|  |
| --- |
| **TAD Graph** |
| Graph<V,E> = {V v=null, E e=null, int w=0} |
| { inv : } |
| Operaciones Primitivas   * Graph Ninguno 🡪 Graph * insertVertex&Edge Object, Object , int 🡪 Graph * dijkistra Object , object 🡪 int * bfs Object 🡪 boolean |

Build

|  |
| --- |
| Graph()  “Initialize the constructor of the Graph class”  {pre : none}  { post : Graph= { v=null,e=null,w=0 } } |

Modifier

|  |
| --- |
| insert(V v, E e, int w)  “Connect two vertices by an edge with a certain weight”  {pre : w!=0,v!=null,e!=null}  { post : Graph= { v connected with 3 with an edge of weight w} |

Analyzer

|  |
| --- |
| dijkistra (V v, V v)  “From vertex v, travel the minimum distance to another vertex v”  {pre : v!=null, v!=null}  { post : int= { distance between both vertices } |

|  |
| --- |
| bfs (V v)  “Traces the graph to determine if the vertex is contained in the graph”  {pre : v!null}  { post : boolean= { true if the vertex is in the graph, false if not} |