Recitation Problems - Com S 311

Week of Mar 18^{th} - Mar 23^{rd}

1 Recap: Breadth-First Search (BFS)

Input: Starting vertex v

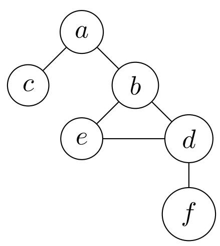
1. Explore all neighbors of v $(v_1, v_2, ..., v_k)$ - these are vertices that can be reached from v in 1 step.
2. Explore the "unexplored" neighbors of $(v_1, v_2, ..., v_k)$ - these are vertices that can reached from v in 2 steps.
3. Continue until we have no new vertices (no "unexplored" vertices remain) to explore.

The starting vertex is in layer 0.
The vertices 1 step away are in layer 1.
The vertices 2 steps away are in layer 2.
The maximum number of layers is $\leq |V| - 1$, where $|V|$ is the number of vertices in the graph.

Algorithm 1 BFS (given starting vertex v and graph $G = (V, E)$)

Initialize $v.explored = false$ for each $v \in V$
Initialize an empty queue Q
Add v to Q (v is the input vertex)
$v.explored = true$
$v.layer = 0$
while Q is not empty **do**
  Remove u from the head of Q
  for each neighbor w of u **do**
    if $w.explored == false$ **then**
      Add w to tail of Q
      $w.explored = true$
      $w.layer = u.layer + 1$

Example:



Adjacency-list

a	$\to b \to c \to /$
b	$\to a \to d \to e \to /$
c	$\to a \to /$
d	$\to b \to e \to f \to /$
e	$\to b \to d \to /$
f	$\to d \to /$

At each vertex, we list its neighbors in lexicographic order in the adjacency list. Initially, $v.explored = false$ for all nodes.

Execution of BFS(a):

- Whenever a node is added to the queue Q, its explored flag is set to true.

$$Q = a \to /$$

  - $a.explored = true$
  - $a.layer = 0$

- Dequeue a from Q. Neighbors of $a = \{b, c\}$, so we add them into the queue Q.

$$Q = b \to c \to /$$

  - $b.explored = true$
  - $b.layer = 1$
  - $c.explored = true$
  - $c.layer = 1$

- Dequeue b from Q. Neighbors of $b = \{a, d, e\}$, but a is already explored, so we only add d and e into Q.

$$Q = c \to d \to e \to /$$

  - $d.explored = true$
  - $d.layer = 2$
  - $e.explored = true$
  - $e.layer = 2$

- Dequeue c from Q. Neighbors of $c = \{a\}$, but a is already explored, so we add nothing to Q.

$$Q = d \to e \to /$$

- Dequeue d from Q. Neighbors of $d = \{b, e, f\}$, but b and e are already explored, so we only add f to Q.

$$Q = e \to f \to /$$

  - $f.explored = true$
  - $f.layer = 3$

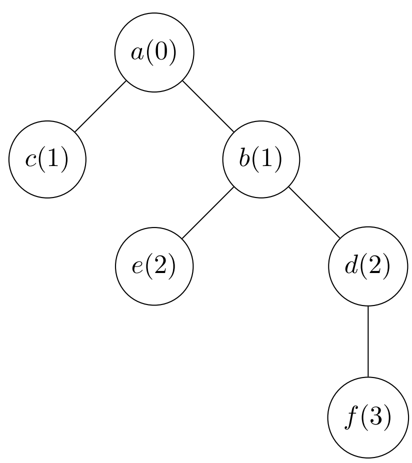
- Dequeue e from Q. Neighbors of $e = \{b, d\}$, but b and d are already explored, so we add nothing to Q.

$$Q = f \to /$$

- Dequeue f from Q. Neighbors of $f = \{d\}$, but d is already explored, so we add nothing to Q.

$$Q = /$$

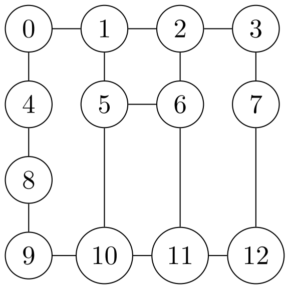
As a result, we have this following breadth first tree:



where each number beside the vertex label corresponds to their layer number.

2 Problems

1. Consider the following undirected graph $G = (V, E)$:



   (a) Specify the following for the graph $G = (V, E)$:

   - $|V| = 13$
   - $|E| = 16$

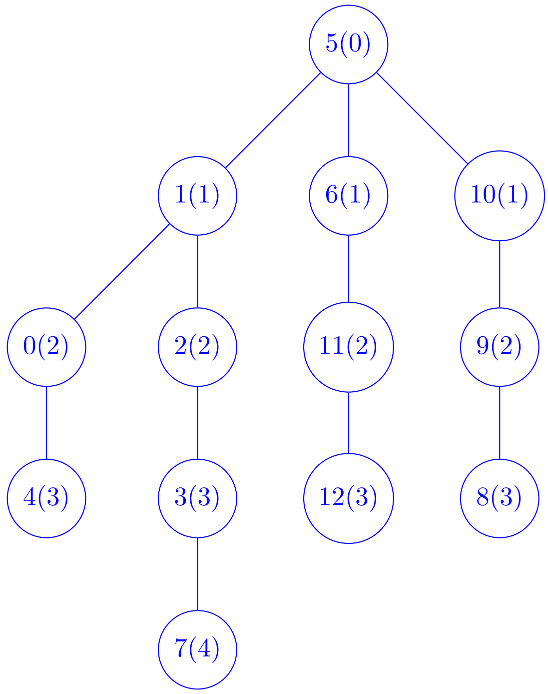
   (b) Write out the adjacency list representation for the graph G. At the list in each node, the vertices connected to it should be in increasing order of vertex number.

	Adjacency-list
0	$\to 1 \to 4 \to /$
1	$\to 0 \to 2 \to 5 \to /$
2	$\to 1 \to 3 \to 6 \to /$
3	$\to 2 \to 7 \to /$
4	$\to 0 \to 8 \to /$
5	$\to 1 \to 6 \to 10 \to /$
6	$\to 2 \to 5 \to 11 \to /$
7	$\to 3 \to 12 \to /$
8	$\to 4 \to 9 \to /$
9	$\to 8 \to 10 \to /$
10	$\to 5 \to 9 \to 11 \to /$
11	$\to 6 \to 10 \to 12 \to /$
12	$\to 7 \to 11 \to /$

   (c) Do a Breadth-First Search starting at vertex 5. That is, draw the BFS tree for $BFS(5)$. For each vertex in the BFS tree, label it with vertex number and layer number. The starting vertex will be:

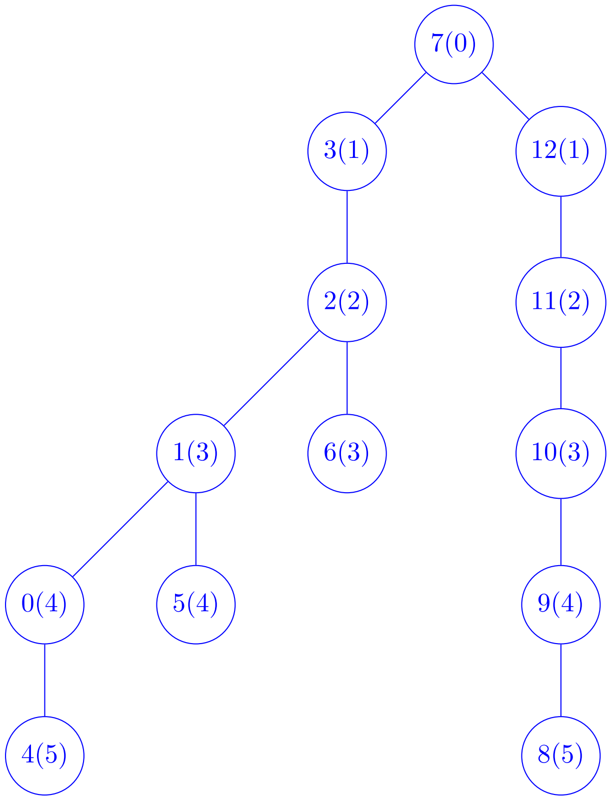
5(0)

   When exploring a vertex, its adjacent vertices should be examined in increasing order of vertex number. What is the highest layer number for your BFS tree $BFS(5)$?



Highest layer number = 4.

(d) Repeat part (c) for $BFS(7)$, that is, draw the BFS tree and the layer numbers for each vertex, starting at vertex 7. What is the highest layer number for the BFS tree $BFS(7)$?



Highest layer number = 5.

(e) Is the tree for $BFS(5)$ the same as the tree for $BFS(7)$?

No, they are different.

(f) In the BFS tree in part (c), that is, $BFS(5)$, is there a layer which contains two vertices that are connected by an edge in G? If so, identify the layer number and the end vertices of this edge.

Cycle: A cycle in a graph is a sequence of vertices, starting at vertex v, and ending at vertex v, with successive vertices in this sequence connected by an edge. $[v, v_1, v_2, ..., v_k, v]$ is a cycle if there is an edge between $< v, v_1 >$, $< v_1, v_2 >$, ..., $< v_k, v >$. The length of this cycle is the number of edges in it; in this example, the length of the cycle is $(k + 1)$.

Does the new edge you found connecting two vertices in a layer help you in finding a cycle in the graph G? If so, give the sequence of vertices in this cycle. Is the cycle you found of odd length or even length? Give a justification for your answer. (**Hint**: Starting at the input vertex of the BFS which is at layer 0, determine the path lengths from the input vertex to each of the end vertices of the edge you found connecting two vertices in a layer)

In layer 3, vertices 4 and 8 are connected by an edge. The cycle is $[5, 1, 0, 4, 8, 9, 10, 5]$ and its length is 7, which is an odd length. Vertex 4 is in layer 3, which has a path length of 3 from the input vertex 5 at layer 0. Same goes for vertex 8, so the length of the cycle is $3 + 3 + 1 = 2 * 3 + 1 = 7$. In general, if there is an edge between two vertices at layer k, the length of the cycle is $2k + 1$, which is an odd number.

(g) Repeat part (f) for the BFS tree in part (d), that is, for $BFS(7)$.

In layer 5, vertices 4 and 8 are connected by an edge. The cycle is $[7, 3, 2, 1, 0, 4, 8, 9, 10, 11, 12, 7]$, which has a length of 11 $(2 * 5 + 1 = 11)$.