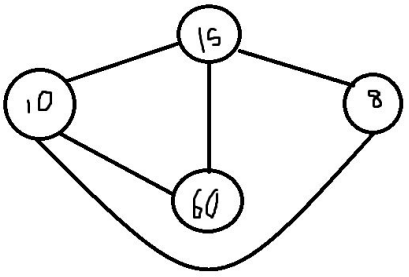
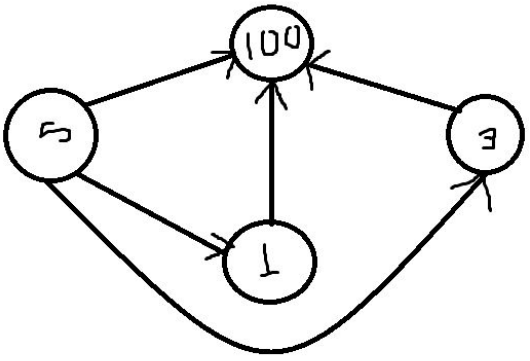
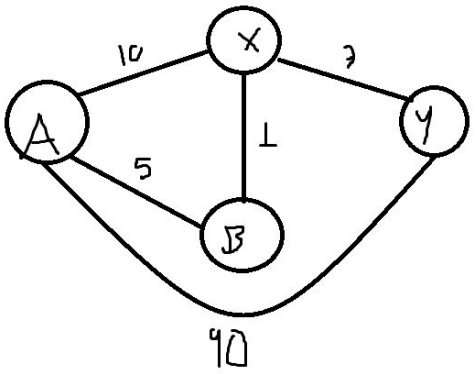
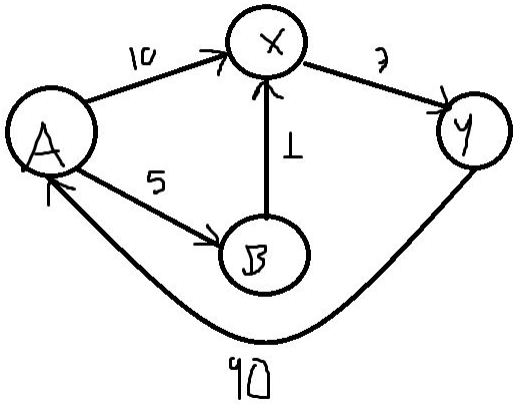
Tests Design

Graph Tests

Scenarios setup:

Name	Class	Scenario
setup1	GraphTest	A graph with vertices with the following structure 
setup2	GraphTest	A graph with vertices with the following structure 

setup3	GraphTest	A graph with vertices with the following structure 
setup4	GraphTest	A graph with vertices with the following structure 

Tests cases design:

Test objective: Verify the correct functionality of the addVertex method of the Graph.

Class	Method	Scenario	Input values	Result

Graph	addVertex	setup1	11(vertexValue) vertex[i] relatedVertices	Vertex was added to the graph and both relations were made
Graph	addVertex	setup1	12(vertexValue) vertex[i] relatedVertices	Vertex was added to the graph and both relations were made
Graph	addVertex	setup2	50(vertexValue) vertex[i] relatedVertices	Vertex was added to the graph and relations from it to relatedVertices were made
Graph	addVertex	setup2	15 (vertexValue) vertex[i] relatedVertices	Vertex was added to the graph and relations from it to relatedVertices were made
Graph	addVertex	setup3	4(vertexValue) vertex[i] relatedVertices int[i] edgeWeights	Vertex was added to the graph, both relations were made and each has the correct weight.
Graph	addVertex	setup3	50(vertexValue) vertex[i] relatedVertices int[i] edgeWeights	Vertex was added to the graph, both relations were made and each

				has the correct weight.
Graph	addVertex	setup4	9(vertexValue) vertex[i] relatedVertices int[i] edgeWeights	Vertex was added to the graph, relations from it to related vertices were made with a weight from edgeWeights.
Graph	addVertex	setup4	10(vertexValue) vertex[i] relatedVertices int[i] edgeWeights	Vertex was added to the graph, relations from it to related vertices were made with a weight from edgeWeights.

Test objective: Verify the correct functionality of the deleteVertex method of the Graph.

Class	Method	Scenario	Input values	Result
Graph	deleteVertex	setup1	10	Vertex was deleted
Graph	deleteVertex	setup1	15	Vertex was deleted
Graph	deleteVertex	setup2	1	Vertex was deleted

Graph	deleteVertex	setup2	3	Vertex was deleted
Graph	deleteVertex	setup3	x	Vertex was deleted
Graph	deleteVertex	setup3	y	Vertex was deleted
Graph	deleteVertex	setup4	b	Vertex was deleted
Graph	deleteVertex	setup4	a	Vertex was deleted

Test objective: Verify the correct functionality of the searchVertex method of the Graph.

Class	Method	Scenario	Input values	Result
Graph	searchVertex	setup1	10	Vertex was found
Graph	searchVertex	setup1	900	Vertex was not found
Graph	searchVertex	setup2	3	Vertex was found
Graph	searchVertex	setup2	1	Vertex was found

Graph	search Vertex	setup3	15	Vertex was not found
Graph	search Vertex	setup3	B	Vertex was found
Graph	search Vertex	setup4	a	Vertex was found
Graph	search Vertex	setup4	m	Vertex was not found

Test objective: Verify the correct functionality of the updateVertex method of the Graph.

Class	Method	Scenario	Input values	Result
Graph	update Vertex	setup1	10 80	Vertex with the key 10 was replaced with a key 80
Graph	update Vertex	setup1	90 1	Vertex was not found and thus not updated
Graph	update Vertex	setup2	100 3	Vertex with the key 100 was replaced with a key 3

Graph	update Vertex	setup2	900 5	Vertex was not found and thus not updated
Graph	update Vertex	setup3	A GH	Vertex with the key A was replaced with key GH
Graph	update Vertex	setup3	m 1561	Vertex was not found and thus not updated
Graph	update Vertex	setup4	Y ABC	Vertex with the key Y was replaced with key ABC
Graph	update Vertex	setup4	xz 1540	Vertex was not found and thus not updated

Test objective: Verify the correct functionality of the bFS method of the Graph.

Class	Method	Scenario	Input values	Result
Graph	bFS	setup1		10,15,60,8
Graph	bFS	setup2		5,100,1,3
Graph	bFS	setup3		A,X,B,Y

Graph	bFS	setup4		A,X,B,Y

Test objective: Verify the correct functionality of the dFS method of the Graph.

Class	Method	Scenario	Input values	Result
Graph	dFS	setup1		10,15,60,8
Graph	dFS	setup2		5,100,1,3
Graph	dFS	setup3		A,X,B,Y
Graph	dFS	setup4		A,X,Y,B

Test objective: Verify the correct functionality of the dijkstra method of the Graph.

Class	Method	Scenario	Input values	Result
Graph	dijkstra	setup3	A	v | d[v] B 5

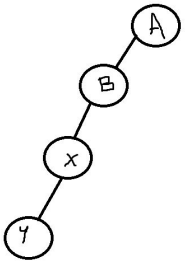
				X 6 Y 11
Graph	dijkstra	setup4	A	v | d[v] B 5 X 6 Y 11

Test objective: Verify the correct functionality of the floydWarshall method of the Graph.

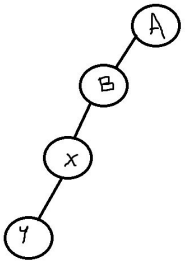
Class	Method	Scenario	Input values	Result
Graph	floydWarshall	setup3		v | d[v] B 5 X 6 Y 11
Graph	floydWarshall	setup4		v | d[v] B 5 X 6 Y 11

Test objective: Verify the correct functionality of the prim method of the Graph.

Class	Method	Scenario	Input values	Result

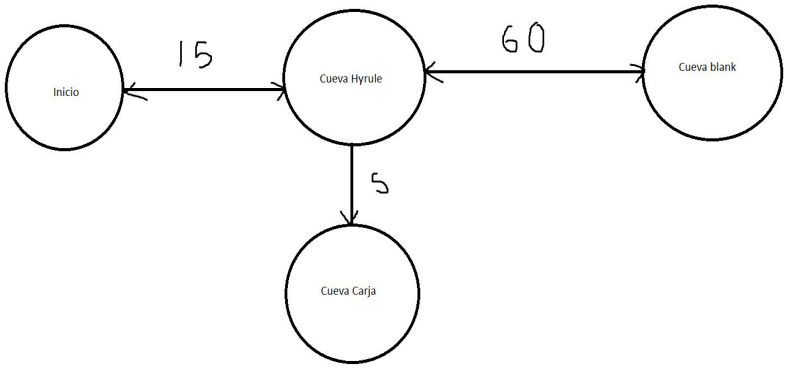
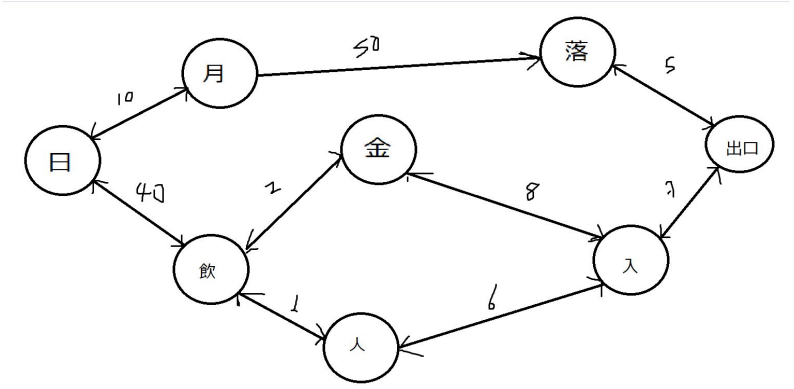
Graph	prim	setup3		A tree with the following structure 

Test objective: Verify the correct functionality of the kruskal method of the Graph.

Class	Method	Scenario	Input values	Result
Graph	kruskal	setup3		A tree with the following structure 

Solution Tests

Name	Class	Scenario

setup1	CaveSystem	A mine with the following structure 
setup2	CaveSystem	A mine with the following structure 

Test objective: Verify the correct functionality of the reachExitFast method of the Graph.

Class	Method	Scenario	Input values	Result
Graph		setup1		Exited from Cueva Carja in 20 hours
Graph		setup2		Exited from “出口” in 54 hours