CS420 Midterm

Department of Computer Science

Spring 2026

Instructions Your answers must be your own work, collaboration with students or AI is not allowed.

Time allowed: 50 minutes

You do not need to answer the questions in order, you may prefer to read all the questions and attempt easier ones first.

Show all reasoning and provide sufficient detail for the steps of calculations for full credit

You may use standard facts proved in lecture (e.g. linearity of expectation, Jensen's inequality)

1. Let $G = (V, E)$ be a graph, and consider the greedy matching algorithm that starts with an empty matching $M = \varnothing$ and processes edges in an order $e_1, \ldots, e_m$, adding e_i to M if it is disjoint from all edges already added to M.
   (a) Prove that there exists an ordering of E for which this algorithm outputs a maximum matching. [4 points]
   (b) Find a graph on 4 vertices and an ordering of its edges such that a maximum matching has size 2 but the greedy algorithm with this ordering finds a matching of size 1. [4 points]
   (c) Boost your answer to the previous question to show that for any $n \geq 4$ which is divisible by 4 there is a graph on n vertices with an ordering of its edges such that the greedy algorithm with this ordering finds a matching of size $n/4$, while the maximum matching has size $n/2$. [4 points]

[Total: 12 points]

2. Let $\Sigma = \{a, b, c, d\}$ be an alphabet with probabilities

$$p(a) = \frac{1}{2}, \quad p(b) = \frac{1}{4}, \quad p(c) = \frac{1}{8}, \quad p(d) = \frac{1}{8}.$$

   (a) Construct the Huffman code tree for this alphabet by showing the merging steps. Give the resulting binary codewords for each symbol. [8 points]

   (b) Compute the expected codeword length of your code [2 points]

   (c) Compare the expected codeword length with the Shannon source coding theorem lower bound. What can you conclude about your code? [2 points]

[Total: 12 points]

3. You have n electronic wire components $w_1, \ldots, w_n$ in a production batch. Exactly one is defective due to a manufacturing error. You can perform non-destructive tests on subsets of components: given any subset S of wires you connect them in a circuit and see if electricity does not flow. The test returns YES if the defect is present in the tested subset, NO otherwise.

   (a) Prove that you need at least $\lceil \log_2 n \rceil$ tests in the worst case. [2 points]

   (b) Describe a testing strategy that achieves this bound, and explain in 2-3 sentences why this is optimal in the information-theoretic sense. [4 points]

   Hints: Think both in terms of the entropy of the unknown information in the problem, and in terms of the entropy of the test that you do. Since each test is binary, how can you design tests that maximize the entropy of the test outcome?

[Total: 6 points]

4. Consider a randomized greedy matching algorithm. Given a graph $G = (V, E)$, choose the edge ordering uniformly at random and run the same greedy matching algorithm as in Question 1 with this ordering.

   (a) Define indicator variables X_e for the event "e is put into the matching M." Argue that

   $$\mathbb{E}[|M|] = \sum_{e \in E} \Pr(e \in M).$$

   [2 points]

   (b) Let $e = uv \in E$ and define its edge-degree $d_E(e) = d(u) + d(v) - 2$, which counts edges sharing an endpoint with e (but not e itself). Show that

   $$\Pr(e \in M) \geq \frac{1}{d_E(e) + 1}$$

   and use this to give a lower bound on $\mathbb{E}[|M|]$. [6 points]

   (c) Compute the lower bound from part (b) for the 8-vertex cycle graph C_8, and the true maximum matching size in C_8. [4 points]

[Total: 12 points]

5. Suppose that we are given an array A of length n containing distinct positive integers. Recall the quickselect algorithm and consider using it to find the k-th largest element of A. Let the elements of A be $a_1 < a_2 < \cdots < a_n$ so that a_i is the i-th largest element of A (not necessarily the i-th element of A). Then the goal is to find the target a_k.

   Two elements a_i and a_j (where $i < j$) are compared if and only if one of them is chosen as a pivot *before they are separated*. In quickselect, a_i and a_j can be separated in two ways:

   - A pivot p is chosen strictly between them (i.e. $a_i < p < a_j$).
   - A pivot p is chosen such that the target a_k falls into one sub-array, while a_i and a_j fall into the other sub-array.

   (a) Comment on how the separation of elements differs between quicksort and quickselect. [2 points]

   (b) Let X_{ij} indicate whether a_i is compared to a_j in quickselect. Explain why the total number of comparisons X satisfies

   $$X = \sum_{1 \le i < j \le n} X_{ij} \qquad \text{and} \qquad \mathbb{E}X = \sum_{1 \le i < j \le n} \mathbb{E}X_{ij}.$$

   [2 points]

   (c) Assume that $\mathbb{E}X_{ij} = \frac{2}{\max(j,k) - \min(i,k) + 1}$, and derive a bound $\mathbb{E}X \le O(n)$ using the expressions above. [8 points]

   Hints: The min/max functions mean that is is convenient to split sum over $1 \le i < j \le n$ into three regions: where $i < j \le k$, where $k \le i < j$, and where $i < k < j$. One approach that works is to show that each region contributes at most $2n$ to the sum separately:

   - $\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{2}{\max(j,k) - \min(i,k) + 1} \le 2n$,
   - $\sum_{i=k}^{n-1} \sum_{j=i+1}^{n} \frac{2}{\max(j,k) - \min(i,k) + 1} \le 2n$,
   - $\sum_{i=1}^{k-1} \sum_{j=k+1}^{k} \frac{2}{\max(j,k) - \min(i,k) + 1} \le 2n$.

   The final case is the hardest, but is easier if you sum over the distance $d = j - i$ from 2 to n and observe that there are at most $d - 1$ pairs (i, j) which satisfy $j - i = d$ and $i < k < j$.

   [Total: 12 points]

[TURN OVER]

[END]