**CSCI 3104, Algorithms**                                          **Escobedo & Jahagirdar**
**Homework 5A (40 points)**                                          **Summer 2020, CU-Boulder**

*Advice 1*: For every problem in this class, you must justify your answer: show how you arrived at it and why it is correct. If there are assumptions you need to make along the way, state those clearly.

*Advice 2*: Verbal reasoning is typically insufficient for full credit. Instead, write a logical argument, in the style of a mathematical proof.

**Instructions for submitting your solution**:

- The solutions **should be typed**, we cannot accept hand-written solutions. Here's a short intro to **Latex**.

- In this homework we denote the asymptomatic *Big-O* notation by $\mathcal{O}$ and *Small-O* notation is represented as $o$.

- We recommend using online Latex editor **Overleaf**. Download the **.tex** file from Canvas and upload it on overleaf to edit.

- You should submit your work through **Gradescope** only.

- If you don't have an account on it, sign up for one using your CU email. You should have gotten an email to sign up. If your name based CU email doesn't work, try the identikey@colorado.edu version.

- Gradescope will only accept **.pdf** files (except for code files that should be submitted separately on Canvas if a problem set has them) and **try to fit your work in the box provided**.

- You cannot submit a pdf which has less pages than what we provided you as Gradescope won't allow it.

Name: [            ]

ID: [            ]

**CSCI 3104, Algorithms**          **Escobedo & Jahagirdar**
**Homework 5A (40 points)**          **Summer 2020, CU-Boulder**

# Piazza threads for hints and further discussion

| Piazza Threads |
|:---:|
| Question 1 |
| Question 2 |
| Question 3 |
| Question 4 |

**Recommended reading**:
Graph Algorithms Intro: Ch. 22 $\rightarrow$ 22.1, 22.2, 22.3
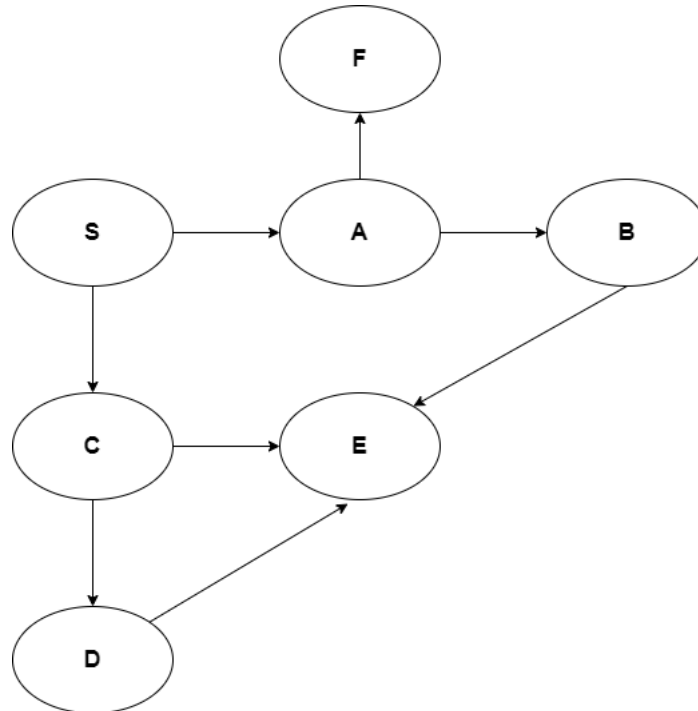Graph Algorithms SSSPs: Ch. 24 $\rightarrow$ 24.3

**CSCI 3104, Algorithms**  **Escobedo & Jahagirdar**
**Homework 5A (40 points)**  **Summer 2020, CU-Boulder**

1. (5 pts) Consider the following unweighted directed graph.



(a) (2.5 pts) Write down the order in which nodes are visited if Depth First Search (DFS) is called on the above graph with **S** as the source node.

(b) (2.5 pts) Write down the order in which nodes are visited if Breadth First Search (BFS) is called on the above graph with **S** as the source node.

**CSCI 3104, Algorithms**
**Homework 5A (40 points)**

2. (5 pts) A simple path $s \to t$ in a graph $G$ is a path starting at $s$, ending at $t$, and never visiting the same vertex twice. Give an example graph (simple, undirected, unweighted) $G = (V; E)$ and vertices $s, t \in V$ such that DFS finds a path from $s$ to $t$ which 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 stack), show the final $s \to t$ path it finds, show a shorter $s \to t$ path, and a longer simple $s \to t$ path.

3. (20 pts) Assume you're given an integer matrix that representing islands in an ocean. A value of 0 in a particular cell indicates water and a value of 1 indicated land. An island is region of land connected vertically, horizontally, or diagonally. The size of an island is the total number of connected land cells. Write an algorithm to compute the sizes of all islands in the matrix. Example:

```
1 0 0 1
0 1 0 1
0 1 0 0
0 1 0 1
```

would output 1, 2, 4.

(a) (6 pts) State if the graph data structure that your algorithm will use for this problem will be i) be weighted or unweighted and ii) directed or undirected. In addition, iii) what makes two nodes have an edge/connection? Give an explanation for each of your answers.

Name:

ID:

**CSCI 3104, Algorithms**        **Escobedo & Jahagirdar**
**Homework 5A (40 points)**      **Summer 2020, CU-Boulder**

(b) (4 pts) Provide a 3-4 sentence description of how your algorithm works, including how the matrix is converted to the graph, how adjacent vertices are identified, and how the algorithm traverses the graph to identify connected vertices.

(c) (10 pts) Write well commented pseudo or real code to solve this problem.
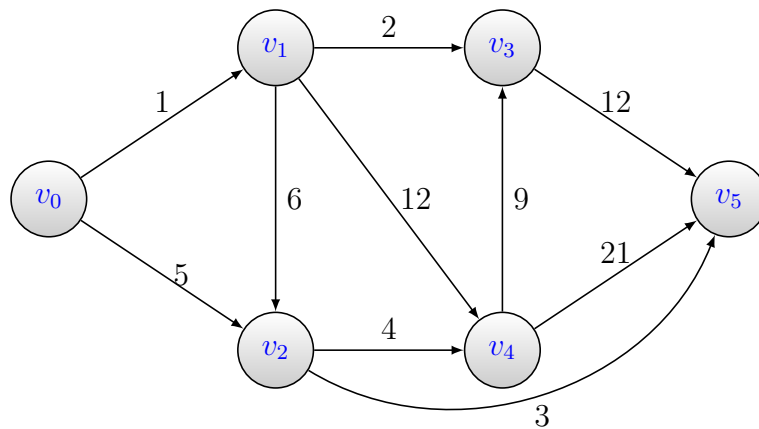
Name:

ID:

CSCI 3104, Algorithms

Escobedo & Jahagirdar

Homework 5A (40 points)

Summer 2020, CU-Boulder

4. (10 pts) Using Dijkstra's algorithm, determine the length of the shortest path from $v_0$ to each of the other vertices in the graph. Clearly specify the distances from $v_0$ to each vertex **after each iteration** of the algorithm.

**CSCI 3104, Algorithms**                                    **Escobedo & Jahagirdar**
**Homework 5A (40 points)**                               **Summer 2020, CU-Boulder**

5. ***Extra Credit (5% of total homework grade)*** *For this extra credit question, please refer the leetcode link provided below or click* here*. Multiple solutions exist to this question ranging from brute force to the most optimal one. Points will be provided based on Time and Space Complexities relative to that of the most optimal solution.*

*Please provide your solution with proper comments which carries points as well.*

https://leetcode.com/problems/keys-and-rooms/

Replace this text with your source code inside of the .tex document