Elementary Graph Algorithms

Module 1

DFS(Graph G) -- choose start vertex arbitrarily and perform DFS

DFS(Graph G, vertex s)

DFS(Graph G, vertex s, vertex t) -- is t reachable from s

Module 2

BFS(Graph G)

BFS(Graph G, vertex s)  
BFS(Graph G, vertex s, vertex t)

Module 3

BFS\_SPATH(G,s,flag=0) -- returns spath in unweighted graphs  
BFS\_SPATH(G,s,flag=1) -- returns spath in weighted graphs

LPATH(Tree T, flag=0) - returns the longest path in T (undirected tree)

LPATH(Tree T, flag=1) - returns the longest path in a DAG

MAX\_Weight\_MST(Graph G)

Module 4

Test\_bipartite(Graph G)

Test\_articulationpoints(Graph G)

Test\_bridges(Graph G)

Module 5

MST\_Prim(Graph G)

MST\_Kruskal(Graph G)

MST\_youralgo(Graph G)

Instructions:

1. do not change function names for any of the above, also the signatures (arguments).  Learn about overloading/overriding.

2. use makefile, use .h, .cpp

3. DFS must be coded once and must be reused wherever needed.  you should not replicate the same code.  Invoke appropriate .h file instead of replicating the code