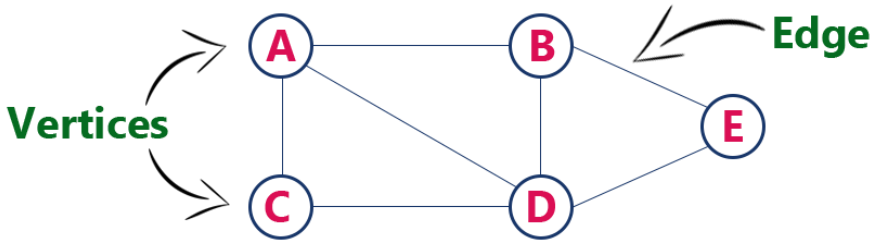


## TAD – TIPO ABSTRACTO DE DATOS

TAD GRAFO		
<b>Representación:</b> Grafo = { <b>Vértices</b> = $\langle V_0, V_1, V_2, \dots, V_i \rangle$ <b>Aristas</b> = $\langle E_0, E_1, E_2, \dots, E_j \rangle$		
		
<b>Invariante:</b>		
<b>Operaciones:</b>		
CreateGraph		→ Graph
AddVertice	Vertice	→ Boolean
DeleteVertice	Vertice	→ Boolean
AddEdge	Vertice1, Vertice2	→ Boolean
BFS	Vertice	→ Tree
DFS	Vertice	→ Tree

CreateGraph()
<b>Pre:</b> null
<b>Post:</b> se crea un grafo vacío

### AddVertice(Vértice)

**Pre:** null

**Post:** se agrega el vértice al grafo

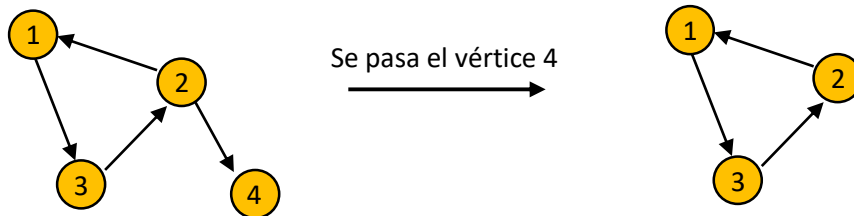
$$\text{Vértices} = V_n + 1$$



### DeleteVertice(Vértice)

**Pre:** Vértice

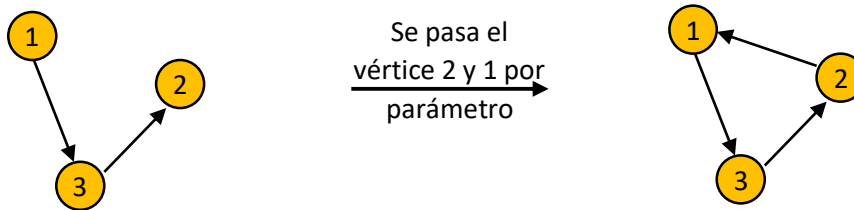
**Post:** se elimina el vértice que se pasa por parámetro



### AddEdge(Vertex1, Vertex2)

**Pre:** Vértice1 y vértice 2 ya existen

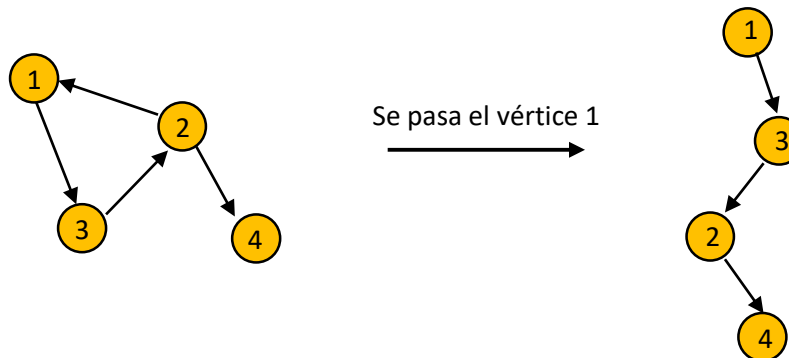
**Post:** se agrega una arista entre los vértices que se pasaron por parámetro



### BFS(Vertex)

**Pre:** Vértices y aristas

**Post:** Luego de recorrer el grafo respecto a su anchura comenzando por el vértice que se pasa por parámetro, y colocando a este como el padre de todos sus nodos adyacentes, se crea un árbol que representa los nodos recorridos.



## DFS(Vertex)

**Pre:** Vértices y aristas

**Post:** Luego de recorrer el grafo a profundidad desde el grafo pasado por parámetro, y este es el que se asigna como padre del árbol.

