

Pseudocode:

```

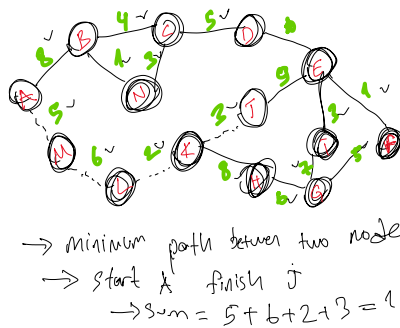
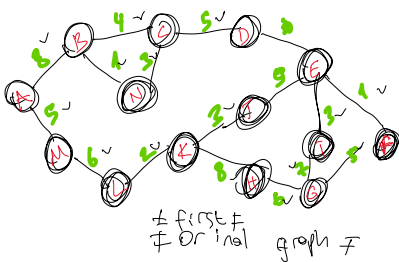
int firstNode,lastNode,pathNext,pathVal;
int sum=0
TNode * createNode(int nameNode,int pathLeft,int pathRight,int pathDown,int pathUp){
TNode *newNode = new TNode;
newNode->pathLeft = NULL;
newNode->pathRight = NULL;
newNode->pathDown = NULL;
newNode->pathUp = NULL;
NewNode->nameNode=NULL
return newNode;
}
While(firstNode!=LastNode){
FindSmallPath( pathLeft,pathRight,pathDown,pathUp,pathNext,pathVal)
Node->pathNext;
Sum=pathVal+sum
Return sum}

```

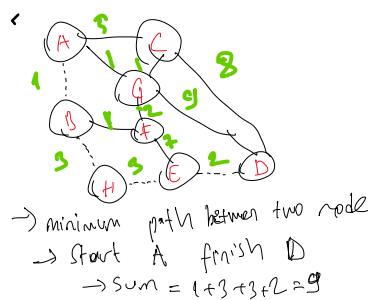
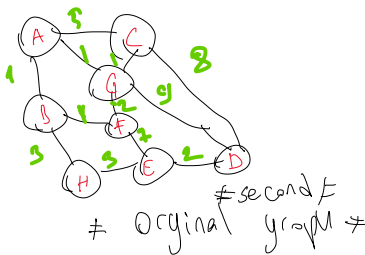
Purpose and Works of Algorithm

My algorithm purpose is going to one node to another node with smallest path . For solution this problem, classify paths randomly like left,right,up,down and given the path value. Users give first and last node name. Algorithm tends to going to shortest path. For example, my left path value is 3 and right path value is 1. Algorithm choose the smallest value of path which is 1. Then algorithm check every node name for finds destination.

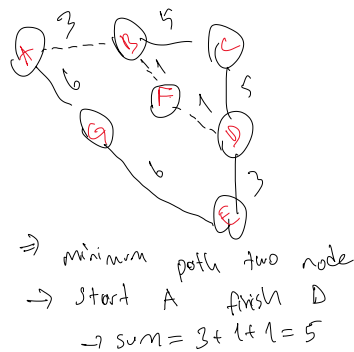
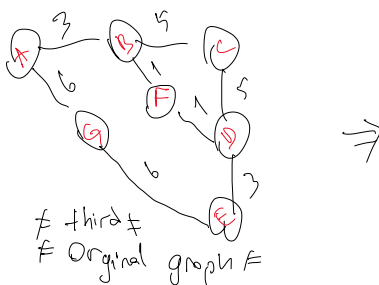
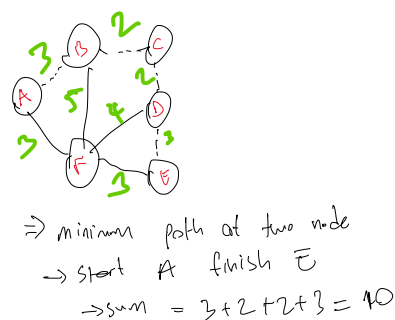
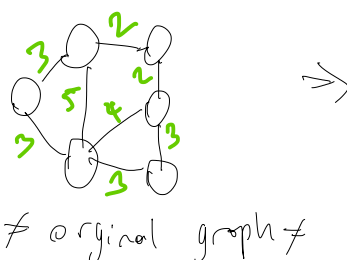
Note: Dash way is algorithm running way



In this graph, my algorithm is working because there is no same path value



In these two graphs, my algorithm is not working effectively because these paths don't have the one the smallest path value. My algorithm just decide one node minimum path solution. This decision could be going to better or worst. That is not stable. Thanks to this way, Kruskal algorithm is better than my algorithm

Question Graph

When we discuss two algorithm, my algorithm can not decide the sortest way because algorithm want to go shortest way. In this reason, shortest way is not optimal solution because next node path could be increase total path value(sum).