PROYECTO FINAL

1. **Configuración de los Escenarios**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Escenario** |
| setUp1() | GraphTest | g= new Graph<Integer>() |
| setUp2() | GraphTest | g= new Graph<Integer>();  g.addNode(1);  g.addNode(2);  g.addNode(3);  g.addNode(4); |
| setUp3() | GraphTest | g= new Graph<Integer>();  for (int i = 0; i <= 5; i++) {  g.addNode(i);  }  g.connect(0, 1, 1);  g.connect(0, 2, 1);  g.connect(1, 3, 1);  g.connect(1, 4, 1);  g.connect(2, 3, 1);  g.connect(3, 5, 1);  g.connect(4, 5, 1); |

1. **Diseño de Casos de Prueba**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** Verificar la correcta inicialización del grafo | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| Graph | Graph | setUp1 | node1 = 1  node2 = 2  node3 = 3  node4 = 4 | Se ha creado correctamente un nuevo Game que inicializa a lo que contiene en su constructor atrapando las excepciones que puede lanzar. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** Verificar que se agregue correctamente un nuevo vertice al grafo correctamente. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| Graph | addNode | setUp1 | node1 = 1  node2 = 2  node3 = 3  node4 = 4 | Se han agregado correctamente los nodos que representan los vértices, comparando los valores que cada uno tiene. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** Verificar que dos vértices se conectan correctamente. | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| Graph | connect | setUp2 | 1, 2, 3, 4, 5, 10 | Se ha creado correctamente la conexión entre ambos vértices. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** Verificar que Floyd-Warshall funcione correctamente | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| Graph | floydWarshall | setUp4 | No tiene | La matriz resultante a partir del grafo ya creado es consistente con el resultado esperado para el algoritmo. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo de la Prueba:** Verificar que Prim funcione correctamente | | | | |
| **Clase** | **Método** | **Escenario** | **Valores de Entrada** | **Resultado** |
| Graph | prim | setUp3 | No tiene | La matriz resultante a partir del grafo ya creado es consistente con el resultado esperado para el algoritmo. |