**ADT Graph**

|  |
| --- |
| **ADT Graph <T>** |
| Graph = { Directed = <directed>, Weighted = <weighted>, Vertices = <vertices>, Edges = <edges> } |
| { inv: } |
| Primitive Operations:   * Graph: directed x weighted 🡪 Graph<T> * getVertices: Graph<T> 🡪 List of vertices * isDirected: Graph<T> 🡪 Boolean * isWeighted: Graph<T> 🡪 Boolean * addVertex: Graph<T> x T 🡪 Graph<T> * addEdge: Graph<T> x T x T 🡪 Graph<T> * removeVertex: Graph<T> x T 🡪 Graph<T> * removeEdge: Graph<T> x T x T 🡪 Graph<T> * getNeighborts: Graph<T> x Vertex<T> 🡪 List of vertices * getNumberOfVertices: Graph<T> 🡪 Number * getNumberOfEdge: Graph<T> 🡪 Number * areAdjacent: Graph<T> x Vertex<T> x Vertex<T> 🡪 Boolean * isInGraph: Graph<T> x T 🡪 Boolean * getEdgeWeight: Graph<T> x Vertex<T> x Vertex<T> 🡪 double * setEdgeWeight: Graph<T> x Vertex<T> x Vertex<T> x double 🡪 Graph * bfs: Graph<T> x Vertex<T> 🡪 List of vertices * dfs: Graph<T> 🡪 List of vertices * dijkstra: Graph<T> x Vertex<T> 🡪 List of vertices * floydWarshall: Graph<T> 🡪 Matrix of double * prim: Graph<T> x Vertex<T> 🡪 Graph<T> * kruskal: Graph<T> 🡪 List of edges * searchVertex: Graph<T> x T 🡪 Vertex<T> * getEdges: Graph<T> 🡪 List of edges * getContests: Graph<T> 🡪 List of T |

|  |
| --- |
| **Graph(directed, weighted)**  “Create a new Graph without edges”  { pre: TRUE ⋀ directed ∈ Boolean ⋀ weighted ∈ Boolean}  { post: graph = { Directed = directed, Weighted = weighted, Vertices = 0, Edge = 0} } |

|  |
| --- |
| **getVertices(graph)**  “Returns a collections of vertices”  {pre: graph = {} }  {post: graph = { } } |

|  |
| --- |
| **isDirected()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **isWeighted()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **addVertex()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **addEdge()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **removeVertex()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **removeEdge()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **getNeighborts()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **getNumberOfVertices()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **getNumberOfEdge()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **areAdjacent()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **isInGraph()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **getEdgeWeight()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **setEdgeWeight()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **bfs()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **dfs()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **dijkstra()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **floydWarshall()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **prim()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **kruskal()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **searchVertex**(**)**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **getEdges()**  “ ”  {pre: TRUE}  {post: graph = { } } |

|  |
| --- |
| **getContests()**  “ ”  {pre: TRUE}  {post: graph = { } } |