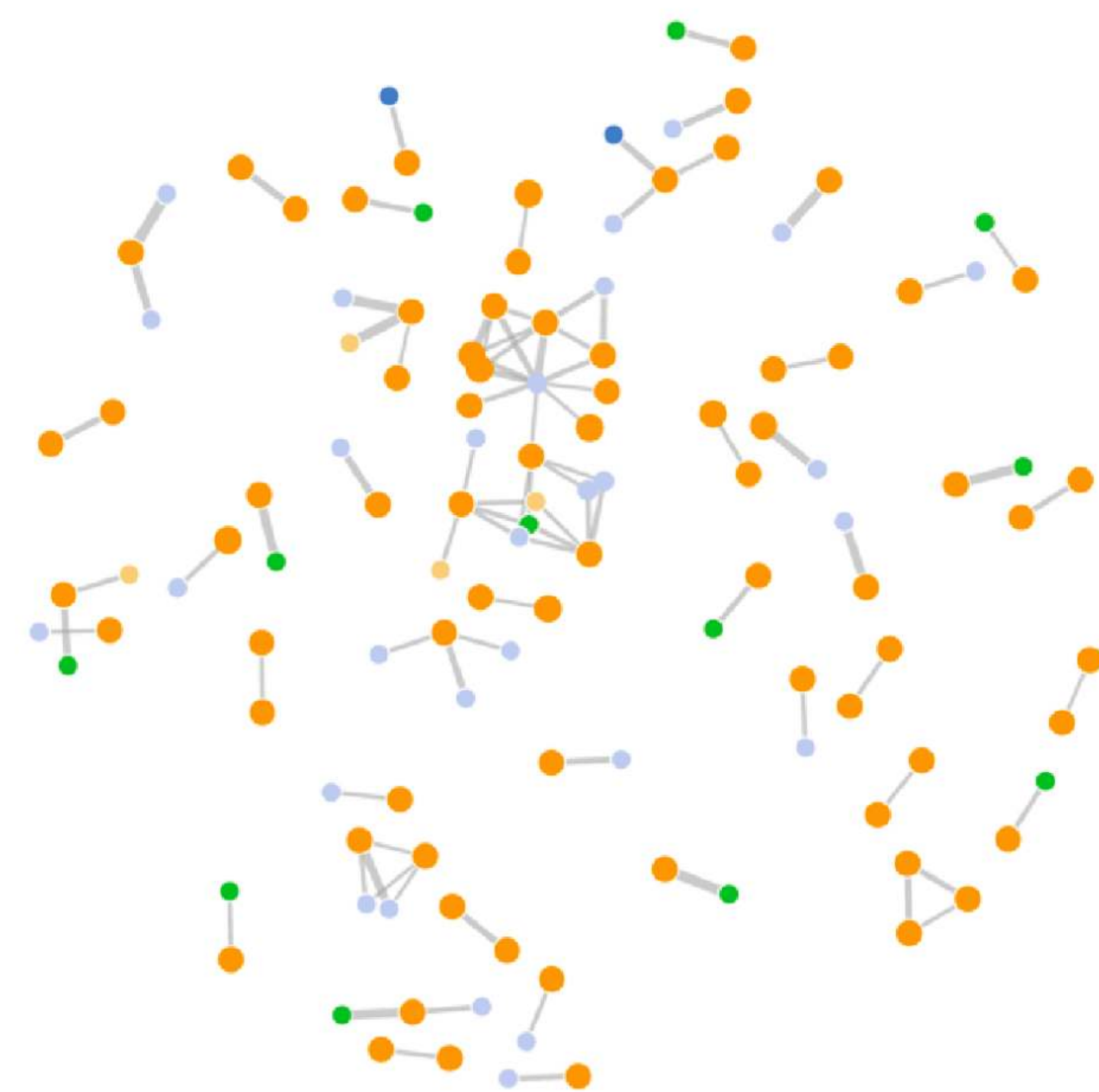
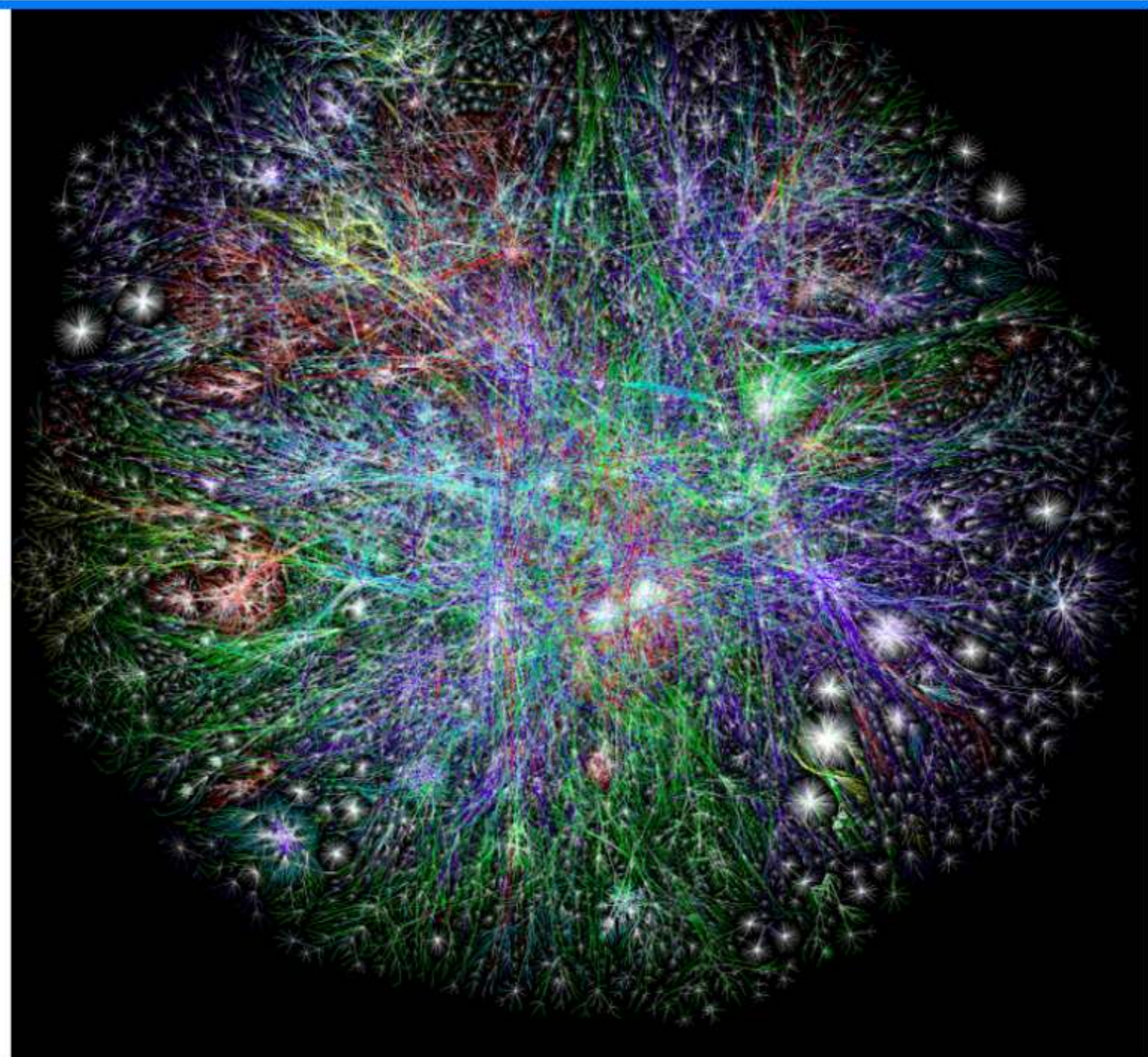


Announcements

- * MP7 is out, on Tue, 05/03 @ 11:59pm; E/C due on Tue, 04/26 @ 11:59pm
- * Lab_graphs - Sun, 05/01
- * **Final Exam: 05/09 1:30-4:30pm**

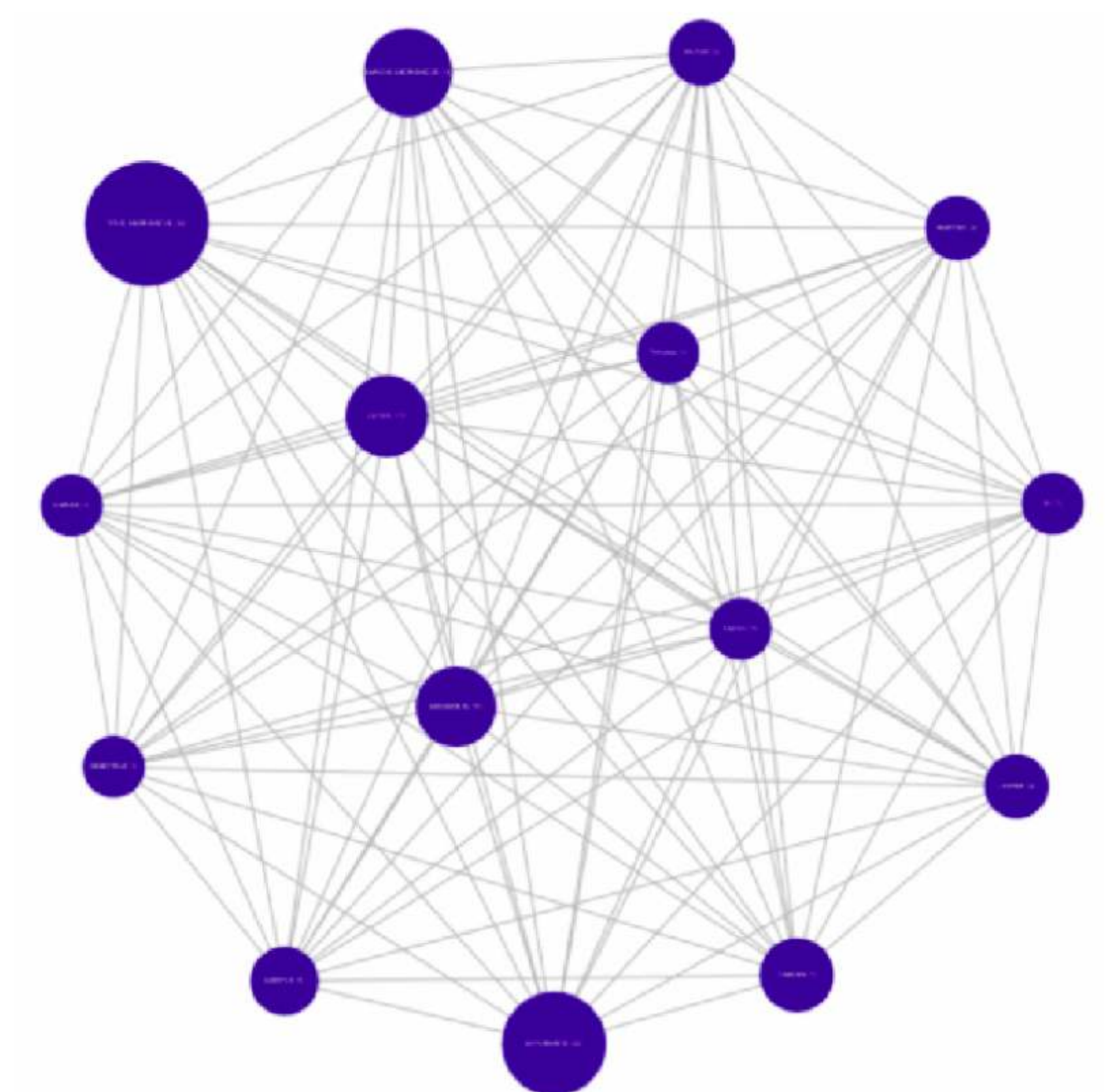
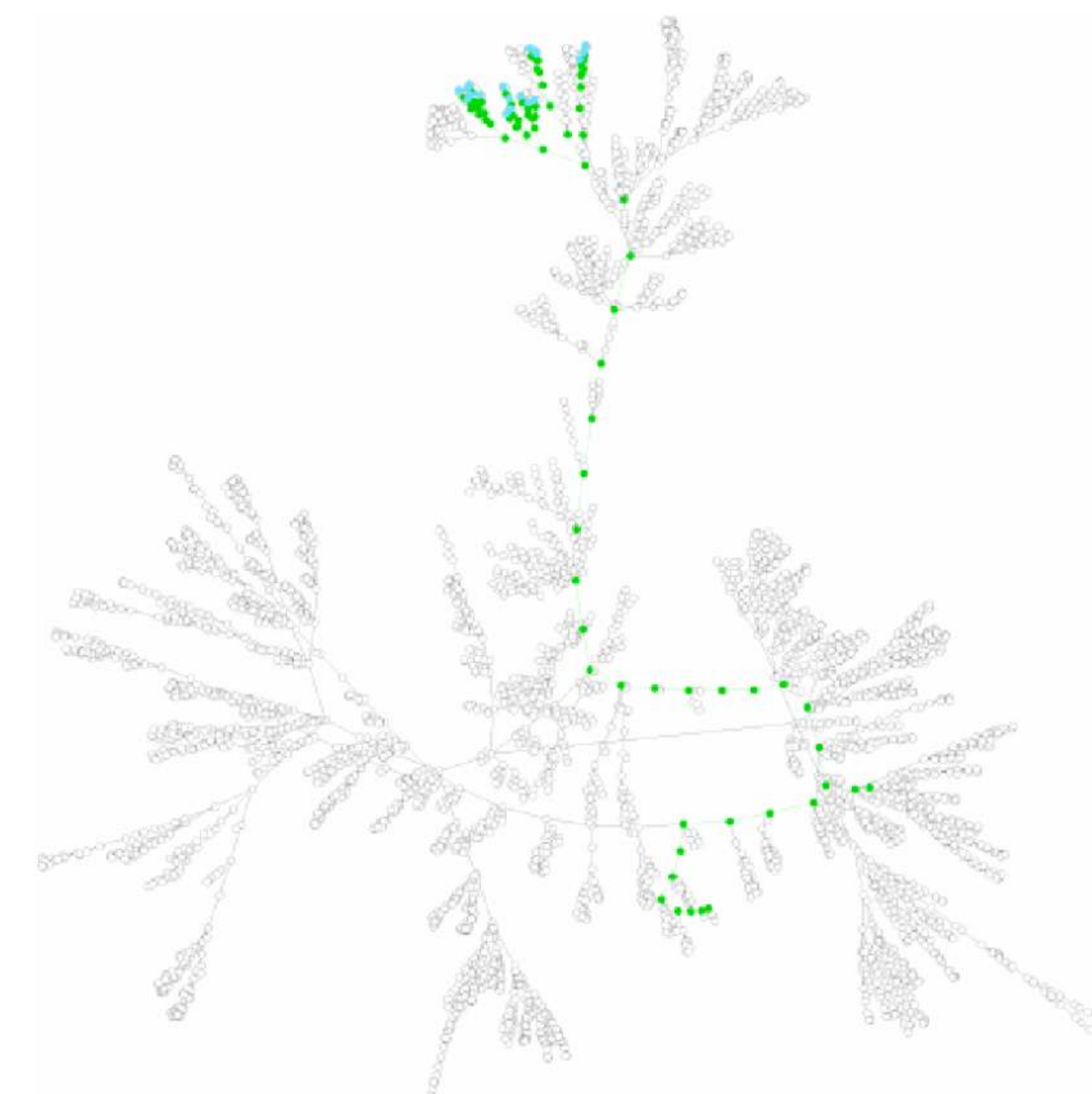
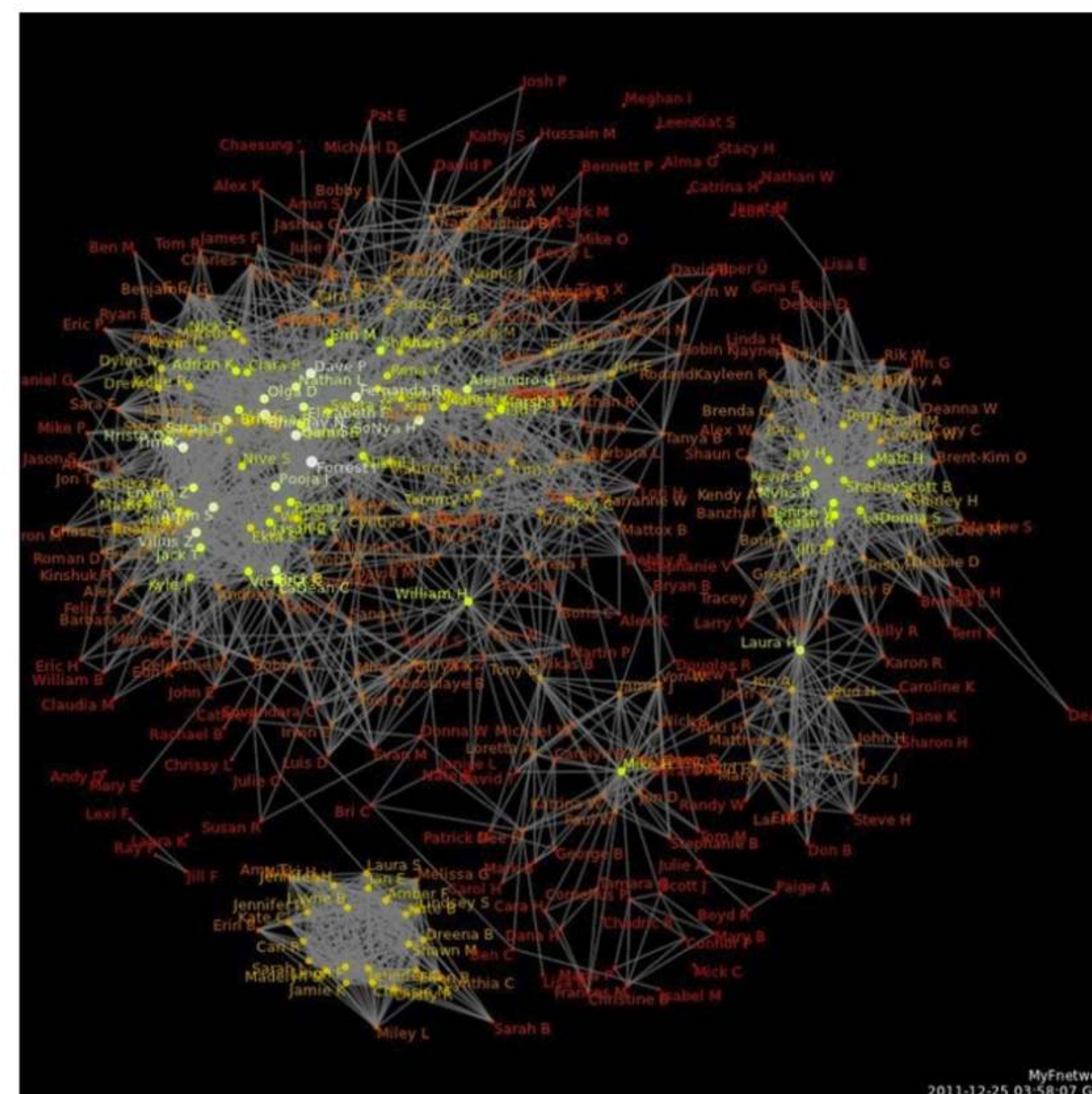
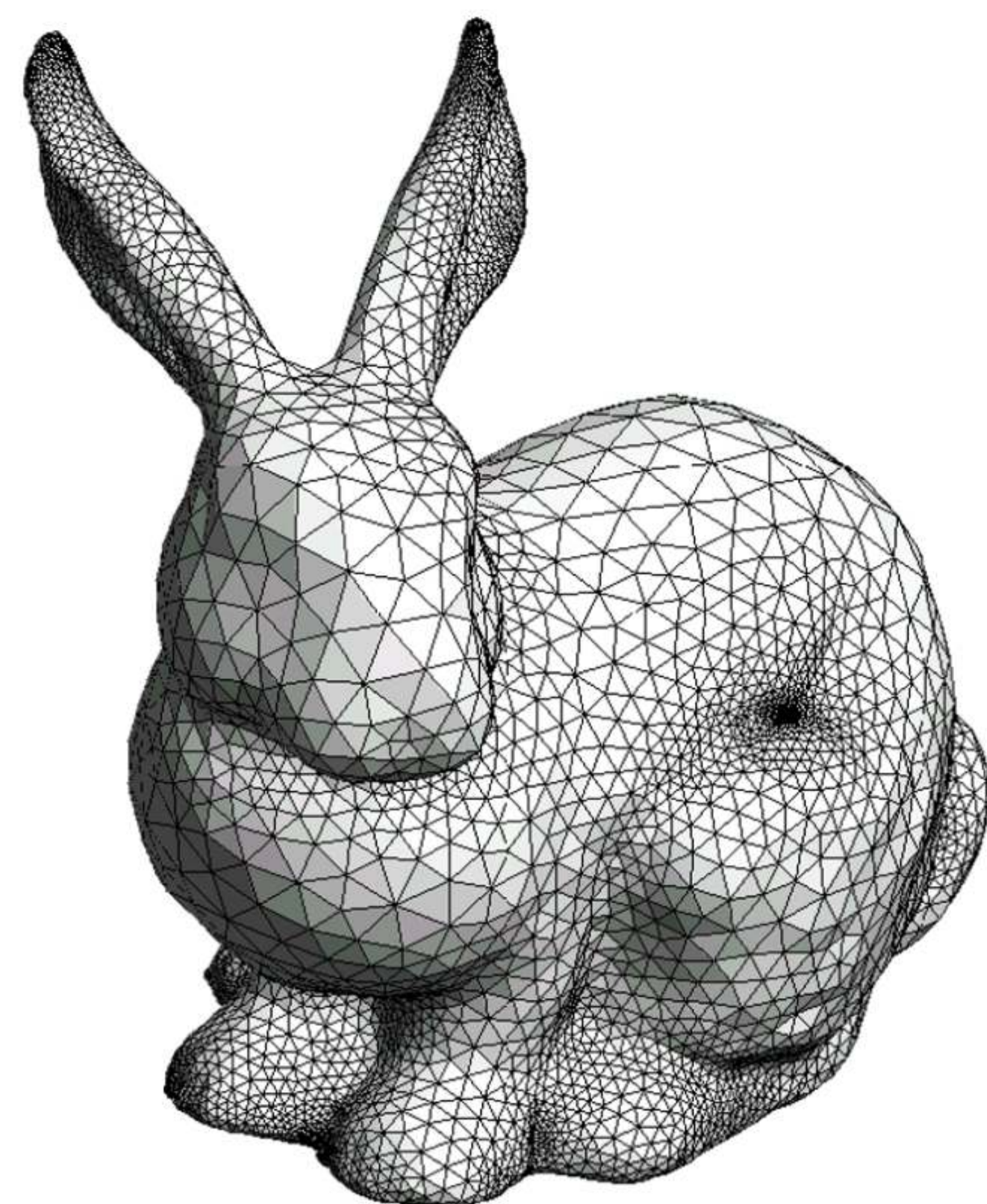
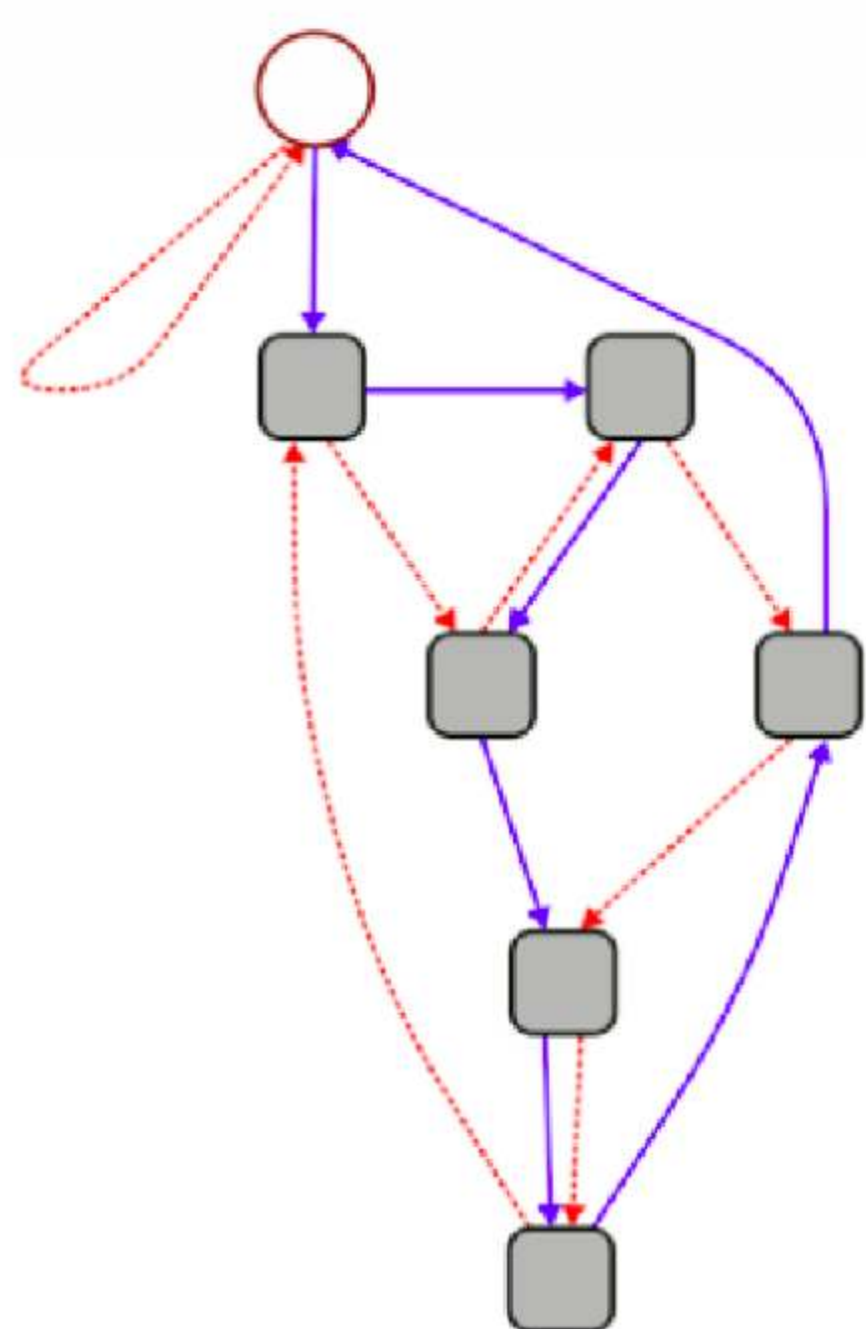
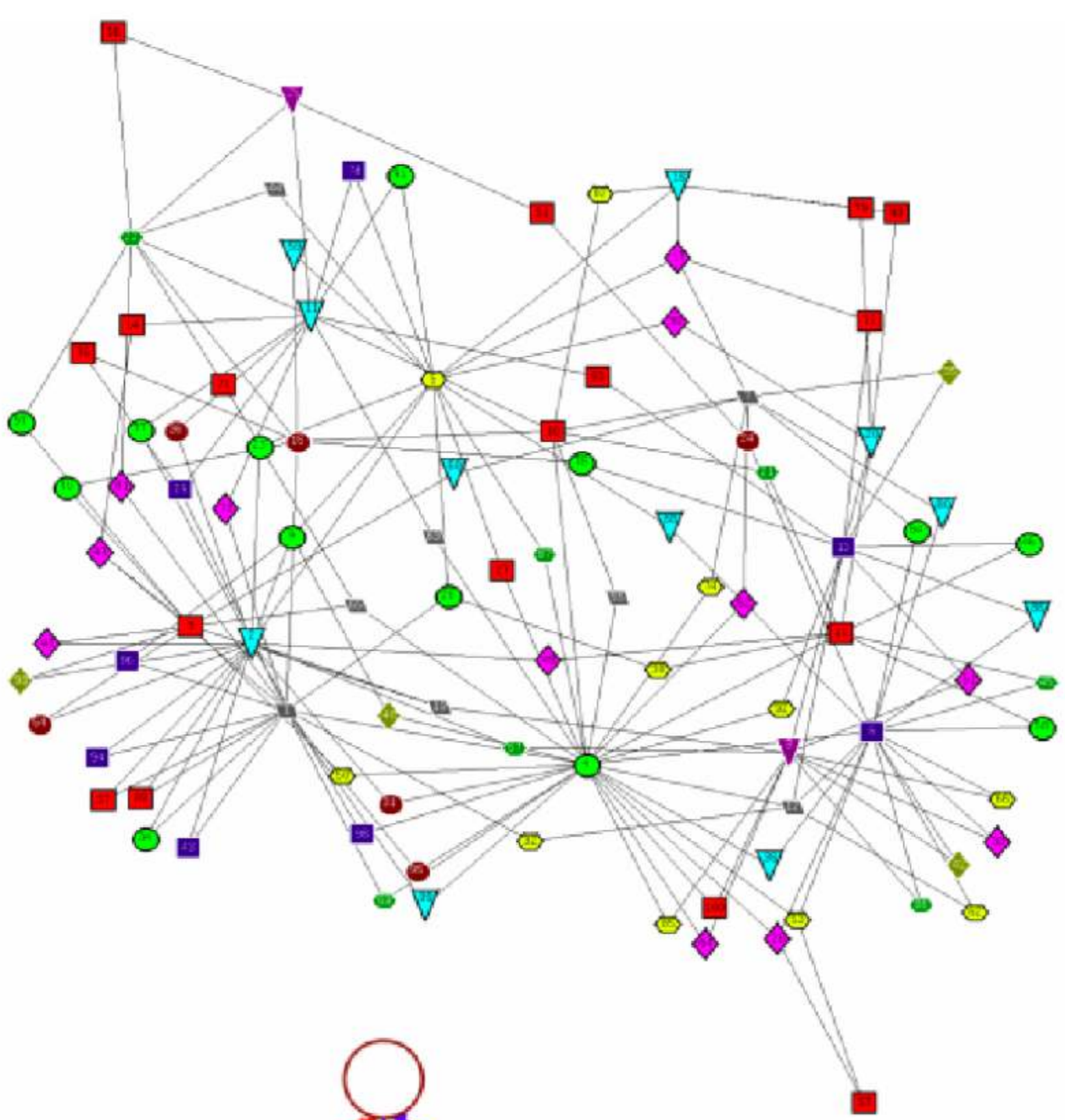
Email to Thierry @ ramais@illinois.edu asap with "conflict" in subject line

- * Please check your current grade at <http://chara.cs.illinois.edu/gb>



How do we get from here to there?
Need:

1. *Common Vocabulary*
2. *Graph implementation*
3. *Traversal*
4. *Algorithms.*

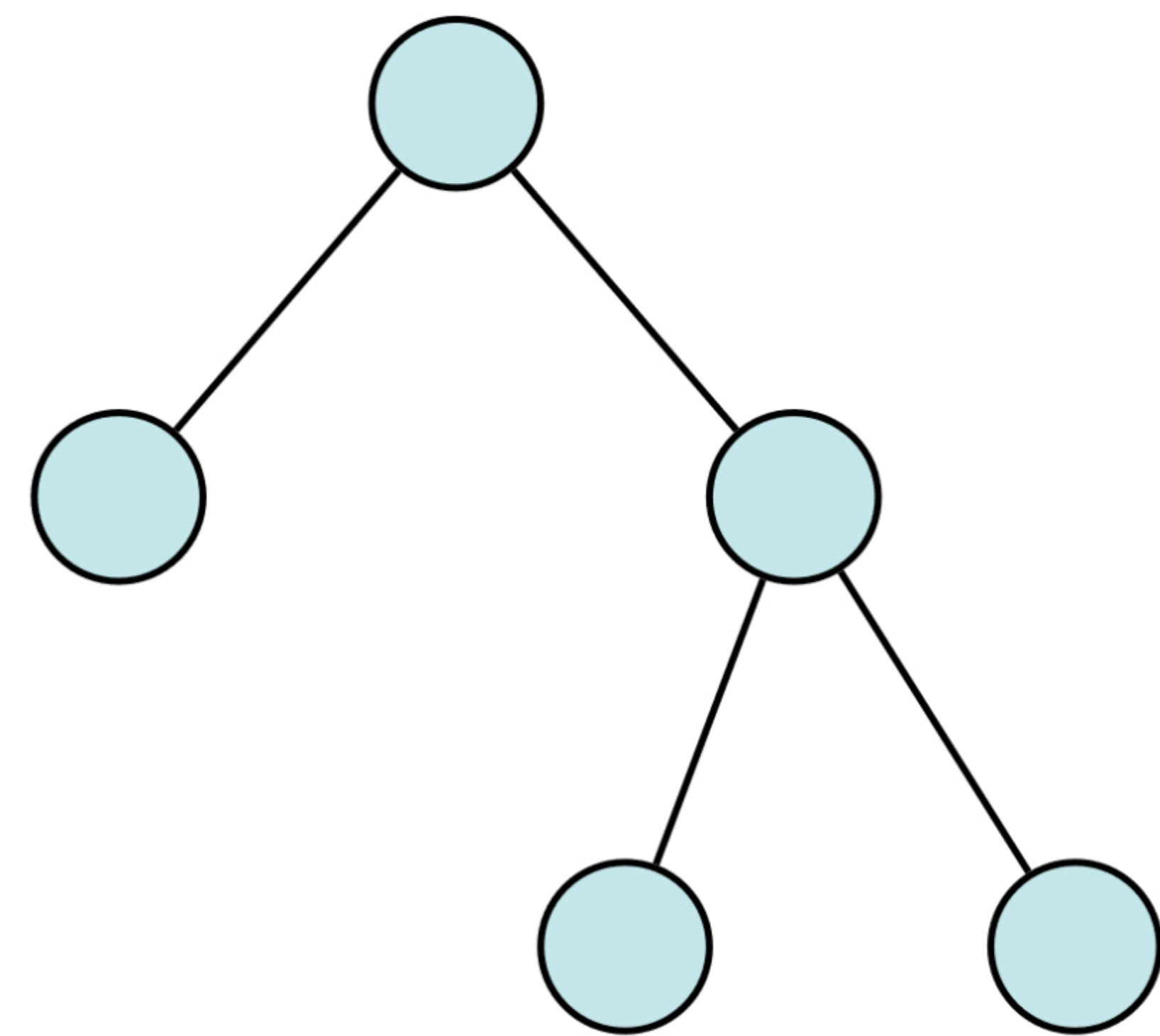


Graphs – traversal

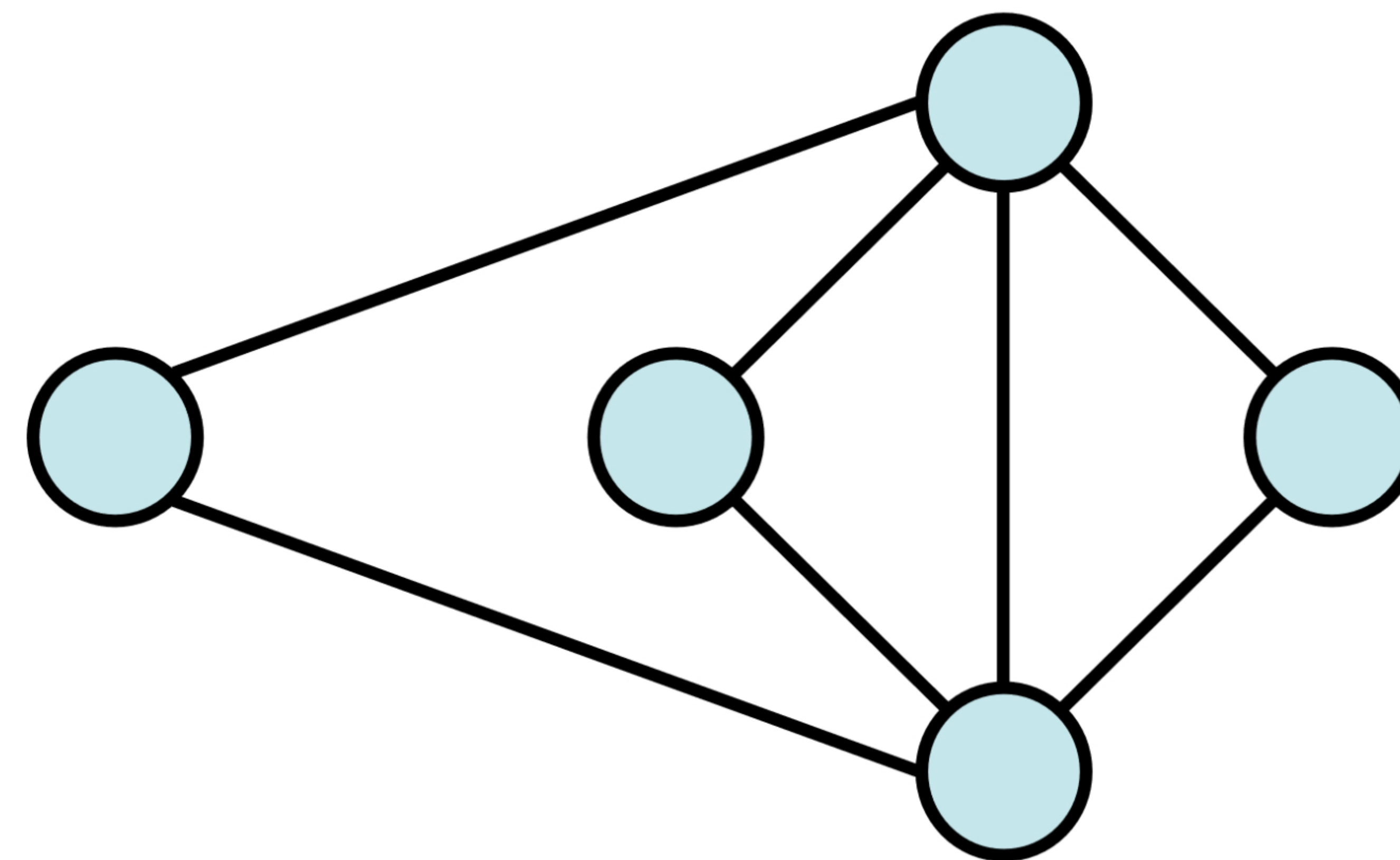
Objective: Visit every vertex and every edge, in the graph,

Purpose: We can search for interesting substructures in the graph,

Contrast graph traversal to BST traversal:



- Ordered
- Obvious start

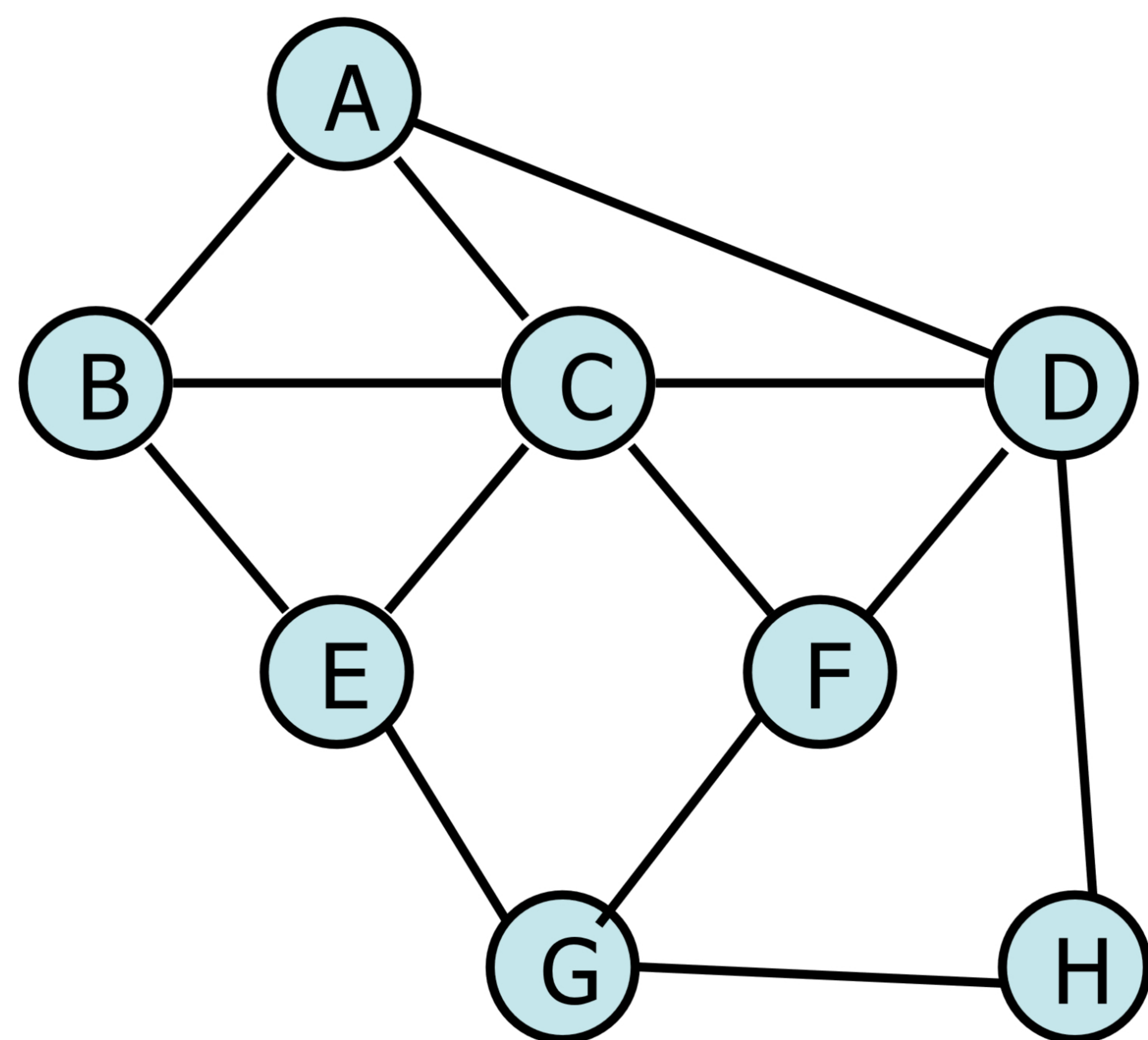


•

•

•

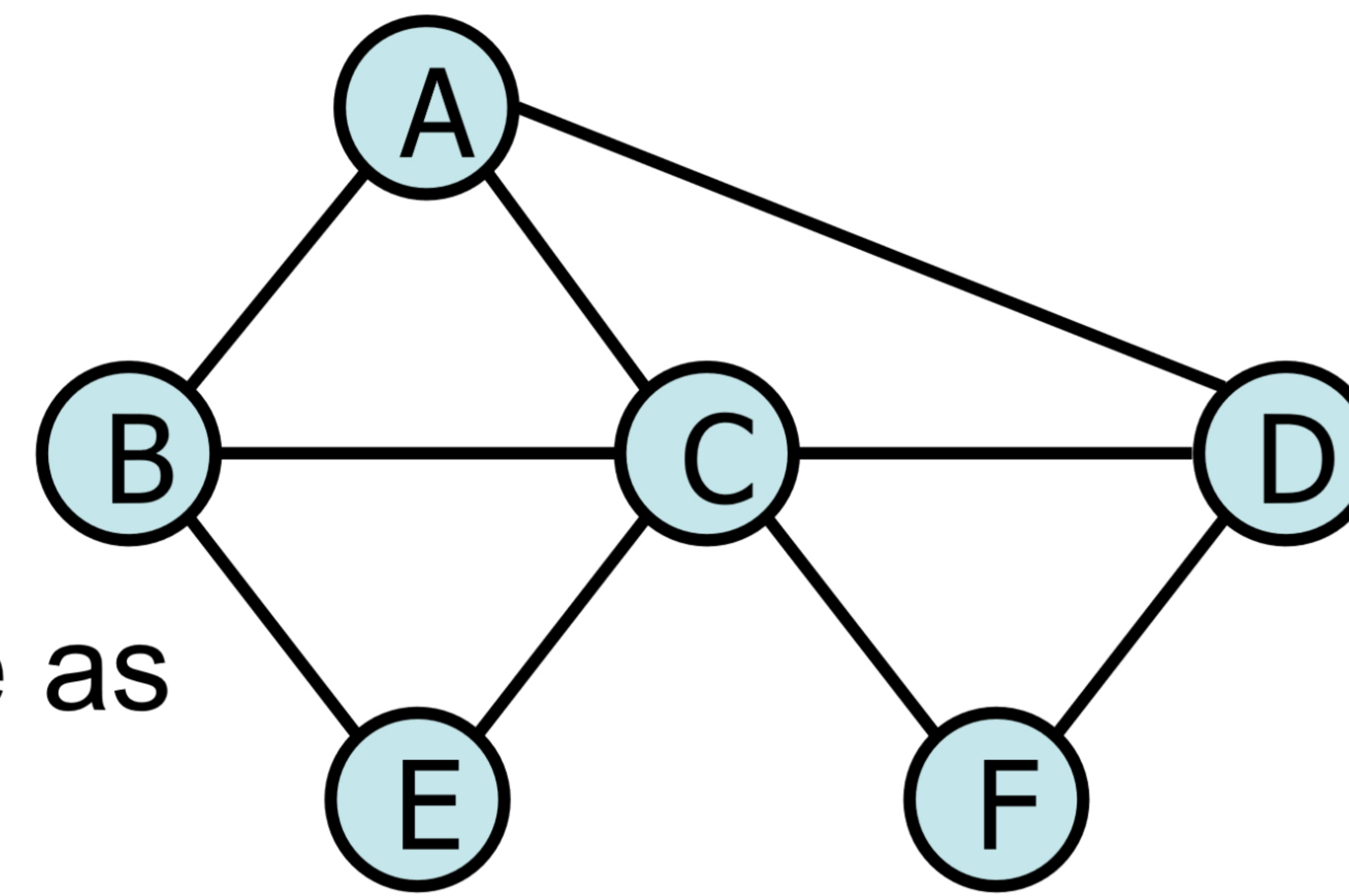
Graphs: BFS example



A	C B D
B	A C E
C	B A D E F
D	A C F H
E	B C G
F	C D G
G	E F H
H	D G

Graphs: Traversal – BFS

Visits every vertex and classifies each edge as either “discovery” or “cross”



Algorithm BFS(G)

Input: graph G

Output: labeling of the edges of G as discovery edges and back edges

For all u in G.vertices()

 setLabel(u, UNEXPLORED)

For all e in G.edges()

 setLabel(e, UNEXPLORED)

For all v in G.vertices()

 if getLabel(v) = UNEXPLORED

 BFS(G,v)

Algorithm BFS(G,v)

Input: graph G and start vertex v

Output: labeling of the edges of G in the connected component of v as discovery edges and cross edges

queue q;

setLabel(v, VISITED)

q.enqueue(v);

While !(q.isEmpty)

 q.dequeue(v)

 For all w in G.adjacentVertices(v)

 if getLabel(w) = UNEXPLORED

 setLabel((v,w),DISCOVERY)

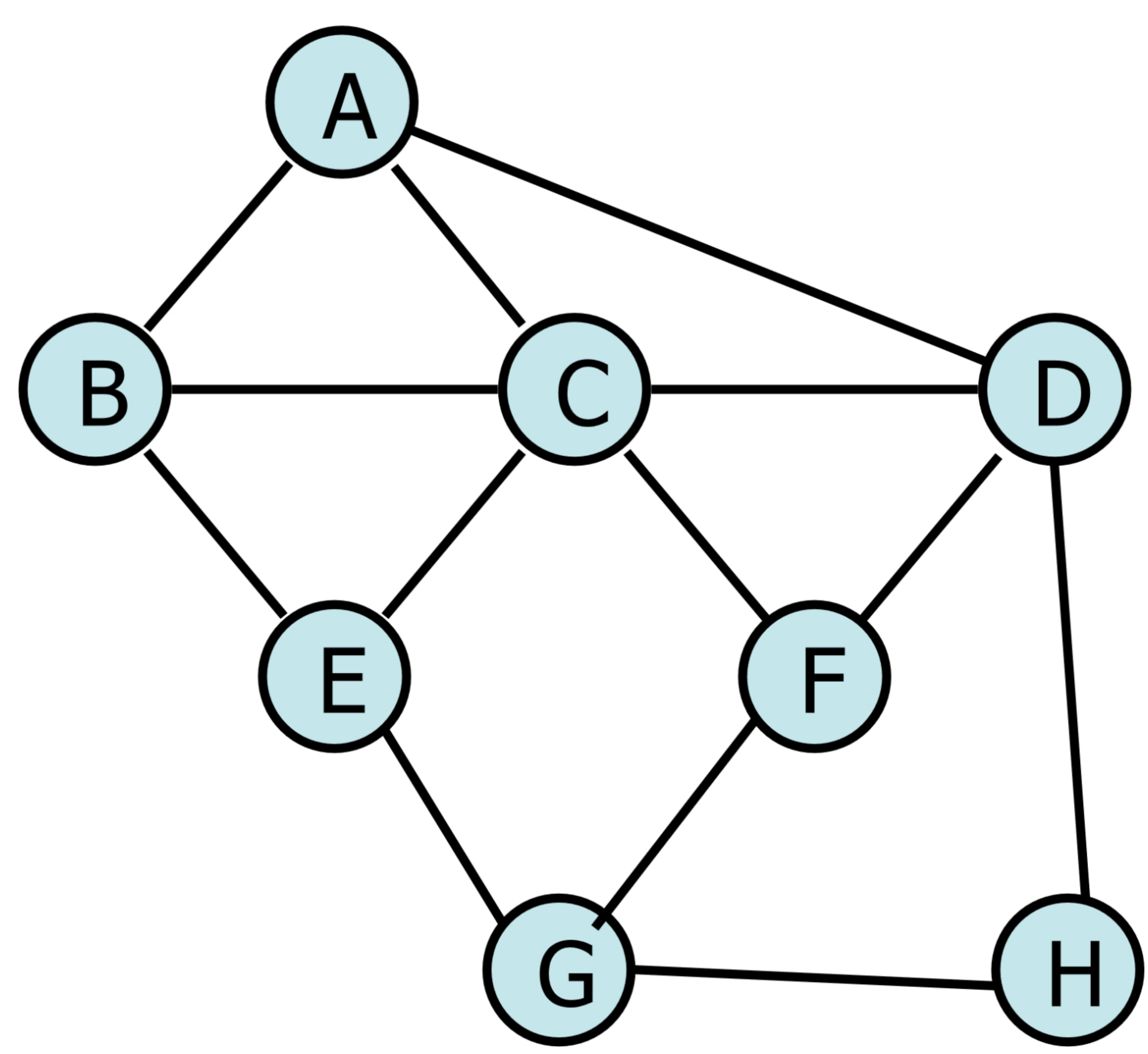
 setLabel(w, VISITED)

 q.enqueue(w)

 else if getLabel((v,w)) = UNEXPLORED

 setLabel((v,w),CROSS)

Graphs: BFS example



A	C B D
B	A C E
C	B A D E F
D	A C F H
E	B C G
F	C D G
G	E F H
H	D G

While loop

For loop

TOTAL RUNNING TIME:

Graphs: BFS observations

