
ArrayListCountry	searchIterative	setUp1	n=0	El método retorna un boolean con valor false

### Diseño de Casos de Prueba

<b>Objetivo de la Prueba:</b> Verificar que el método searchRecursive de la clase ArrayListCountry funcione correctamente, según el ArrayList del setUp.				
Clase	Método	Escenario	Valores de Entrada	Resultado
ArrayListCountry	searchRecursive	setUp1	n=1	El método retorna un boolean con valor true
ArrayListCountry	searchRecursive	setUp1	n=2	El método retorna un boolean con valor true
ArrayListCountry	searchRecursive	setUp1	n=3	El método retorna un boolean con valor true
ArrayListCountry	searchRecursive	setUp1	n=4	El método retorna un boolean con valor true
ArrayListCountry	searchRecursive	setUp1	n=5	El método retorna un boolean con valor true

ArrayListCountry	searchRecursive	setUp1	n=0	El método retorna un boolean con valor false
------------------	-----------------	--------	-----	--

**Objetivo de la Prueba:** Verificar que el método deleteIterative de la clase ArrayListCountry funcione correctamente, según el ArrayList del setUp.

Clase	Método	Escenario	Valores de Entrada	Resultado
ArrayListCountry	deleteIterative	setUp1	n=1	El método retorna un boolean con valor true. Verifica que en la posición donde estaba el valor eliminado sea 2
ArrayListCountry	deleteIterative	setUp1	n=5	El método retorna un boolean con valor true. Verifica que en la posición donde estaba el valor eliminado sea 4
ArrayListCountry	deleteIterative	setUp1	n=3	El método retorna un boolean con valor true. Verifica que en la posición donde estaba el valor eliminado sea 4
ArrayListCountry	deleteIterative	setUp1	n=0	El método retorna un boolean con valor false.

**Objetivo de la Prueba:** Verificar que el método deleteRecursive de la clase ArrayListCountry funcione correctamente, según el ArrayList del setUp.

Clase	Método	Escenario	Valores de Entrada	Resultado
ArrayListCountry	deleteRecursive	setUp1	n=1	El método retorna un boolean con valor true. Verifica que en la posición donde estaba el valor eliminado sea 2
ArrayListCountry	deleteRecursive	setUp1	n=5	El método retorna un boolean con valor true. Verifica que en la posición donde estaba el valor eliminado sea 4
ArrayListCountry	deleteRecursive	setUp1	n=3	El método retorna un boolean con valor true. Verifica que en la posición donde estaba el valor eliminado sea 4
ArrayListCountry	deleteRecursive	setUp1	n=0	El método retorna un boolean con valor false.

## Diseño de pruebas de la clase LinkedListCountry

### Configuración de los Escenarios

Nombre	Clase	Escenario
setup1	LinkedListCountry	Crea un objeto de la clase LinkedListCountry, con una LinkedList vacía
setup2	LinkedListCountry	Crea un objeto de la clase LinkedListCountry, con una LinkedList llc.add(1) llc.add(2) llc.add(3) llc.add(4)
setup3	LinkedListCountry	Crea un objeto de la clase LinkedListCountry, con una LinkedList llc.add(1) llc.add(2) llc.add(3) llc.add(4) llc.add(5)
setup4	LinkedListCountry	Crea un objeto de la clase LinkedListCountry, con una LinkedList llc.add(1)

### Diseño de Casos de Prueba

**Objetivo de la Prueba:** Verificar que el método addIterative de la clase LinkedListCountry funcione correctamente, según la lista enlazada creadas en el setUp1.

Clase	Método	Escenario	Valores de Entrada	Resultado
LinkedListCountry	addIterative	Setup1	llc.addIterative(1)	El dato número 1 se ha añadido a correctamente a la linkedList
LinkedListCountry	addIterative	Setup1	llc.addIterative(2)	El dato número 2 se ha añadido a correctamente a la linkedList
LinkedListCountry	addIterative	Setup1	llc.addIterative(3)	El dato número 3 se ha añadido a correctamente a la linkedList
LinkedListCountry	addIterative	Setup1	llc.addIterative(4)	El dato número 4 se ha añadido a correctamente a la linkedList

**Objetivo de la Prueba:** Verificar que el método addRecursive de la clase LinkedListCountry funcione correctamente, según la lista enlazada creadas en el setUp1.

Clase	Método	Escenario	Valores de Entrada	Resultado
LinkedListCountry	addRecursive	Setup1	llc.addRecursive(1)	El dato número 1 se ha añadido a correctamente a la linkedList
LinkedListCountry	addRecursive	Setup1	llc.addRecursive(2)	El dato número 2 se ha añadido a correctamente a la linkedList
LinkedListCountry	addRecursive	Setup1	llc.addRecursive(3)	El dato número 3 se ha añadido a correctamente a la linkedList
LinkedListCountry	addRecursive	Setup1	llc.addRecursive(4)	El dato número 4 se ha añadido a correctamente a la linkedList

**Objetivo de la Prueba:** Verificar que el método searchIterative de la clase LinkedListCountry funcione correctamente, según las listas enlazadas creadas en los setup.

Clase	Método	Escenario	Valores de Entrada	Resultado
LinkedListCountry	searchIterative	Setup1	l1c.searchIterative (1)	No se ha encontrado ningún elemento en la linkedList
LinkedListCountry	searchIterative	Setup2	l1c.searchIterative (1)	El dato número 1 se ha sido encontrado a en la linkedList
LinkedListCountry	searchIterative	Setup2	l1c.searchIterative (2)	El dato número 2 se ha sido encontrado a en la linkedList
LinkedListCountry	searchIterative	Setup2	l1c.searchIterative (3)	El dato número 3 se ha sido encontrado a en la linkedList
LinkedListCountry	searchIterative	Setup2	l1c.searchIterative (4)	El dato número 4 se ha sido encontrado a en la linkedList
LinkedListCountry	searchIterative	Setup2	l1c.searchIterative (5)	El dato no existe en la linkedList
LinkedListCountry	searchIterative	Setup2	l1c.searchIterative (0)	El dato no existe en la linkedList

**Objetivo de la Prueba:** Verificar que el método searchRecursive de la clase LinkedListCountry funcione correctamente, según las listas enlazadas creadas en los setup.

Clase	Método	Escenario	Valores de Entrada	Resultado
LinkedListCountry	searchRecursive	Setup1	l1c. searchRecursive(1)	No se ha encontrado ningún elemento en la linkedList
LinkedListCountry	searchRecursive	Setup2	l1c. searchRecursive(1)	El dato número 1 se ha sido encontrado a en la linkedList
LinkedListCountry	searchRecursive	Setup2	l1c. searchRecursive(2)	El dato número 2 se ha sido encontrado a en la linkedList
LinkedListCountry	searchRecursive	Setup2	l1c. searchRecursive(3)	El dato número 3 se ha sido encontrado a en la linkedList
LinkedListCountry	searchRecursive	Setup2	l1c. searchRecursive(4)	El dato número 4 se ha sido encontrado a en la linkedList
LinkedListCountry	searchRecursive	Setup2	l1c. searchRecursive(5)	El dato no existe en la linkedList
LinkedListCountry	searchRecursive	Setup2	l1c. searchRecursive(0)	El dato no existe en la linkedList

**Objetivo de la Prueba:** Verificar que el método deletelterative de la clase LinkedListCountry funcione correctamente, según las listas enlazadas creadas en los setup.

Clase	Método	Escenario	Valores de Entrada	Resultado
LinkedListCountry	deletelterative	Setup1	l1c.deletelterative (1)	No se ha encontrado ningún elemento en la linkedList
LinkedListCountry	deletelterative	Setup2	l1c.deletelterative (1)	El dato número 1 se ha sido encontrado a en la linkedList
LinkedListCountry	deletelterative	Setup2	l1c.deletelterative (2)	El dato número 2 se ha sido encontrado a en la linkedList
LinkedListCountry	deletelterative	Setup2	l1c.deletelterative (3)	El dato número 3 se ha sido encontrado a en la linkedList
LinkedListCountry	deletelterative	Setup2	l1c.deletelterative (4)	El dato número 4 se ha sido encontrado a en la linkedList
LinkedListCountry	deletelterative	Setup2	l1c.deletelterative (5)	El dato no existe en la linkedList
LinkedListCountry	deletelterative	Setup2	l1c.deletelterative (0)	El dato no existe en la linkedList

**Objetivo de la Prueba:** Verificar que el método deleteRecursive de la clase LinkedListCountry funcione correctamente, según las listas enlazadas creadas en los setup.

Clase	Método	Escenario	Valores de Entrada	Resultado
LinkedListCountry	deleteRecursive	Setup1	l1c.deleteRecursive (1)	No se ha encontrado ningún elemento en la linkedList
LinkedListCountry	deleteRecursive	Setup2	l1c. deleteRecursive (1)	El dato número 1 se ha sido encontrado a en la linkedList
LinkedListCountry	deleteRecursive	Setup2	l1c.deleteRecursive(2)	El dato número 2 se ha sido encontrado a en la linkedList
LinkedListCountry	deleteRecursive	Setup2	l1c.deleteRecursive(3)	El dato número 3 se ha sido encontrado a en la linkedList
LinkedListCountry	deleteRecursive	Setup2	l1c.deleteRecursive(4)	El dato número 4 se ha sido encontrado a en la linkedList
LinkedListCountry	deleteRecursive	Setup2	l1c.deleteRecursive(5)	El dato no existe en la linkedList
LinkedListCountry	deleteRecursive	Setup2	l1c.deleteRecursive(0)	El dato no existe en la linkedList

## Diseño de pruebas de la clase ABBCountry

### Configuración de los Escenarios

Nombre	Clase	Escenario
setup1	ABBCountry	Crea un objeto de la clase ABBCountry, con un abbc vacío
setup2	ABBCountry	Crea un objeto de la clase ABBCountry, con una abbc abbc.addIterative (20) abbc.addIterative (40) abbc.addIterative (60) abbc.addIterative (80) abbc.addIterative (100) abbc.addIterative (120)

### Diseño de Casos de Prueba

**Objetivo de la Prueba:** Verificar que el método addIterative de la clase ABBCountry funcione correctamente, según la lista enlazada creadas en el setUp1.

Clase	Método	Escenario	Valores de Entrada	Resultado
ABBCountry	addIterative	Setup1	abbc.addIterative (300) abbc.addIterative (30)	Se agregaron los elementos en el orden correcto
ABBCountry	addIterative	Setup1	abbc.addIterative (10) abbc.addIterative (1250)	Se agregaron los elementos en el orden correcto

**Objetivo de la Prueba:** Verificar que el método addRecursive de la clase ABBCountry funcione correctamente, según la lista enlazada creadas en el setUp1.

Clase	Método	Escenario	Valores de Entrada	Resultado
ABBCountry	addRecursive	Setup1	abbc.addRecursive(300) abbc.addRecursive(30)	Se agregaron los elementos en el orden correcto
ABBCountry	addRecursive	Setup1	abbc.addRecursive(10) abbc.addRecursive(1250)	Se agregaron los elementos en el orden correcto

**Objetivo de la Prueba:** Verificar que el método searchIterative de la clase ABBCountry funcione correctamente, según la lista enlazada creadas en el setUp2.

Clase	Método	Escenario	Valores de Entrada	Resultado
ABBCountry	searchIterative	Setup2	abbc.searchIterative(20)	El dato número 20 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchIterative(40)	El dato número 40 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchIterative(60)	El dato número 60 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchIterative(80)	El dato número 80 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchIterative(100)	El dato número 100 ha sido encontrado en el ABB

ABBCountry	searchIterative	Setup2	abbc.searchIterative(10)	El dato no existe en el ABB
------------	-----------------	--------	--------------------------	-----------------------------

**Objetivo de la Prueba:** Verificar que el método searchRecursive de la clase ABBCountry funcione correctamente, según la lista enlazada creadas en el setUp2.

Clase	Método	Escenario	Valores de Entrada	Resultado
ABBCountry	searchIterative	Setup2	abbc.searchREcursive(20)	El dato número 20 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchREcursive(40)	El dato número 40 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchREcursive(60)	El dato número 60 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchREcursive(80)	El dato número 80 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchREcursive(100)	El dato número 100 ha sido encontrado en el ABB
ABBCountry	searchIterative	Setup2	abbc.searchREcursive(10)	El dato no existe en el ABB

**Objetivo de la Prueba:** Verificar que el método deleteliterativo de la clase ABBCountry funcione correctamente, según la lista enlazada creadas en el setUp2.

Clase	Método	Escenario	Valores de Entrada	Resultado
ABBCountry	deleteliterativo	Setup2	abbc.deleteliterativo(20)	El número 20 se elimina correctamente del ABB, siendo 40 el nuevo top
ABBCountry	deleteliterativo	Setup2	abbc.deleteliterativo(40)	El número 40 se elimina correctamente del ABB, siendo 60 el nuevo top
ABBCountry	deleteliterativo	Setup2	abbc.deleteliterativo(80)	El número 80 se elimina correctamente del ABB, siendo 60 el nuevo top
ABBCountry	deleteliterativo	Setup2	abbc.deleteliterativo(10)	El dato no existe dentro del ABB

**Objetivo de la Prueba:** Verificar que el método deleteRecursive de la clase ABBCountry funcione correctamente, según las listas enlazadas creadas en los setup.

Clase	Método	Escenario	Valores de Entrada	Resultado
ABBCountry	deleteRecursive	Setup2	abbc.deleteRecursive(20)	El número 20 se elimina correctamente del ABB, siendo 40 el nuevo top
ABBCountry	deleteRecursive	Setup2	abbc.deleteRecursive(40)	El número 40 se elimina correctamente del ABB, siendo 60 el nuevo top
ABBCountry	deleteRecursive	Setup2	abbc.deleteRecursive(80)	El número 80 se elimina correctamente del ABB, siendo 60 el nuevo top
ABBCountry	deleteRecursive	Setup2	abbc.deleteRecursive(10)	El dato no existe dentro del ABB