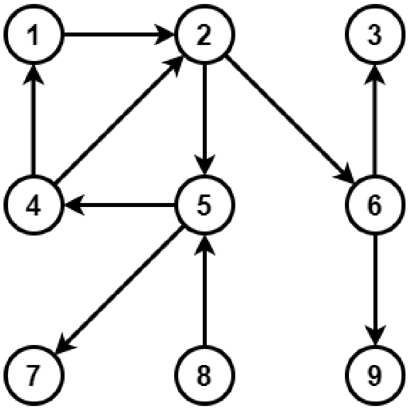
CSE 101
Winter 2022
Quiz 2 **Solutions**

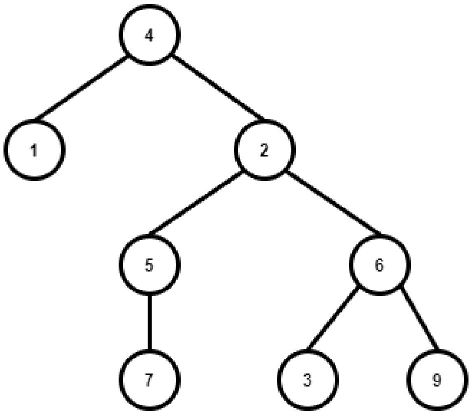
1. (25 Points) Run the BFS algorithm on the digraph pictured below, with vertex **4** as the source. Fill in the table giving the adjacency list representation, colors, distances from the source, and parents in the BFS tree. List the discovered vertices in the order that they enter the queue. Draw the resulting BFS tree.



vertex	*adj*	*color*	*distance*	*parent*
1	2	**black**	**1**	**4**
2	5 6	**black**	**1**	**4**
3		**black**	**3**	**6**
4	1 2	**black**	**0**	**nil**
5	4 7	**black**	**2**	**2**
6	3 9	**black**	**2**	**2**
7		**black**	**3**	**5**
8	5	**white**	**infinity**	**nil**
9		**black**	**3**	**6**

Queue: 4 1 2 5 6 7 3 9

BFS Tree:



(see next page for problem 2)

2. (25 Points) Given a graph G, the *eccentricity* of a vertex $x \in V(G)$ is the maximum possible distance from x to any vertex $y \in V(G)$, i.e.

$$\text{eccentricity}(x) = \max_{y \in (V(G))} \delta(x, y).$$

Using only the Graph ADT functions defined in the project description for pa2, fill in the definition of the client function below that computes and returns the eccentricity of its vertex (`int`) argument.

```
int eccentricity(Graph G, int x){

   int max, y;

   BFS(G, x);
   max = getDist(G, 1);
   for(y=2; y<=getOrder(G); y++){
      if( getDist(G, y)>max )
         max = getDist(G, y);
   }
   return max;

}
```