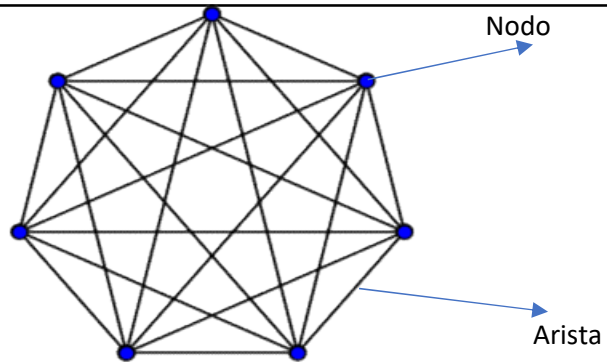


## TAD GRAFO



{inv:  
→: está conectado  
 $\forall \text{nodo} \rightarrow \forall \text{Nodo}$ }

Operaciones primitivas:

Grafo:

InsertarNodo: String

EliminarNodo: String

EditarNodo: String x String

BuscarNodo: String                      Nodo

FloydWarshall: String                      Nodos[]



<b>Grafo()</b>
Crea un grafo de tamaño indeterminado sin nodos (vacío)
{pre: TRUE}
{pos: grafo=[]}
<b>InsertarNodo(ciudad)</b>
Inserta un nodo con el nombre de una ciudad al grafo, y establece una arista con el resto de nodos.
{pre: ciudad ∈ String}
{pos: grafo.nodo(ciudad)}
<b>EliminarNodo(ciudad)</b>
Busca el nodo, cuyo identificador coincida con *ciudad*
{pre: ciudad ∈ String}
{post: [(if <nodo(ciudad) ∈ grafo>): TRUE] (else: FALSE)]}
<b>BuscarNodo(ciudad)</b>
Dentro del grafo, busca al nodo cuyo identificador coincida con *ciudad*

{pre: ciudad ∈ String}
{post: [(if <nodo(ciudad) ∈ grafo>): nodo(ciudad)) (else: null)]}
EditarNodo( ciudadNodoBuscado, ciudad)
Cambia el identificador del nodo buscado, al deseado *ciudad*
{pre: ciudad, ciudadNodoBuscado ∈ String}
{post: [(if <nodo(ciudadNodoBuscado) ∈ grafo>):TRUE) (else: FALSE)]}
FloydWarshall(ciudad)
Busca el camino más corto para recorrer el grafo
{pre: ciudad ∈ String}
{pos: nodeList[]}