TEST CASES
GRAPH

Scenario configuration

Test Cases Design

Test Objective	Validate that the method addEdge is working correctly			
Class	Method	Scenario	Input Values	Result
Strcs_LA	testAddGraph()	setUpScenary1 ()	VertexLA<E> vi VertexLA<E> vf int cost (from scenary1) vi= vertexForTest1 vf = vertexForTestF cost = 0	The Vertex of type Child is added on to the graph, Which means, the previously empty graph, now is not.

Test Objective	Validate that the method search is working correctly			
Class	Method	Scenario	Input Values	Result
Strcs_LA	testSearchGraph()	setUpScenary2 ()	VertexLA<E> vi VertexLA<E> vf int cost (from scenary2) vi= vertexForTest1 vf = vertexForTestF cost = 0	The Vertex of type Child is added on to the graph, We then proceed to use the method search(VertexLA<E> v) to check if the vertexForTest1 is indeed inside the graph

Test Objective	Validate that the method removeVertex is working correctly			
Class	Method	Scenario	Input Values	Result
Strcs_LA	testDeleteGraph()	setUpScenary1 ()	VertexLA<E> vi VertexLA<E> vf int cost (from scenary1) vi= vertexForTest1 vf = vertexForTestF cost = 0	The Vertex of type Child is added on to the graph, We then proceed to use the method removeVertex(VertexLA<E> vertex) to delete the Vertex from the graph. Now

				all we need to do is check if the graph is empty, which it should be since we added and removed the Vertex from the graph

Test Objective	Validate that the method bfs is working correctly			
Class	Method	Scenario	Input Values	Result
Strcs_LA	testbfsGraph()	setUpScenary2()	VertexLA<E> vi VertexLA<E> vf int cost (from scenary2) vi= vertexForTestI vf = vertexForTestF cost = 0	The Vertex of type Child is added on to the graph, We then proceed to use the bfs(VertexLA<E> origin, Object characteristic) to organize the graph in the order given by the characteristic. This will change the order of the graph, which is determined by the graphIndex, who is in charge of having the graph ordered from the first added onto the last. So the bfs will return a different first child from what we had before using the bfs method