**SIXTH QUIZ**

You have 15 minutes from the start of class to complete this quiz. Give partial answers if you can’t give complete ones. Read the questions with care; work with deliberate speed. Don’t give us more than we ask for. The usual instructions apply. Good luck!

**Problem 1** (10 points)

Below is a relevance tree for pizzas. Fill in the six blanks to calculate the ratings of each alternative. Showing your work might increase your chances of getting partial credit if your arithmetic is wrong.

```
<table>
<thead>
<tr>
<th>Pizza</th>
<th>Toppings</th>
<th>Flavor</th>
<th>Variety</th>
<th>Cost</th>
<th>Delivery time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fratello’s</td>
<td>4</td>
<td>8</td>
<td>10</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Sorella’s</td>
<td>6</td>
<td>4</td>
<td>8</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>
```

(a) (8 points) Fill in the six blanks to calculate the ratings of each alternative. Showing your work might increase your chances of getting partial credit if your arithmetic is wrong. Finally, according to this relevance tree, which pizza should you choose?

(b) (1 point) Which of the two pizzas costs less?

(c) (1 point) Which of the two pizzas has the faster delivery time?
Problem 2 (4 points)
You run a customer support center where operators receive calls from customers. The customer requests fall into three categories: returning purchased merchandise (“returns”), technical support (“support”), and general information (“info”). Of all the calls, 25% are returns (which take an average of 4 minutes each), 50% are info (which take 2 minutes each), and 25% are support (which take 12 minutes each). What’s the expected length of a customer call (i.e., the expected value of the length of a call, counting all categories)?

\[ 4 \times 0.25 + 2 \times 0.5 + 12 \times 0.25 = 1 + 1 + 3 = 5 \text{ minutes} \]

Problem 3 (8 points)
In class we discussed the “software crisis.”

(a) (2 points) In just a few words, what is the “software crisis”?

(b) (2 points) We covered three different approaches to “solving” the crisis; list two of them.

(c) Pick one of the two approaches you listed above, and write its name here:

(c.1) (2 points) In one brief English sentence, describe the approach you just listed.

(c.2) (2 points) In one brief English sentence, give a reason why this approach did not (completely) solve the software crisis (other than “It was too time-consuming and expensive”).

Problem 4 (3 points)
If “easy to implement” means that you can use only the existing conventional methods a data structure provides, is each of the following true or false?

(a) Using a dequeue (double-ended queue), it’s easy to implement a stack.

(b) Using a queue (first-in, first-out), it’s easy to implement a stack.

(c) Using a stack, it’s easy to implement a priority queue.
<table>
<thead>
<tr>
<th>Brief definition or description</th>
<th>One reason why it didn’t “solve” the software crisis (other than “It’s too expensive and time-consuming”).</th>
<th>One situation where this approach is appropriate or useful today, or one useful concept or technique that this approach produced.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structured (“gotoless”) programing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>N-version programing (“design diversity”)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Formal verification (program proofs)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>