

Homework Assignment 5

Graphs, Trees, and the Pigeonhole Principle

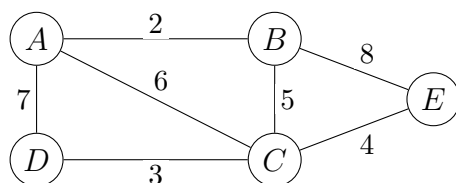
Due: Monday, October 26 at 11:59 p.m.

Instructions. Submit your answers as a PDF on Webcourses. I encourage you to work in groups and discuss the problems with one another. You may also use AI tools, but remember that AI will not be available during exams. Use these problems to check your own understanding of the material. Your grade is based on participation, not correctness. You must submit your answers to receive credit for this assignment.

Justify every answer. When using the pigeonhole principle, identify the objects and the boxes. You may use results proved in lecture, except a result that the question asks you to prove. Unless stated otherwise, graphs are finite, simple, and undirected.

Question 1 Minimum spanning trees when costs change

The edge labels in this connected graph are weights.



- Find an MST. Give its edges and total weight, with a brief explanation of why it is minimum.
- Suppose every edge weight increases by 10. Must your tree still be an MST? Give its new total weight. Prove that adding the same constant to every edge weight preserves the set of MSTs in *any* connected weighted graph.
- Return to the original weights, then replace only the weight of AD by a positive real number x . For which values of x is your tree from part (a) still an MST? At the boundary value, is it the only MST? Justify your answer without listing every spanning tree.

Hint. Compare the cheapest tree that uses AD with the cheapest tree that does not.

Question 2 Huffman coding and average length

A source produces six symbols with the following probabilities:

Symbol	A	B	C	D	E	F
Probability	0.32	0.24	0.18	0.12	0.09	0.05

At each Huffman merge, put the smaller-weight tree on the left. Label left edges 0 and right edges 1. No ties occur. The decoder already knows the codebook, so ignore the cost of transmitting it.

If symbol s has probability $p(s)$ and codeword length $\ell(s)$, its contribution to the expected length is $p(s)\ell(s)$. Thus

$$L = \sum_s p(s)\ell(s)$$

is the expected number of bits per symbol.

- Construct the Huffman tree. Show the weight created at each merge, and give every symbol's codeword and codeword length.
- Compute L . What is the expected number of bits used to encode 1000 symbols, each with the distribution above? Compare this with a fixed-length binary code using the fewest possible bits per symbol.
- Keep the tree from part (a), but exchange the symbols A and F at its leaves. Compute the new expected length. More generally, suppose $p(s) > p(t)$ but $\ell(s) > \ell(t)$. Show that exchanging their codewords strictly decreases expected length. Explain why this exchange preserves the prefix-free property.
- A student says, "Since Huffman has a smaller expected length than the fixed-length code, it uses fewer bits for *every* message." Is that true for this example? Give a proof or a specific counterexample, and explain what the expected-length comparison actually guarantees.

Question 3 What happens when a tree loses a vertex?

A **connected component** is one whole connected piece of a graph: any two vertices in that piece are joined by a path, and no vertex in the piece is joined by a path to a vertex outside it. An isolated vertex forms a component by itself.

Example. The graph below has edges joining 1 to 2, 2 to 3, and 4 to 5. It has three connected components:

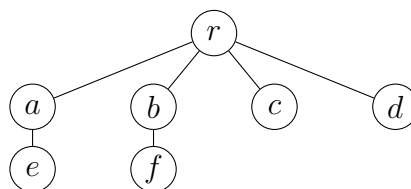


Their vertex sets are

$$\{1, 2, 3\}, \quad \{4, 5\}, \quad \{6\}.$$

Vertices 1 and 3 belong to the same component even though they are not adjacent: the path $1 - 2 - 3$ connects them.

Deleting a vertex also deletes every edge incident to it. Consider this tree:



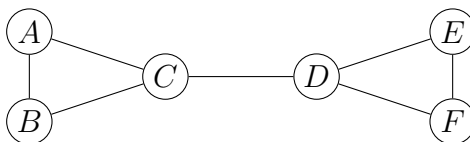
- Treat each deletion independently. How many connected components remain after deleting r , after deleting a , and after deleting e ? List the vertices in each component.
- Let T be any tree with at least two vertices, and let v have degree d . Prove that deleting v leaves exactly d connected components.

Hint. Show that different neighbors of v end up in different components, and that every remaining vertex is connected to one of these neighbors.

- Does the conclusion in part (b) hold for every connected simple graph? Give a proof or a counterexample. Explain which property of a tree your proof in part (b) needed.

Question 4 Which roads must be retraced?

Each edge below represents a road of length 1. A route may revisit vertices; its length counts every edge traversal, including repeats. An *Eulerian tour* is a closed route that uses each edge exactly once.



- Find the degree of each vertex. Does this graph have an Eulerian tour? Explain.
- Now allow repeated edges. Find a shortest closed route starting at A that uses every edge at least once. Give its length and explain why no shorter route works.
Hint. How can a route move between the two triangles?
- A *bridge* is an edge whose deletion disconnects a connected graph. Prove that a connected graph containing a bridge cannot have an Eulerian tour.
- Is having no bridges enough to guarantee that a connected simple graph has an Eulerian tour? Give a proof or a counterexample.

Question 5 A consecutive block with divisible sum

We want a *consecutive* block, so we may not rearrange entries or skip entries inside the block. For example, in $4, -2, 5, 1$, the block $-2, 5$ is allowed, but the selection $4, 5$ is not.

- For the list

$$4, -2, 5, 1, -3, 6, 2,$$

compute the remainders modulo 7 of its partial sums, including the empty partial sum $S_0 = 0$. Use two equal remainders to find a nonempty consecutive block whose sum is divisible by 7.

- Let $n \geq 2$, and let $a_1, \dots, a_n$ be any n integers. Prove that some nonempty consecutive block has sum divisible by n . Your proof must allow negative entries, repeated entries, and zeros.

Hint. Set $S_0 = 0$ and $S_j = a_1 + \dots + a_j$ for $1 \leq j \leq n$. What does $S_j - S_i$ represent when $i < j$?

- Show that n entries are necessary for a guarantee: for every $n \geq 2$, give a list of $n - 1$ integers with no nonempty consecutive block whose sum is divisible by n .