# MATH-UA-240 / UY-4314 — Combinatorics

**Professor Charles Stine**

**Name:** **Homework Assignment #5**
**NetID:** **Due Date:** October 7, 2024, 11:59 PM

- This homework should be submitted via Gradescope by 11:59 PM on the date listed above.
- There are two main ways you might want to write up your work.
  - Write a PDF using a tablet
  - Write your answers on paper, clearly numbering each question and part.
    - You can use an app such as OfficeLens to take pictures of your work with your phone and convert them into a single pdf file. Gradescope will only allow pdf files to be uploaded.
- You must show all work. You may receive zero or reduced points for insufficient work. Your work must be neatly organized and written. You may receive zero or reduced points for incoherent work.
- Please start a fresh page for each numbered problem. You may have parts a), b) and c) on the page for example, but problems 1) and 2) should be on separate pages.
- When uploading to Gradescope, **you must match each question to the page that your answer appears on**. If you do not, we will be unable to grade the unmatched problems.
- When appropriate, please put a box or circle around your final answer.
- The problems on this assignment will be graded on correctness and completeness.

# Lecture 9

1. (10 points) Dijkstra's Algorithm: use the graph below for parts (b) and (c).

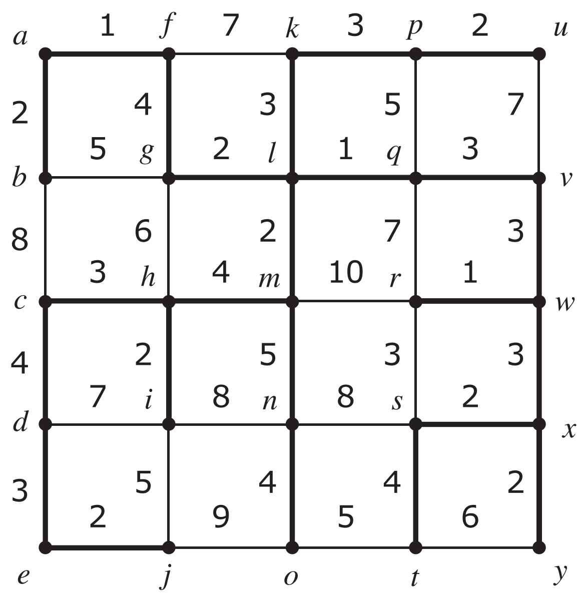



Figure 1: A Network to test the shortest path algorithm.

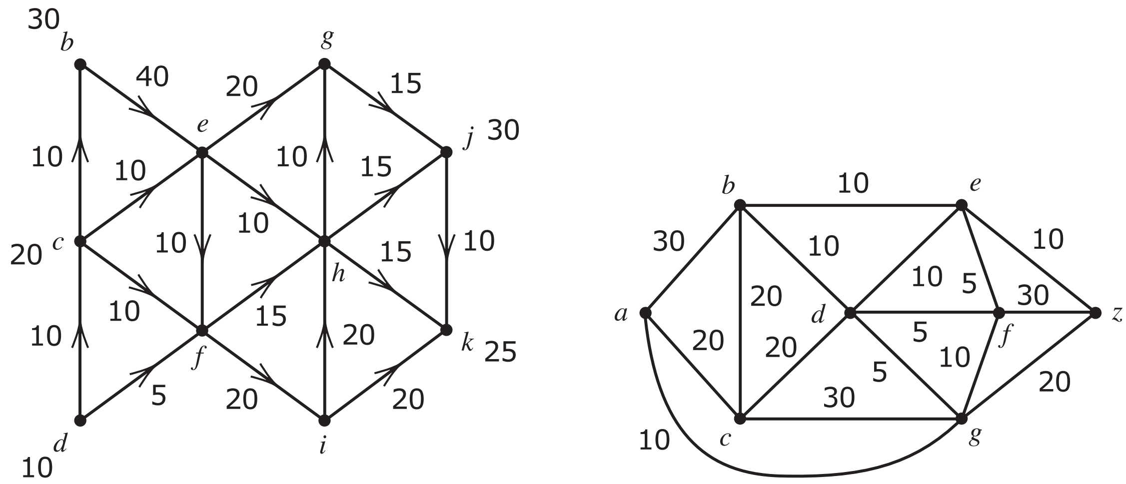
   (a) Prove by induction on $m$ that the shortest path algorithm actually does find the shortest path from a given vertex to the chosen vertex $A$.
   (b) In the graph above, apply the algorithm to find the shortest path from $a$ to $y$.
   (c) Find the shortest path from $d$ to $r$.

2. (10 points) Understanding the Merge Sorting algorithm. Read Example 1 on page 122-123 of our textbook *Applied Combinatorics, 6th edition* by Alan Tucker, which explains how to do the merge-sort algorithm. Based on your understanding of that example, answer the following questions.

   (a) Apply the merge sorting algorithm to the list:

$$L = (15, 27, 4, -7, 9, 13, 8, 28, 12, 20, -80)$$

   (b) Assume two sub-lists have lengths which add up to $n$. Show that it takes at most $(n-1)$ comparisons to correctly merge the two lists by describing a procedure which uses at most that many.
   (c) Obtain a lower bound on the height of the merge-sort tree for any list $L$ with length $n$. Combine this bound with your answer to $B$ to show that the merge-sort algorithm completes in approximately $n \log_2(n)$ time.




Networks for Problem 3 (left) and Problem 4 (right).

# Lecture 10

3. (10 points) Network Flows with simple Supply and Demand. Consider the network shown above (left), in which vertices $b,\ c,\ d$ have supplies 30, 20, 10 respectively, while vertices $j,\ k$ have demands of 30 and 25.

   (a) Find a flow satisfying these demands, if such a flow exists.

   (b) Reverse the direction of the edge $(h, g)$ and repeat part (a).

4. (10 points) A-Z Flows and Planar Graphs. Suppose a Network is a planar graph and $a$ (the left-most vertex) and $z$ (the right-most vertex) are both on the unbounded region surrounding the network. Construct the *Dual Network* as follows: first add edges (rays) connecting vertex $a$ to infinity to the left and vertex $z$ to infinity to the right. Give each ray infinite capacity (weight). Now form the dual graph by placing a vertex in the center of every face and connecting the centers of faces which share an edge by a new edge which passes through the midpoint of the old edge. Assign a weight to each dual edge that is equal to the weight of the edge it crosses.

   (a) Show that a shortest path from the center of the upper unbounded face in the dual network to the lower unbounded face corresponds to a minimal $a$-$z$ cut in the original network. This will require you to show that paths in the dual correspond to cuts in the original, AND that this correspondence sends shorter paths to lower-capacity cuts.

   (b) Draw the dual network for the network shown above (right), and find the shortest path, using the algorithm from Problem 1, from the vertex corresponding to the upper unbounded face to that of the lower unbounded face. Give a minimal $a$-$z$ cut for $N$.