CSCI 3104 Fall 2021

# Quiz 2 - BFS/DFS

Due Date ..... Sep/17/2021
Name ..... **Michael Ghattas**
Student ID ..... **109200649**

## Contents



## 1 Instructions

- The solutions **should be typed**, using proper mathematical notation. We cannot accept hand-written solutions. Here's a short intro to LaTeX.
- You should submit your work through the **class Canvas page** only. Please submit one PDF file, compiled using this LaTeX template.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please submit this document with no fewer pages than the blank template (or Gradescope has issues with it).
- You **may not collaborate with other students**. **Copying from any source is an Honor Code violation. Furthermore, all submissions must be in your own words and reflect your understanding of the material.** If there is any confusion about this policy, it is your responsibility to clarify before the due date.
- Posting to **any** service including, but not limited to Chegg, Discord, Reddit, StackExchange, etc., for help on an assignment is a violation of the Honor Code.
- You **must** virtually sign the Honor Code (see Section 2). Failure to do so will result in your assignment not being graded.

## 2 Honor Code (Make Sure to Virtually Sign)

**Problem 1.**

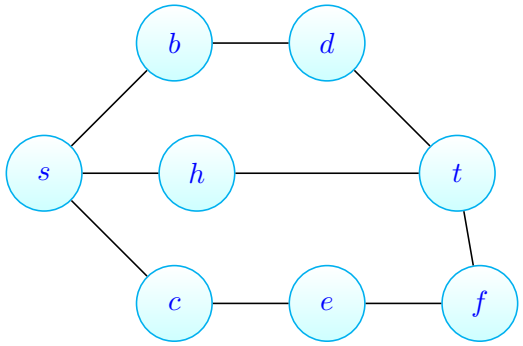
- My submission is in my own words and reflects my understanding of the material.
- Any collaborations and external sources have been clearly cited in this document.
- I have not posted to external services including, but not limited to Chegg, Reddit, StackExchange, etc.
- I have neither copied nor provided others solutions they can copy.

*I agree to the above, Michael Ghattas.* □

# 3 Standard 2- BFS/DFS

## 3.1 Problem 2

**Problem 2.** A *simple* $s \to t$ *path* in a graph $G$ is a path in $G$ starting at $s$, ending at $t$, and never visiting the same vertex twice. Using the graph $G(V, E)$ below and vertices $S, T \in V(G)$, find an $s \to t$ that (i) DFS traverses, and (ii) is neither a shortest path nor a longest simple path. Detail the execution of DFS (list the contents of the stack at each step, and which vertex it pops off the queue). Additionally, show (i) the final $s \to t$ path it finds, (ii) show a shorter $s \to t$ path, and (iii) a longer simple $s \to t$ path.

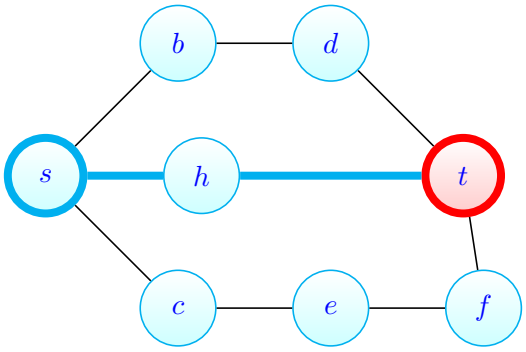



*Answer:*

**Given the edges are not weighted, we will depend on the number of edges traversed from ($s \to t$) to measure each path's length.**

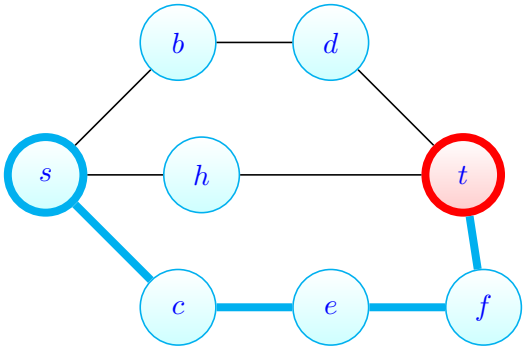
### (a) Shortest Path

$s \to h \to t =$ **[2 Edges]**



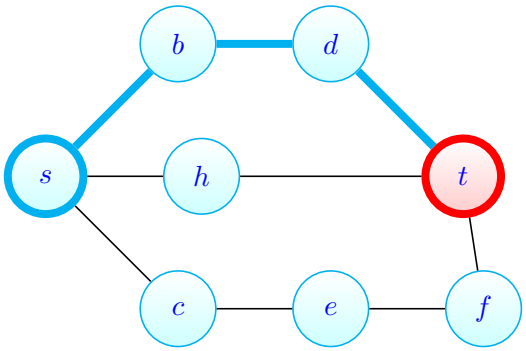

### (b) Longest Simple Path

$s \to c \to e \to f \to t =$ **[4 Edges]**



**(c) DFS** {Different from parts (a) & (b)}

$s \to b \to d \to t$ = **[3 Edges]**




**Priority Queue:**

**[Longest Path = 4 Edges]**

Queue (s), then (c), then (e), then (f), then (t).

(b) $s \to c \to e \to f \to t$

Pop (t), then (f), then (e), then (c).

**[Neither Longest nor Shortest Path (DFS) = 3 Edges]**

Queue (s), then (b), then (d), then (t).

(c) $s \to b \to d \to t$

Pop (t), then (d), then (b).

**[Shortest Path = 2 Edges]**

Queue (s), then (h), then (t).

(a) $s \to h \to t$

Pop (t), then (h), then (s).

**Conclusion:** DFS starts at point (s), it can then move to node (c), (h), or (b). Assuming it moves to node (c) first, it will continue on the path passing through nodes (e) & (f) before it gets to node (t). However, if it moves to node (h) first, it will continue straight to node (t). Furthermore, if it moves to node (b) first, it will continue on the path passing through nodes (d) before it gets to node (t). In a normal situation (i.e. without the current SP & SLP conditions), DFS would be identical to the shortest path in part (a). Though given the specifics of this problem, DFS will ignore the shortest path in part (a) and and choose the path highlighted in part (c). □