# **Hashtable Test Cases**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpEmptyHashTable | HashTableTest | An empty hashtable. |

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpHashTable | HashTableTest | (“Daniel”,1)  (“Andrew”,2)  (“Carlos”,3) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** putTest, probar si el método put agrega adecuadamente objetos dentro de una vacia HashTable. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| HashTableTest | put | setUpEmptyHashTable | (“Capitán Teemo”,666)  (“Capitán Gareb”,667)  (“Capitán Ashe”,668) | El resultado esperado es que el tamaño de la hashtable vaya aumentando a medida que insertamos objetos en ella |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** putTest, probar si el método put agrega adecuadamente objetos dentro del HashTable aun si ya hay objetos dentro de la hashtable. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| HashTableTest | put | setUpHashTable | (“Capitán Teemo”,666)  (“Capitán Gareb”,667)  (“Capitán Ashe”,668) | El resultado esperado es que el tamaño de la hashtable vaya aumentando a medida que insertamos objetos en ella |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** getTest, probar si el método get lidia correctamente con las colisiones en la HashTable. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| HashTableTest | get | setUpHashTable | (“Capitán Teemo”,666)  (“Capitán Gareb”, 666)  (“Capitán Ashe”, 666) | El resultado esperado es que el método get nos devuelva correctamente los objetos esperados. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** getTest, comprobar el comportamiento adecuado de el método cuando se busca objetos en una Hashtable vacia. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| HashTableTest | get | setUpEmptyHashTable |  | El resultado esperado es que el método get nos devuelva un aviso de que la hashtable esta vacia. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** removeTest, probar si el método remove borra correctamente objetos del HashTable. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| HashTableTest | remove | setUpHashTable | 1,2,3 | El resultado esperado es que el método remove borre correctamente los objetos en la hashtable. |

# **Queue Test Cases**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpEmptyQueue | QueueTest | An empty Queue. |

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpQueue | QueueTest | (“Pablo”)  (“Daniel”)  (“Andrea”) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
|  |  |  |  |  |

# **Stack Test Cases**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpEmptyStack | StackTest | An empty Stack. |

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpStack | StackTest | (“Thing”)  (“Magical”)  (“a”)  (“not”) |

# **LinkedList Test Case**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpEmptyLinkedList | LinkedListTest | An empty LinkedList. |

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpLinkedList | LinkedListTest | { 1,2,3,4,5,6,7,8,9} |

# **PriorityQueue Test Case**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUpEmptyPriorityQueue | PriorityQueueTest | An empty PrirorityQueue. |

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUp PriorityQueueTest | PriorityQueueTest | { (“Camile”,1) ,(“Carlos”,2) , (“Daniel”,3) } |

# **BST Test Case**

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

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUp BSTTest | BSTTest | { 23,5,8,32,12,4,7 } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** testRegisterClient, prueba si el método añadir cliente a la base de datos del bank funciona (testRegisterClient) de manera correcta. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| Bank | testRegisterClient () | setUp2() | (“Capitán Teemo”,666) | el resultado de la prueba es validar que el cliente fue registrado de manera correcta en la base de datos del banco. |