| TAD <Graph > |
| --- |
| Graph={arrayList} |
| Inv: {vertice can’t be null} |
| Primitive Operations:  CreateArraylist(): -> void  AddVertice(name,IP): Vertice-> Void  RemoveVertice(IP): Graph -> Vertice  AddEdge(IP1, IP2): Connection -> Void  RemoveEdge(IP1, IP2): GraphxVertice -> Void  Find(IP): -> Vertice  FindShortestPath(IP1, IP2, Wieght): ->ArrayList |

**Graph**

| CreateArraylist()  “Creates a new graph arraylist which includes vertices and inside them their edges”    {pre: void }    {pos: “Successful Creation”} |
| --- |

| AddVertice(Name,IP)  “Creates a new vertice and adds it to graph”    {pre: Name, IP}    {pos: Vertice} |
| --- |

| RemoveVertice(IP)  “Removes vertice from graph and all its connections”    {pre: IP}    {pos: Vertice} |
| --- |

| AddEdge(IP1, IP2)  “Adds edge between two vertices”    {pre: IP1,IP2}    {pos: Vertice1, Vertice2} |
| --- |

| RemoveEdge(IP1, IP2)  “Removes edge between two vertices”    {pre: IP1,IP2}    {pos: Vertice1, Vertice2} |
| --- |

| Find()  “Finds Vertice”    {pre: IP}    {pos: Vertice} |
| --- |

| FindShortestPath(IP1, IP2, Weight)  “Finds shortest path between two vertices when sending a certain weight”    {pre: IP1, IP2, Weight}    {pos: ArrayList} |
| --- |