

Testing Grafo

addVertexTest:

Objetivo de la Prueba: Verificar que un vértice sea agregado al grafo.				
Clase	Método	Escenario	Valores de Entrada	Resultado
Graph	addVertex	setupStage1	Vertex.data=0; Vertex.data=1; Vertex.data=2;	El grafo contienen los tres vertices agregados.

removeVertexTest:

Objetivo de la Prueba:. Verificar que un vértice sea eliminado del grafo.				
Clase	Método	Escenario	Valores de Entrada	Resultado
Graph	removeVertex()	setupStage1	ValuetoDelete=1”	Se elimina del grafo el vértice con data=1, mientras que los vértices con data=0 y data=2 siguen aún en el grafo.

addEdgeTest:

Objetivo de la Prueba:. Verificar que se agregue correctamente una arista al grafo,				
Clase	Método	Escenario	Valores de Entrada	Resultado
Graph	addEdge() ()	setupStage1	(1,2)	Se crea una arista entre el vértice con data=1 y el vértice con data=2.

addEdgeCostTest:

Objetivo de la Prueba:. Verificar que se agregue correctamente una arista de un grafo ponderado.				
Clase	Método	Escenario	Valores de Entrada	Resultado

Graph	addEdge() (1,2,1)	setupStage1	(1,2,1)	Se crea una arista entre el vértice con data=1 y el vértice con data=2, la arista contiene un costo de 1.
-------	----------------------	-------------	---------	---

bfsTest:

Objetivo de la Prueba: Verificar el correcto funcionamiento del algoritmo bfs del grafo.				
Clase	Método	Escenario	Valores de Entrada	Resultado
Graph	bfs()	setupStage1	(grafo, 0)	El algoritmo recorre el grafo y devuelve el buscado y un arreglo con los vértices visitados.

dfsTest:

Objetivo de la Prueba: Verificar el correcto funcionamiento del algoritmo dfs del grafo.				
Clase	Método	Escenario	Valores de Entrada	Resultado
Graph	dfs()	setupStage1	(grafo, 0)	El algoritmo recorre el grafo y devuelve el buscado y un arreglo con los vértices visitados.