
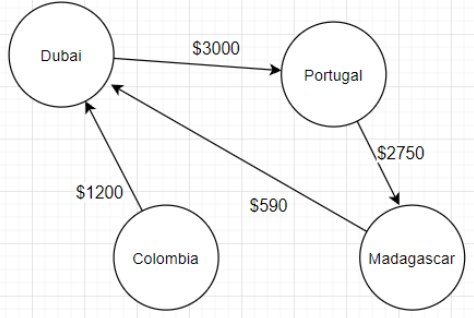
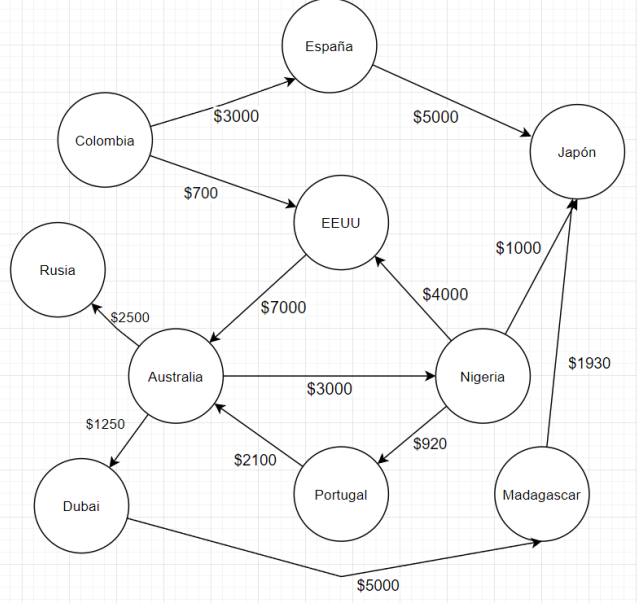
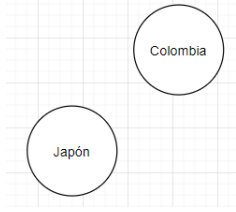
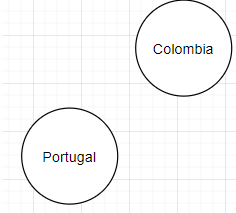


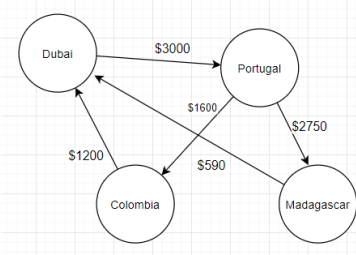
CASOS DE PRUEBA

Configuración:

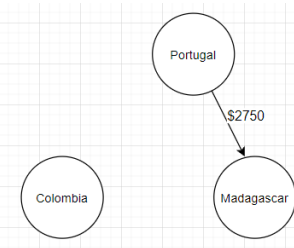
Nombre	Clase	Escenario
setupScenary1	ButterflyTest	
setupScenary2	ButterflyTest	
setupScenary3	ButterflyTest	

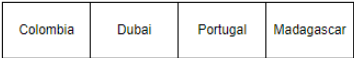

Diseño de casos de prueba para el grafo representado con matrices de adyacencia:


Objetivo de la prueba: Verificar que se agrega un vértice correctamente al grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	addVertice	setupScenary1	Vértice = "Japón"	
MatrixGraph	addVertice	setupScenary1	Vértice = "Portugal"	

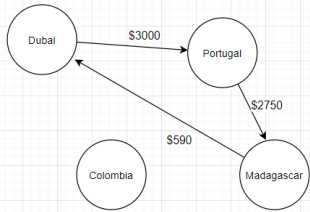
Objetivo de la prueba: Verificar que se agrega una arista correctamente al grafo de peso no negativo.				
Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	addEdge	setupScenary2	Vértice inicial = "Portugal" Vértice final = "Colombia" Costo = 1600	
MatrixGraph	addEdge	setupScenary2	Vértice inicial = "Madagascar" Vértice final = "Colombia" Costo = -200	No se pudo añadir la arista al grafo, ya que su costo es un número negativo.

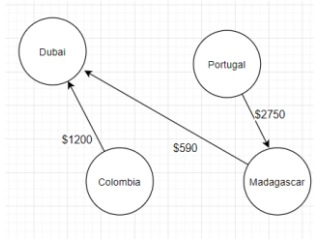
Objetivo de la prueba: Verificar que se elimina un vértice correctamente del grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	deleteVertice	setupScenary2	Vértice = "Dubai"	

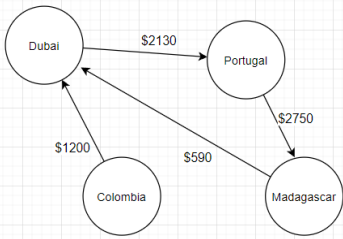
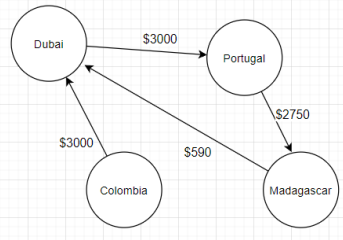
				
MatrixGraph	deleteVertice	setupScenary1	Vértice = "Colombia"	Arraylist de vértices vacía.

Objetivo de la prueba: Verificar que el algoritmo bfs funciona correctamente				
Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	bfs	setupScenary2	Vértice inicial = "Colombia"	
MatrixGraph	bfs	setupScenary3	Vértice inicial = "Colombia"	

Objetivo de la prueba: Verificar que el algoritmo dfs funciona correctamente				
Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	dfs	setupScenary3		 Size=10

Objetivo de la prueba: Verificar que se elimina una arista correctamente del grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	deleteEdge	setupScenary2	Vértice inicial = "Colombia" Vértice final = "Dubai" Costo = 1200	

MatrixGraph	deleteEdge	setupScenary2	Vértice inicial = "Dubai" Vértice final = "Portugal" Costo = 3000	

Objetivo de la prueba: Validar que se modificó una adyacencia dentro del grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	setEdge	setupScenary2	Vértice inicial = "Dubai" Vértice final = "Portugal" Costo = 2130	
MatrixGraph	setEdge	setupScenary2	Vértice inicial = "Colombia" Vértice final = "Dubai" Costo = 3000	

Objetivo de la prueba: Verificar que el algoritmo Dijkstra funciona correctamente.				
Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	makeDijsktra	setupScenary3	Vértice = "Portugal" Vértice = "Japón"	\$6100 – El costo mínimo que hay en el vuelo de Colombia a Japón.
MatrixGraph	makeDijsktra	setupScenary3	Vértice = "Portugal" Vértice = "null"	[∞, ∞, \$6100, \$9100, \$2100, \$5100, \$4600, \$0, \$3350, \$8350]

				Donde los precios corresponden a los siguientes países respectivamente: Colombia, España, Japón, EEUU, Rusia, Australia, Nigeria, Portugal, Dubai y Madagascar.
--	--	--	--	---

Objetivo de la prueba: Verificar que el algoritmo Dijkstra NO encuentra valores mínimos cuando no hay vuelos entre dos países.

Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	makeDijkstra	setupScenary3	Vértice = "Madagascar" Vértice = "Dubai"	No se encontró un costo mínimo en el vuelo de Madagascar a Dubái.
MatrixGraph	makeDijkstra	setupScenary3	Vértice = "España" Vértice = "Andorra"	No se encontró un costo mínimo en el vuelo de España a Andorra.

Objetivo de la prueba: Verificar que el algoritmo Prim funciona correctamente.

Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	prim	setupScenary3	Vértice = "Colombia"	\$28370 – El costo mínimo que hay en el vuelo de Colombia a todos los demás países.
MatrixGraph	prim	setupScenary2	Vértice = "Colombia"	\$6950 – El costo mínimo de ir desde Colombia a todos los demás países.

Objetivo de la prueba: Verificar que el algoritmo Prim NO encuentra el costo mínimo de viajar por todos los países si hay países a los que no se puede viajar.

Clase	Método	Escenario	Valores de entrada	Resultado
MatrixGraph	prim	setupScenary3	Vértice = "Madagascar"	\$1930 – Solo se encontró el valor de ir de Madagascar a Japón, ya que no se podía viajar hacia más países.
MatrixGraph	prim	setupScenary3	Vértice = "Rusia"	No se encontró un costo mínimo

			teniendo como país de origen a Rusia.
--	--	--	---------------------------------------

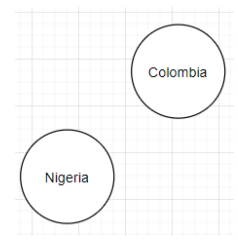
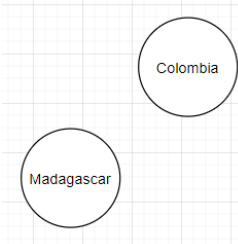
Objetivo de la prueba: Verificar que el algoritmo de floyd-warshall acomoda de forma correcta los costos de cada par de vértices.

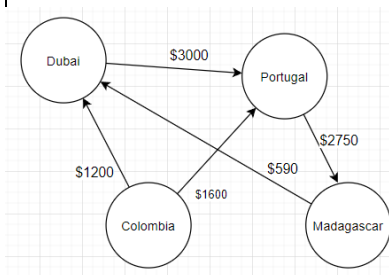
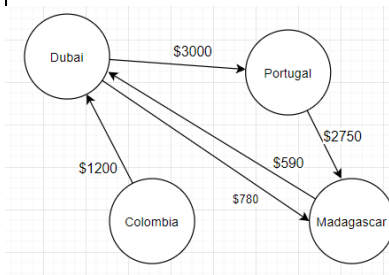
Clase	Método	Escenario	Valores de entrada	Resultado																																																																																																																									
ListGraph	floyd	setupScenary3		Se organizó correctamente la matriz con los costos de cada par de vértices obteniendo el siguiente resultado.																																																																																																																									
				<table><tr><td></td><td>Colombia</td><td>España</td><td>Japón</td><td>EEUU</td><td>Rusia</td><td>Australia</td><td>Nigeria</td><td>Dubai</td><td>Portugal</td><td>Madagascar</td></tr><tr><td>Colombia</td><td>0</td><td>3000</td><td>8000</td><td>700</td><td>10200</td><td>7700</td><td>10700</td><td>8950</td><td>11620</td><td>13950</td></tr><tr><td>España</td><td>0</td><td>0</td><td>5000</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>Japón</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>EEUU</td><td>0</td><td>0</td><td>11000</td><td>0</td><td>9500</td><td>7000</td><td>10000</td><td>8250</td><td>10920</td><td>13250</td></tr><tr><td>Rusia</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>Australia</td><td>0</td><td>0</td><td>4000</td><td>7000</td><td>2500</td><td>0</td><td>3000</td><td>1250</td><td>3920</td><td>6250</td></tr><tr><td>Nigeria</td><td>0</td><td>0</td><td>1000</td><td>4000</td><td>5520</td><td>3020</td><td>0</td><td>4270</td><td>920</td><td>9270</td></tr><tr><td>Dubai</td><td>0</td><td>0</td><td>6930</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>5000</td></tr><tr><td>Portugal</td><td>0</td><td>0</td><td>6100</td><td>9100</td><td>4600</td><td>2100</td><td>5100</td><td>3350</td><td>0</td><td>8350</td></tr><tr><td>Madagascar</td><td>0</td><td>0</td><td>1930</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>		Colombia	España	Japón	EEUU	Rusia	Australia	Nigeria	Dubai	Portugal	Madagascar	Colombia	0	3000	8000	700	10200	7700	10700	8950	11620	13950	España	0	0	5000	0	0	0	0	0	0	0	Japón	0	0	0	0	0	0	0	0	0	0	EEUU	0	0	11000	0	9500	7000	10000	8250	10920	13250	Rusia	0	0	0	0	0	0	0	0	0	0	Australia	0	0	4000	7000	2500	0	3000	1250	3920	6250	Nigeria	0	0	1000	4000	5520	3020	0	4270	920	9270	Dubai	0	0	6930	0	0	0	0	0	0	5000	Portugal	0	0	6100	9100	4600	2100	5100	3350	0	8350	Madagascar	0	0	1930	0	0	0	0	0	0	0
					Colombia	España	Japón	EEUU	Rusia	Australia	Nigeria	Dubai	Portugal	Madagascar																																																																																																															
				Colombia	0	3000	8000	700	10200	7700	10700	8950	11620	13950																																																																																																															
				España	0	0	5000	0	0	0	0	0	0	0																																																																																																															
				Japón	0	0	0	0	0	0	0	0	0	0																																																																																																															
				EEUU	0	0	11000	0	9500	7000	10000	8250	10920	13250																																																																																																															
				Rusia	0	0	0	0	0	0	0	0	0	0																																																																																																															
				Australia	0	0	4000	7000	2500	0	3000	1250	3920	6250																																																																																																															
				Nigeria	0	0	1000	4000	5520	3020	0	4270	920	9270																																																																																																															
				Dubai	0	0	6930	0	0	0	0	0	0	5000																																																																																																															
				Portugal	0	0	6100	9100	4600	2100	5100	3350	0	8350																																																																																																															
Madagascar	0	0	1930	0	0	0	0	0	0	0																																																																																																																			
ListGraph	floyd	setupScenary1		La matriz utilizada por el algoritmo no se encuentra inicializada ya que no hay un grafo en el cual trabajar.																																																																																																																									
ListGraph	floyd	setupScenary2		Se organizó correctamente la matriz con los costos de cada par de vértices obteniendo el siguiente resultado.																																																																																																																									
				<table><tr><td></td><td>Dubai</td><td>Portugal</td><td>Colombia</td><td>Madagascar</td></tr><tr><td>Dubai</td><td>0</td><td>3000</td><td>0</td><td>5750</td></tr><tr><td>Portugal</td><td>3340</td><td>0</td><td>0</td><td>2750</td></tr><tr><td>Colombia</td><td>1200</td><td>4200</td><td>0</td><td>6950</td></tr><tr><td>Madagascar</td><td>590</td><td>3590</td><td>0</td><td>0</td></tr></table>		Dubai	Portugal	Colombia	Madagascar	Dubai	0	3000	0	5750	Portugal	3340	0	0	2750	Colombia	1200	4200	0	6950	Madagascar	590	3590	0	0																																																																																																
					Dubai	Portugal	Colombia	Madagascar																																																																																																																					
				Dubai	0	3000	0	5750																																																																																																																					
				Portugal	3340	0	0	2750																																																																																																																					
Colombia	1200	4200	0	6950																																																																																																																									
Madagascar	590	3590	0	0																																																																																																																									

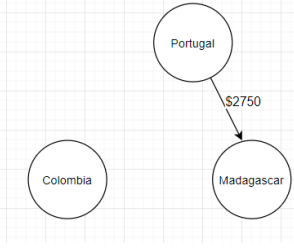
Objetivo de la prueba: Validar que el algoritmo de kruskal calcula el menor recorrido entre todos los vértices

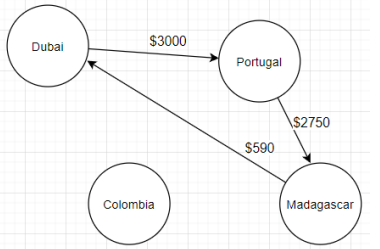
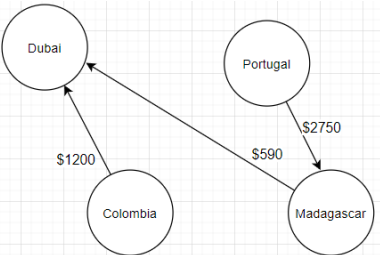
Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	kruskal	setupScenary3		Retorna el valor de mínimo de recorrer todos los países int = 17.400
ListGraph	kruskal	setupScenary1		No hay un grafo en el cuál trabajar int = 0
ListGraph	kruskal	setupScenary2		Retorna el valor mínimo de recorrer todos los países. int = 4.540

Diseño de casos de prueba para el grafo representado con listas de adyacencia:

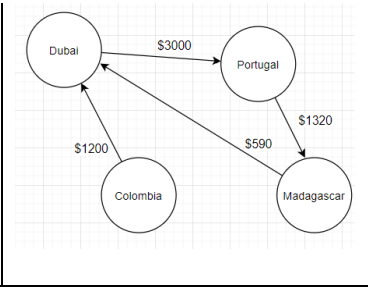
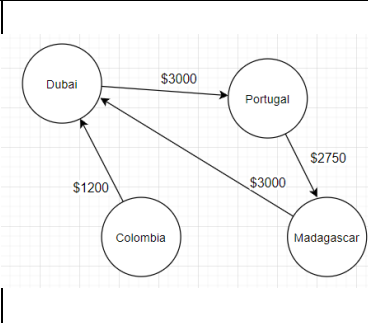
Objetivo de la prueba: Verificar que se agrega un vértice correctamente al grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	addVertice	setupScenary1	Vértice = "Nigeria"	
ListGraph	addVertice	setupScenary1	Vértice = "Madagascar"	

Objetivo de la prueba: Verificar que se agrega una arista correctamente al grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	addEdge	setupScenary2	Vértice inicial = "Colombia" Vértice final = "Portugal" Costo = 1600	
ListGraph	addEdge	setupScenary2	Vértice inicial = "Dubai" Vértice final = "Madagascar" Costo = 780	

Objetivo de la prueba: Verificar que se elimina un vértice correctamente del grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	deleteVertice	setupScenary2	Vértice = "Dubai"	
ListGraph	deleteVertice	setupScenary1	Vértice = "Colombia"	El arraylist de vértices está vacío.

Objetivo de la prueba: Verificar que se elimina una arista correctamente del grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	deleteEdge	setupScenary2	Vértice inicial = "Colombia" Vértice final = "Dubai" Costo = 1200	
ListGraph	deleteEdge	setupScenary2	Vértice inicial = "Dubai" Vértice final = "Portugal" Costo = 3000	

Objetivo de la prueba: Validar que se modificó una adyacencia dentro del grafo.				
Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	setEdge	setupScenary2	Vértice inicial = "Portugal" Vértice final = "Madagascar" Costo = 1320	

				
ListGraph	setEdge	setupScenary2	Vértice inicial = "Madagascar" Vértice final = "Dubai" Costo = 3000	

Objetivo de la prueba: Verificar que el algoritmo Dijkstra funciona correctamente.				
Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	makeDijsktra	setupScenary3	Vértice = "Portugal" Vértice = "Japón"	\$6100 – El costo mínimo que hay en el vuelo de Colombia a Japón.
ListGraph	dijsktra	setupScenary3	Vértice = "Portugal" Vértice = "null"	<p>[∞, ∞, \$6100, \$9100, \$4600, \$2100, \$5100, \$3350, \$0, \$8350]</p> <p>Donde los precios corresponden a los siguientes países respectivamente: Colombia, España, Japón, EEUU, Rusia, Australia, Nigeria, Portugal, Dubai y Madagascar.</p>

Objetivo de la prueba: Verificar que el algoritmo Dijkstra NO encuentra valores mínimos cuando no hay vuelos entre dos países.

Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	makeDijkstra	setupScenary3	Vértice = "Madagascar" Vértice = "Dubai"	No se encontró un costo mínimo en el vuelo de Madagascar a Dubái.
ListGraph	makeDijkstra	setupScenary3	Vértice = "España" Vértice = "Andorra"	No se encontró un costo mínimo en el vuelo de España a Andorra.

Objetivo de la prueba: Verificar que el algoritmo Prim funciona correctamente.

Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	prim	setupScenary3	Vértice = " Colombia"	\$28370 – El costo mínimo que hay en el vuelo de Colombia a todos los demás países.
ListGraph	prim	setupScenary2	Vértice = "Colombia"	\$6950 – El costo mínimo de ir desde Colombia a todos los demás países.

Objetivo de la prueba: Verificar que el algoritmo Prim NO encuentra el costo mínimo de viajar por todos los países si hay países a los que no se puede viajar.

Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	prim	setupScenary3	Vértice = "Madagascar"	\$1930 – Solo se encontró el valor de ir de Madagascar a Japón, ya que no se podía viajar hacia más países.
ListGraph	prim	setupScenary3	Vértice = "Rusia"	No se encontró un costo mínimo teniendo como país de origen a Rusia.

Objetivo de la prueba: Verificar que el algoritmo de floyd-warshall acomoda de forma correcta los costos de cada par de vértices.

Clase	Método	Escenario	Valores de entrada	Resultado
-------	--------	-----------	--------------------	-----------

ListGraph	floyd	setupScenary3	Se organizó correctamente la matriz con los costos de cada par de vértices obteniendo el siguiente resultado.										
				Colombia	España	Japón	EEUU	Rusia	Australia	Nigeria	Dubai	Portugal	Madagascar
			Colombia	0	3000	8000	700	10200	7700	10700	8950	11620	13950
			España	0	0	5000	0	0	0	0	0	0	0
			Japón	0	0	0	0	0	0	0	0	0	0
			EEUU	0	0	11000	0	9500	7000	10000	8250	10920	13250
			Rusia	0	0	0	0	0	0	0	0	0	0
			Australia	0	0	4000	7000	2500	0	3000	1250	3920	6250
			Nigeria	0	0	1000	4000	5520	3020	0	4270	920	9270
			Dubai	0	0	6930	0	0	0	0	0	0	5000
			Portugal	0	0	6100	9100	4600	2100	5100	3350	0	8350
			Madagascar	0	0	1930	0	0	0	0	0	0	0
ListGraph	floyd	setupScenary1	La matriz utilizada por el algoritmo no se encuentra inicializada ya que no hay un grafo en el cual trabajar.										
ListGraph	floyd	setupScenary2	Se organizó correctamente la matriz con los costos de cada par de vértices obteniendo el siguiente resultado.										
				Dubai	Portugal	Colombia	Madagascar						
			Dubai	0	3000	0	5750						
			Portugal	3340	0	0	2750						
			Colombia	1200	4200	0	6950						
			Madagascar	590	3590	0	0						

Objetivo de la prueba: Validar que el algoritmo de kruskal calcula el menor recorrido entre todos los vértices

Clase	Método	Escenario	Valores de entrada	Resultado
ListGraph	kruskal	setupScenary3		Retorna el valor de mínimo de recorrer todos los países int = 17.400
ListGraph	kruskal	setupScenary1		No hay un grafo en el cuál trabajar int = 0
ListGraph	kruskal	setupScenary2		Retorna el valor mínimo de recorrer todos los países. int = 4.540

Objetivo de la prueba: Verificar que el algoritmo bfs funciona correctamente

Clase	Método	Escenario	Valores de entrada	Resultado

ListGraph	bfs	setupScenary2	Vértice inicial =" Colombia"	<table><tr><td>Colombia</td><td>Dubai</td><td>Portugal</td><td>Madagascar</td></tr></table>	Colombia	Dubai	Portugal	Madagascar						
Colombia	Dubai	Portugal	Madagascar											
ListGraph	bfs	setupScenary3	Vértice inicial =" Colombia"	<table><tr><td>Colombia</td><td>España</td><td>EEUU</td><td>Japon</td><td>Australia</td><td>Nigeria</td><td>Rusia</td><td>Dubai</td><td>Portugal</td><td>Madagascar</td></tr></table>	Colombia	España	EEUU	Japon	Australia	Nigeria	Rusia	Dubai	Portugal	Madagascar
Colombia	España	EEUU	Japon	Australia	Nigeria	Rusia	Dubai	Portugal	Madagascar					

Objetivo de la prueba: Verificar que el algoritmo dfs funciona correctamente																				
Clase	Método	Escenario	Valores de entrada	Resultado																
ListGraph	dfs	setupScenary3		<table><tr><td>Colombia</td><td>España</td><td>EEUU</td><td>Japon</td><td>Australia</td><td>Nigeria</td><td>Rusia</td><td>Dubai</td><td>Portugal</td><td>Madagascar</td></tr></table> <p>Size=10</p>							Colombia	España	EEUU	Japon	Australia	Nigeria	Rusia	Dubai	Portugal	Madagascar
Colombia	España	EEUU	Japon	Australia	Nigeria	Rusia	Dubai	Portugal	Madagascar											