# Diseño de casos de prueba:

# Clase: ALGraphTest

|  |  |
| --- | --- |
| Nombre | AddVertexTest1 |
| Objetivos | verificar que se agregue correctamente el vértice |
| Descripción | Se va a crear un vértice, se le agrega un valor de 1 y se busca dentro del grafo para verificar que no arroje null |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | AddVertexTest2 |
| Objetivos | Retorna un valor falso en caso de que el vertice para agregar ya existia |
| Descripción | Se crea un vertice de valor 1 y verifica que retorne falso si se trata de agregar un vertice exactamente igual |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | AddVertexTest3 |
| Objetivos | Retorna un valor verdadero si el vertice a añadir no existe |
| Descripción | Verifica que se retorne verdadero si se trata de agregar un vertice que no existe |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest1 |
| Objetivos | Retornar un valor falso si el vertice a eliminar no existe |
| Descripción | Verifica que se retorne un valor falso si se trata de eliminar un vertice que no existe |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest2 |
| Objetivos | Retornar un valor positivo si el vertice a borrar ya existe |
| Descripción | Se crea un vertice y luego se verifica que retorne un valor verdadero si el vertice a borrar ya existía |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest3 |
| Objetivos | Borrar un vertice especificado correctamente |
| Descripción | Se crea un vertice de valor 1 y luego se borra el vertice de valor 1, finalmente verifica que se haya borrado el vertice de valor 1 buscandolo en el grafo |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | AddEdgeTest1 |
| Objetivos | Hacer que cada uno de los 2 vertices tenga otro vertice adyacente |
| Descripción | Crea 4 vertices en el grafo, conecta los vertices 1, 3 y 4 con el vertice 2 y después verifica que las aristas entre los vertices existan |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | AddEdgeTest2 |
| Objetivos | Retorna verdadero si la arista no existia antes |
| Descripción | Crea 2 vertices y los añade al grafo y retorna un valor verdadero al añadir una arista entre los vertices |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | RemoveEdgeTest2 |
| Objetivos | Borrar una arista correctamente a pesar del orden de los vértices |
| Descripción | Se crean 2 vertices y se añaden a un mismo grafo, después se crea una arista entre los 2 vertices de peso 2, posteriormente se borra la arista y finalmente se verifica que la arista haya sido borrada del grafo retornando un valor falso |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | RemoveEdgeTest3 |
| Objetivos | Retorna verdadero si la arista ya existe |
| Descripción | Se crean 2 vertices y se añaden al grafo y luego se crea una arista entre los vertices de peso 2 y retorna un valor verdadero si se puede borrar la arista |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | RemoveEdgeTest4 |
| Objetivos | Retorna un valor falso si la arista no existe |
| Descripción | Crea 2 vertices y los añade al grafo y verifica retornando un valor falso si la arista no existe |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | AddEdgeTest3 |
| Objetivos | Retorna un valor falso si la arista existía antes |
| Descripción | Crea 2 vertices y los añade a un grafo, luego crea una arista de peso 2 entre los 2 vertices y para verificar retorna un valor falso si la arista existía antes |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | RemoveEdgeTest1 |
| Objetivos | Borrar la arista correctamente |
| Descripción | Se crean 4 vertices y se añaden al grafo, luego se añaden 3 aristas entre los vertices 1 y 2, 3 y 2, 4 y 2, luego borra la arista entre los vertices 3 y 2 y verifica que la arista se haya eliminado correctamente. Después verifica que las otras aristas sigan existiendo |
| Escenario | SetupStage1 |

# Clase: ALVertexTest

|  |  |
| --- | --- |
| Nombre | CreationTest2 |
| Objetivos | Testea que el valor del vertice vacio sea nulo |
| Descripción | Se verifica que pasando un vertice sea no nulo |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | CreationTest3 |
| Objetivos | Testea que la lista de vertices adyacentes no es nula |
| Descripción | Se verifica que los vertices adyacentes a un vertice no sean nulos |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | CreationTest4 |
| Objetivos | Testea que la lista de vertices adyacentes sea no nula pero cuando se usa un constructor diferente |
| Descripción | Se verifica que los vertices adyacentes a un vertice no sean nulos, mientras se usa un costructor diferente |
| Escenario | SetupStage2 |

|  |  |
| --- | --- |
| Nombre | CreationTest5 |
| Objetivos | Testea que el nuevo vertice no es nulo cuando se usa otro constructor |
| Descripción | Se pasa un vertice para verificar que no es nulo |
| Escenario | SetupStage2 |

|  |  |
| --- | --- |
| Nombre | CreationTest6 |
| Objetivos | Testea que un nuevo vertice tiene un valor especifico |
| Descripción | Toma el valor de un vertice para verificar que no es nulo y luego verifica que el valor del vertice sea igual a 3 |
| Escenario | SetupStage2 |

|  |  |
| --- | --- |
| Nombre | SetTest1 |
| Objetivos | Testea que el valor de un vertice sea actualizado a uno nuevo |
| Descripción | Se le asigna el valor 8 a un vertice y luego se verifica que haya un vertice con el valor 8 |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | AddAdjacentTest1 |
| Objetivos | Testea que los vertices adyacentes sean añadidos correctamente |
| Descripción | Se verifica que los vertices adyacentes a un vertice no sean nulos, mientras se usa un costructor diferente |
| Escenario | SetupStage2 |

|  |  |
| --- | --- |
| Nombre | AddAdjacentTest2 |
| Objetivos | Retorna falso si el vertice adyacente ya existe |
| Descripción | Se verifica que los vertices adyacentes estén creados previamente |
| Escenario | SetupStage2 |

|  |  |
| --- | --- |
| Nombre | RemoveAdjacentTest1 |
| Objetivos | Testea que se eliminen los vertices especificados correctamente |
| Descripción | Verifica si se borra correctamente el vertice adyacente |
| Escenario | SetupStage3 |

|  |  |
| --- | --- |
| Nombre | RemoveAdjacentTest2 |
| Objetivos | Testea que se eliminen los vertices especificados correctamente |
| Descripción | Verifica que se elimine la arista del vertice adyacente |
| Escenario | SetupStage3 |

# Clase: GraphTest

|  |  |
| --- | --- |
| Nombre | AddVertexTest1 |
| Objetivos | Verificar que la matriz tenga un solo vertice |
| Descripción | Se crea un grafo de valor 1 y se verifica obteniendo el mismo grafo en la matriz |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | AddVertexTest2 |
| Objetivos | Verificar que la matriz tenga solo 2 vertices y los pesos de las aristas todos sean -1 |
| Descripción | Se crean 2 vertices en una matriz y luego se asignan valores negativos a las aristas |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | AddVertextTest3 |
| Objetivos | Verificar que la matriz solo tenga 3 vertices |
| Descripción | Se crean 3 vertices y verifica que se encuentren en la matriz |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | AddEdgeTest1 |
| Objetivos | Verificar que la matriz tenga 2 vertices y haya una arista de peso 2 entre ellos |
| Descripción | Se crean 2 vertices en un matriz, luego se crea una arista entre esos vertices de peso 2 y severifica que unidireccionalmente el peso sea el adecuado |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | AddEdgeTest2 |
| Objetivos | Verificar que la matriz tenga 3 vertices con una arista entre el vertice 1 y 2 de peso 2 |
| Descripción | Se crean 3 vertices en la matriz, luego se crea una arista entre los vertices 1 y 2 de peso 2 y se verifica que el resto de aristas tengan peso -1 |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | AddEdgeTest3 |
| Objetivos | Crear 3 vertices en una matriz y añadir 1 arista entre el vertice 1 y 2 de peso 2 y otra arista entre el vertice 1 y 3 con una arista de peso 999 |
| Descripción | Se crean 3 vertices en la matriz y se crean una arista entre el vertice 1 y 2 de peso 2 y luego otra entre los vertices 3 y 1 de peso 999 y verifica que el resto de las aristas tengan el peso -1 |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest1 |
| Objetivos | Se añaden 3 vertices a la matriz y se borra la arista de peso 3 |
| Descripción | Se crean 3 vertices en una matriz, se remueve el vertice de valor 3 y se verifica que no este el vertice 3 siendo adyacente con el |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest2 |
| Objetivos | Testea que se eliminen los vertices especificados correctamente |
| Descripción | Se crean 3 vertices en el grafo y se añade una arista de peso 2 entre los vertices 1 y 2, luego se borra el vertice 3 y se verifica que los vertices adyacentes sean los restantes |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest3 |
| Objetivos | Testea que se eliminen los vertices especificados correctamente |
| Descripción | Se crean 3 vertices en el grafo y se añade una arista de peso 2 entre los vertices 1 y 2, y luego se crea otra arista entre los vertices 1 y 3 de peso 6, luego se borra el vertice 3 y finalmente se verifican las aristas alrededor del vertice 2 |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest4 |
| Objetivos | Testea que se eliminen los vertices especificados correctamente |
| Descripción | Se crean 3 vertices en el grafo y se añade una arista de peso 2 entre los vertices 1 y 2, y luego se crea otra arista entre los vertices 1 y 3 de peso 999, luego se remueve el vertice 2 y se verifica que los pesos de las aristas sean correctos |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest5 |
| Objetivos | Testea que se eliminen los vertices especificados correctamente |
| Descripción | Se crean 3 vertices en el grafo y se añade una arista de peso 2 entre los vertices 1 y 2, y luego se crea otra arista entre los vertices 1 y 3 de peso 6, luego se borra el vertice 3 y finalmente se verifican las aristas alrededor del vertice 2 sean correctas y que los valores de los vertices 1 y 2 tambien sean correctos |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest6 |
| Objetivos | Testea que se eliminen los vertices especificados correctamente |
| Descripción | Se crean 4 vertices en el grafoy y se crean 3 aristas, 1 entre los vertices 1 y 2 de peso 2, y otra entre los vertices 1 y 3 de peso 999, y otra arista entre los vertices 4 y 2 de peso 13 y luego se remueve el vertice 2 finalmente verificando que el peso de las aristas sea el correcto |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | RemoveVertexTest7 |
| Objetivos | Testea que se eliminen los vertices especificados correctamente |
| Descripción | Se crean 4 vertices en el grafoy y se crean 3 aristas, 1 entre los vertices 1 y 2 de peso 2, y otra entre los vertices 1 y 3 de peso 999, y otra arista entre los vertices 4 y 2 de peso 13 y luego se remueve el vertice 2 y 4 finalmente verificando que el peso de las aristas sea el correcto |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | SearchVertexTest1 |
| Objetivos | Verificar la búsqueda de vertices en el grafo |
| Descripción | Se crean 4 vertices en el grafo y verifica que el vertice 2 no sea nulo |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | SearchVertexTest2 |
| Objetivos | Verificar la búsqueda de vertices en el grafo |
| Descripción | Se crean 4 vertices en el grafo y verifica que el vertice 4 no sea nulo |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | SearchVertexTest3 |
| Objetivos | Verificar la búsqueda de vertices en el grafo |
| Descripción | Se crean 4 vertices en el grafo y verifica que el vertice 4 no sea nulo, y después se borra en vertice 4 y se verifica que este vertice sea nulo |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | SearchVertexTest4 |
| Objetivos | Verificar la búsqueda de vertices en el grafo |
| Descripción | Se crean 4 vertices en el grafo y verifica que no se pueda añadir un vertice que ya existe |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | SearchEdgeTest1 |
| Objetivos | Verificar la búsqueda correcta de aristas |
| Descripción | Se crean 4 vertices en el grafo y se añaden 3 aristas y luego verifica que se puedan buscar las aristas correctamente |
| Escenario | SetupStage |

|  |  |
| --- | --- |
| Nombre | SearchEdgeTest2 |
| Objetivos | Verificar la búsqueda correcta de aristas |
| Descripción | Se crean 4 vertices en el grafo y se añaden 3 aristas y luego verifica que se puedan buscar las aristas correctamente, luego borra el vertice 2 y verifica que no se pueda buscar la arista entre el vertice 1 y 2 |
| Escenario | SetupStage |

# Clase: BoardTest

|  |  |
| --- | --- |
| Nombre | CreationTest1 |
| Objetivos | Verifica que el grafo creado tenga el número de casillas especificado |
| Descripción | Verifica que el grafo creado con la lista de adyacencia tenga un numero de 20 casillas |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | CreationTest2 |
| Objetivos | Verificar que las casillas están siendo bien conectadas en el modo normal |
| Descripción | Se obtiene el grafo y se verifica que las casillas se conecten automáticamente en el modo normal |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | CreationTest3 |
| Objetivos | Verificar que las casillas están siendo bien conectadas en el modo normal |
| Descripción | Se obtiene el grafo y se verifica que las casillas se conecten automáticamente en el modo normal |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | CreationTest4 |
| Objetivos | Verificar que las casillas del grafo en los extremos estén siendo conectadas adecuadamente |
| Descripción | Se obtiene un grafo y se verifica que las casillas de los extremos tengan únicamente 3 conecciones |
| Escenario | SetupStage6 |

|  |  |
| --- | --- |
| Nombre | CreationTest5 |
| Objetivos | Que las casillas estén teniendo las conexiones adecuadas en el sentido vertical |
| Descripción | Se obtiene un grafo y se verifica que las casillas tengan las conexiones adecuadas en el sentido vertical |
| Escenario | SetupStage6 |

|  |  |
| --- | --- |
| Nombre | CreationAMTest2 |
| Objetivos | Verificar que las casillas estén bien conectadas en el grafo |
| Descripción | Se obtiene un grafo y se crean las casillas 6, 7 y 8, luego se crean los vertices 1, 2 y 3 y se crean las aristas entre ellos, y finalmente verifica que las casillas hayan sido asignadas a los vertices |
| Escenario | SetupStage8 |

# Clase: SquareTest

|  |  |
| --- | --- |
| Nombre | PlayerTest1 |
| Objetivos | Verificar que las representaciones de los jugadores estén bien hechas en el tablero |
| Descripción | Se le asigna el jugador de Rick al jugador Y, también se le asigna el jugador de Morty al jugador D y verifica que la representación de los jugadores este bien en el tablero |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | PlayerTest2 |
| Objetivos | Verificar que las representaciones de los jugadores estén bien hechas en el tablero |
| Descripción | Se le asigna el jugador de Rick al jugador Y, también se le asigna el jugador de Morty al jugador D y verifica que la representación de los jugadores este bien en el tablero |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | PlayerTest3 |
| Objetivos | Verificar que las representaciones de los jugadores estén bien hechas en el tablero |
| Descripción | Se le asigna el jugador de Rick al jugador Y, también se le asigna el jugador de Morty al jugador D y borra el jugador de Morty para finalmente verificar que solo salga la representacion de Rick en el tablero |
| Escenario | SetupStage1 |

|  |  |
| --- | --- |
| Nombre | PlayerTest4 |
| Objetivos | Verificar que las representaciones de los jugadores estén bien hechas en el tablero |
| Descripción | Se le asigna el jugador de Rick al jugador Y, también se le asigna el jugador de Morty al jugador D y borra el jugador de Rick para finalmente verificar que solo salga la representacion de Morty en el tablero |
| Escenario | SetupStage1 |