CSCI 3104 Fall 2021
Instructor: Profs. Grochow and Waggoner

Midterm 1- Standard 7

Due Date ..... TODO
Name ..... **Abeal Sileshi**
Student ID **104326364**

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.

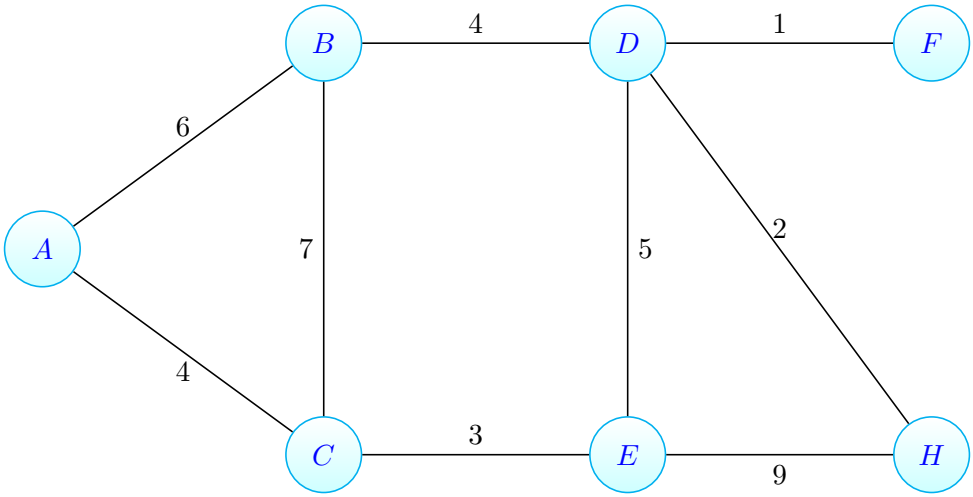
- My submission is in my own words and reflects my understanding of the material.
- I have not collaborated with any other person.
- I have not posted to external services including, but not limited to Chegg, Discord, Reddit, StackExchange, etc.
- I have neither copied nor provided others solutions they can copy.

Agreed (Abeal Sileshi). □

3 Standard 7- Kruskal's Algorithm

3.1 Problem 2

Problem 2. Consider the following graph $G(V, E, w)$. Clearly indicate the order in which Kruskal's algorithm adds the edges to the minimum-weight spanning tree. You may simply list the order of the edges; it is not necessary to exhibit the state of the algorithm (i.e., the disjoint-set data structure) at each iteration.



Answer.

First Kruskal's Algorithm puts edges into a priority queue ordered by weight. Then it polls edges one at a time, adding an edge to the intermediate spanning forest only if it doesn't create a cycle. The initial state of the queue is:
$Q = [(\{D, F\}, 1), (\{D, H\}, 2), (\{C, E\}, 3), (\{A, C\}, 4), (\{B, D\}, 4), (\{D, E\}, 5), (\{A, B\}, 6), (\{B, C\}, 7), (\{E, H\}, 9)]$

1) It polls edge {D,F} from Q, as {D,F} doesn't create a cycle, it's added to our spanning forest.
2) It polls edge {D, H} from Q, as {D, H} doesn't create a cycle, it's added to our spanning forest.
3) It polls edge {C, E} from Q, as {C, E} doesn't create a cycle, it's added to our spanning forest.
4) It polls edge {A, C} from Q, as {A, C} doesn't create a cycle, it's added to our spanning forest.
5) It polls edge {B, D} from Q, as {B, D} doesn't create a cycle, it's added to our spanning forest.
6) It polls edge {D, E} from Q, as {D, E} doesn't create a cycle, it's added to our spanning forest.
7) It polls edge {A, B} from Q, as {A, B} creates a cycle in our spanning forest, we do not add {A, B} to our spanning forest
8) It polls edge {B, C} from Q, as {B, C} creates a cycle in our spanning forest, we do not add {B, C} to our spanning forest
9) It polls edge {E, H} from Q, as {E, H} creates a cycle in our spanning forest, we do not add {E, H} to our spanning forest

Kruskal's Algorithms added the edges in order: {D,F}, {D, H}, {C, E}, {A, C}, {B, D}, {D, E} □