Diseño de pruebas unitarias

Alejandro Arce

Duvan Cuero

Alexander Samaca

Universidad ICESI

Algoritmos y Estructuras de datos

Santiago de Cali

Noviembre 2020

**Diseños de pruebas unitarias**

En este documento se desarrollará la documentación de los casos de pruebas de la base de datos, así como de la estructura de datos

**BSTAVL test Cases**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpEmptyBSTAVL | BSTAVLTest | An empty BSTAVL. |

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUp BSTAVLTest | BSTAVLTest | { (23,Armando)(5,Luis)(8,Alejandro)(32,Duvan)(12,Alex)(4,Andres)(7,Esteban) } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** addTest, probar si el método add agrega adecuadamente objetos dentro del BSTAVL aun si ya hay objetos dentro del BSTAVL. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTAVLTest | Add | setUpBSTAVL | key=59  key=67  key=98 | El resultado esperado es que el tamaño del BSTAVL vaya aumentando a medida que insertamos objetos en la estructura |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** deleteTest, probar si el método delete borra adecuadamente objetos dentro del BSTAVL. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTAVLTest | remove | setUpBSTAVL |  | El resultado esperado es que el elemento sea borrado por lo cual nos retorna el objeto a borrar. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** searchTest, probar si el método search busca adecuadamente objetos dentro del BSTAVL | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTAVLTest | search | setUpBSTAVL | key=23 | El resultado esperado es que el método retorne el elemento buscado dentro del árbol binario de búsqueda. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** deleteTest, probar si el método remove no elimina en un árbol vacio BSTAVL | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTAVLTest | remove | setUpEmptyBSTAVL | key=5 | El resultado esperado es que el método retorne falso al no eliminar ningún objeto del árbol binario de búsqueda. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** searchTest, probar si el método search no busca en un árbol vacio BSTAVL | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTAVLTest | search | setUpEmptyBSTAVL | key=5 | El resultado esperado es que el método retorne falso al no encontrar ningún objeto del árbol binario de búsqueda. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** removeAllTest, probar si el método removeAll elimina todos los elementos de un árbol vacio BSTAVL | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTAVLTest | removeAll | setUpBSTAVL |  | El resultado esperado es que el método removeAll elimine todos objeto del árbol binario de búsqueda y así mismo su tamaño disminuye a 0. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** getValueBalancedTest, probar si el método getValueBalanced retorna el factor de balanceo de un BSTAVL | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTAVLTest | getValueBalanced | setUpBSTAVL |  | El resultado esperado es que el método retorne un entero con el valor de factor de balanceo del árbol binario de búsqueda. |

# **BST Test Case**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpEmptyBST | BSTTest | An empty BST. |

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUp BSTTest | BSTTest | { (23,Armando)(5,Luis)(8,Alejandro)(32,Duvan)(12,Alex)(4,Andres)(7,Esteban) } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** addTest, probar si el método add agrega adecuadamente objetos dentro del BST aun si ya hay objetos dentro del BST. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTTest | Add | setUpBST | (72,“Capitán Teemo”)  (90,“Capitán Gareb”)  (102“Capitán Ashe”) | El resultado esperado es que el tamaño del BST vaya aumentando a medida que insertamos objetos en la estructura |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** removeTest, probar si el método remove borra adecuadamente objetos dentro del BST. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTTest | remove | setUpBST |  | El resultado esperado es que el elemento sea borrado por lo cual nos retorna el objeto a borrar. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** searchTest, probar si el método add agrega adecuadamente objetos dentro del BST aun si ya hay objetos dentro del BST. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTTest | search | setUpBST |  | El resultado esperado es que el método retorne el elemento buscado dentro del árbol binario de búsqueda. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** isEmpty, prueba que el método isEmpty funciona correctamente y retorna el valor booleano esperado | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTTest | isEmpty | setUpEmptyBST |  | El resultado esperado es que el método debe retornar una booleano indicando que la estructura se encuentra vacia. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** transverInOrder, prueba que el método transverInOrder funciona correctamente y retorna el String con los valores del arbol | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTTest | transverInOrder | setUpBST |  | El resultado esperado es que el método debe retornar un String con los valores del árbol binario según el orden del metodo. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** transverPostOrder, prueba que el método transverPostOrder funciona correctamente y retorna el String con los valores del arbol | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTTest | transverPostOrder | setUpBST |  | El resultado esperado es que el método debe retornar un String con los valores del árbol binario según el orden del metodo. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** transverPreOrder, prueba que el método transverPreOrder funciona correctamente y retorna el String con los valores del arbol | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| BSTTest | transverPreOrder | setUpBST |  | El resultado esperado es que el método debe retornar un String con los valores del árbol binario según el orden del metodo. |

**Diseños de casos de prueba del modelo**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpEmpyDataBase | DataBaseTest | An empty DataBase |

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpDataBase | DataBaseTest | A Data base with 10000 people |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** generate, prueba que el método generate funciona correctamente y genera el número de registros de personas en la base de datos | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| DataBaseTest | generate | setUpEmptyDataBase | numsOfPeople=10000000 | El resultado esperado es que el genere 10000000 registros de personas con atributos que se sacan de unos datasets y unos random, el método retorna el tamaño de la base de datos. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** add, prueba que el método add funciona correctamente y añade una nueva persona a la base de datos | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| DataBaseTest | add | setUpEmptyDataBase | name = Alejandro  lastName = Arce  birthday=”03/02/2001”  height=1.7  sex = male  nacionality: “Colombia” | El resultado esperado es que el método añada a una persona a la base de datos de las personas |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** edit, prueba que el método edit funciona correctamente y edite los atributos de las personas | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| DataBaseTest | edit | setUpDataBase | heigh = 1.9  code = 100263129424 | El resultado esperado es que se pueda editar la altura de la persona con código 10026312924 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** search, prueba que el método search funciona correctamente y busca una persona en la base de datos según el criterio dado | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| DataBaseTest | search | setUpDataBase | fullname = Alejandro Arce | El resultado esperado es que el método busque en la base de datos a Alejandro Arce |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** delete, prueba que el método delete funciona correctamente y elimina un registro de una persona de la base de datos | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| DataBaseTest | delete | setUpDataBase | Code = 10001 | El resultado esperado es que el método borre el registro con código 10001 |