This is a diagnostic exam intended to help you evaluate your readiness for the real exam.

The following rules apply to you, whether you think they do or not. Read and understand them; failure to abide by these rules, or directions given by course staff during the exam, may result in disciplinary action, including but not limited to a failing grade in the class.

- This exam is solely for students enrolled in this lecture. Anyone not enrolled in this lecture may not take an exam.
- You may not open the exam or begin writing until the instructor has explicitly given you permission to do so.
- Keep your UCI ID readily accessible during the test. Proctors may request to see it.
- This exam is closed book, closed notes, and is individual effort. Once course staff begin passing out exams, you may not communicate with anyone other than proctors for any reason, nor may you have electronics, including calculators watches and phones, available to you during the test for any reason. **YOU DO NOT NEED A CALCULATOR!**
- If you leave your seat during the test for any reason, your instructor may collect it and deem you to have turned it in. Do not ask proctors for an exemption to this, they are not authorized to grant such.
- You must take the exam in your assigned seat unless the professor (not a TA) tells you otherwise. The instructor will call to cease writing at 11:48 AM, at which point you must immediately cease writing and close the exam. You may not write any further at that point, including finishing one's current sentence.
- If you believe a question is ambiguous, write at least two reasonable interpretations and indicate clearly which one you will be using. Then answer your question with that assumption. Unless your interpretation makes the problem much more trivial than intended, we will grade your response as if one of us had made that clarification.
- The purpose of the real exam is to evaluate how well you understand the material presented in the course. It is an academic integrity violation to do anything that subverts the goals of this assessment including, but not limited to, not doing your own work or submitting that of anyone else.
- We will only grade responses marked in the space provided for each question.
Nothing you write on this page will be graded.  The next page in this booklet contains a spot to answer these questions. You may use this page as scratch paper if you would like, and room to do so exists.

1. (2 points) Suppose I am trying to compute a Minimum Spanning Tree using Kruskal’s algorithm and the Union-Find Data Structure.

   As a reminder, Kruskal’s Algorithm considers the edges in non-descending order of weight. For each edge, if including it would create a cycle, the algorithm rejects the edge. Otherwise it accepts the edge.

   For this problem, assume that the Union-Find data structure is using Union-by-Rank (which has the same meaning it did in lecture) and, if ranks are tied, the tiebreaker is arbitrary. Assume also that the structure is not using path compression.

   I recommend you read the full problem before starting to solve it. It is possible that there are multiple correct answers to this problem; you need only provide one of them.

   ![Graph Image]

   (a) How many edges are accepted prior to the first time we reject an edge? Write only an integer answer.

   (b) Which edge is the first to be rejected?

   (c) At the time the first edge is rejected, what does find(d) return?

2. (2 points) Use Huffman’s algorithm to produce an encoding tree for the phrase ATHELSTAN’S STANS STEAL AT LEAST TEN TENTS. Note that there will be nine leaf nodes, and the period at the end of the sentence is not part of the phrase.

   The count of characters is:

<table>
<thead>
<tr>
<th>A</th>
<th>T</th>
<th>H</th>
<th>E</th>
<th>L</th>
<th>S</th>
<th>N</th>
<th>'</th>
<th>–</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>9</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>7</td>
<td>4</td>
<td>1</td>
<td>6</td>
</tr>
</tbody>
</table>

© CompSci 161 Winter 2023 – Michael Shindler – University of California, Irvine
This exam may not be reposted without the express written permission of the professor teaching this course.
Write your answers to question one in the appropriate boxes below.

<table>
<thead>
<tr>
<th>1a:</th>
<th>1b:</th>
</tr>
</thead>
</table>

Draw your tree for question two below.
3. (3 points) Suppose we have a set of \( n \) homework assignments, each of which have the same deadline. However, they do not all take the same amount of time to do: assignment \( i \) takes \( t_i > 0 \) minutes to do. For each assignment turned in prior to the deadline, we get zero penalty. For each assignment turned in after the deadline, we lose one point for each minute it is late. For example, if two homework assignments are late, one by one minute and the other by three minutes, we lose four points.

Our goal is to minimize the number of points we lose. Prove that doing our homework assignments in order of shortest task first (i.e., non-decreasing value of \( t_i \); we do the shortest task, then the second shortest, and so on) will optimize this objective.

Remember to write your answer on the answer page →, not here, although you may use this space for scratch work.
Write your proof for question three on this page. It is possible that you won’t use the entire page.