|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Graph | | | | |
| G= (V, E)      V= {A, B, C, D, E, F, H, I}  𝐸={(𝐴,𝐺),(𝐴,𝐶),(𝐶,𝐻),(𝐷,𝐵),(𝐷,𝐼)…} | | | | |
| Note: The number on each of the edges is the weight of the respective edge, which can be any value as long as it is an integer. Weight = w | | | | |
| 𝑖𝑛𝑣={𝑛𝑜 𝑙𝑜𝑝𝑠}  𝑖𝑛𝑣={∀ 𝑤 {𝑤| 𝑤>0}}  𝑖𝑛𝑣={𝑉 != ∅}  𝑖𝑛𝑣={𝐸 !=∅}  𝑖𝑛𝑣={𝑑𝑒𝑟𝑖𝑔𝑔𝑒𝑑} | | | | |
| Primitive Operations | | | | |
|  | Graph | … | Graph |  |
| Create Vertex | Graph x Pos | Graph x Vertex |
| Create Edge | Graph x V1 x V2 | Graph x Edge |
| Dijkstra Path | Graph x V1 x V2 | Graph x VertexList |
| Dijkstra Amount | Graph x VertexList | Integer |
| Floyd Warshall | Graph | Graph x VertexList |

|  |
| --- |
| Create Vertex(Pos): Creator |
| Creates a new Vertex in the Graph, this vertex must have an unique identification |
| Pre ={true,Graph} Pos= {a new vertex without conexion} |

|  |
| --- |
| Graph (): Constructor |
| Creates a new Simple Graph |
| Pre = {true} Pos = { Graph } |

|  |
| --- |
| Create Edge(V vertex1, V vertex2, W weight) : Creator |
| Creates a new edge between two vertexes, this edge has a weight and its an integer. |
| 𝑝𝑟𝑒 = {𝑣𝑒𝑟𝑡𝑒𝑥1, 𝑣𝑒𝑟𝑡𝑒𝑥2}  𝑝𝑜𝑠 = {𝑒𝑑𝑔𝑒 𝑏𝑒𝑡𝑤𝑒𝑒𝑛 𝑣𝑒𝑟𝑡𝑒𝑥1, 𝑣𝑒𝑟𝑡𝑒𝑥2} |

|  |
| --- |
| Dijkstra Path(V vertex1, Vertex2) : Analyzer |
| Visit all the paths between the vertex1(Source) and vertex2, evaluate any paths to know which one is the shortest one between those vertexes, and return a list with the vertexes of the  path |
| 𝑝𝑟𝑒 = {𝑡𝑟𝑢𝑒, 𝑣𝑒𝑟𝑡𝑒𝑥1, 𝑣𝑒𝑟𝑡𝑒𝑥2}  𝑝𝑜𝑠 = {𝑙𝑖𝑠𝑡 𝑤𝑖𝑡ℎ 𝑣𝑒𝑟𝑡𝑒𝑥𝑒𝑠} |

|  |
| --- |
| Floyd Warshall( ) : Analyzer |
| Evaluates all paths between each vertex to find the shortest path between each pair of vertexes |
| 𝑝𝑟𝑒 = {𝑡𝑟𝑢𝑒}  𝑝𝑜𝑠 = {𝑙𝑖𝑠𝑡 𝑤𝑖𝑡ℎ 𝑣𝑒𝑟𝑡𝑒𝑥𝑒𝑠} |