The Exam will cover:
1. Rosson and Carroll Chapters 1, 2, and 5. Please read the chapters and the instructor’s lecture notes.
4. All materials covered in lectures, quizzes, and projects.

When taking the exam, please...
1. Define all acronyms at their first occurrence in your answer.
   For example, given the question: “What is HCI?”
   Good: “Human-computer interaction (HCI) is a field of study that examines how people…”
   Bad: “HCI is how humans interact…”
2. Do not use a term in its definition.
   Good: “Interaction design is the process of evaluating alternatives for the dynamic aspects…”
   Bad: “Interaction design is the process of designing how a person will interact…”
3. When defining terms, be sure to say what something is before providing additional details.
   Good: “HCI is a field of research and practice that examines how people…”
   Bad: “HCI is how humans interact…”

A general suggestion for studying: You should be able to take any two random ideas, concepts, or terms from the materials, and explain how they relate. Midterm questions will likely require you to do this.

Scenario-Based Usability Engineering - Rosson & Carroll Chapter 1
The page numbers refer to the book. Many of these topics are also discussed in the instructor’s lecture notes.
1. What is scenario-based usability engineering?
2. What is a scenario? How are scenarios used in user interface design? How do scenarios contribute to the design of easy-to-use, easy-to-learn, and useful human-computer interfaces?
3. What is a prototype? How are prototypes used in the design and development of user interfaces?
4. What is iterative development? Why is iterative development a critical component of interface design?
5. How is the expression “there is no silver bullet” used in the context of computer science?
6. Usually, in a design process, there is one correct answer, and the goal of the design process is to find that one answer. Do you agree or disagree? (Hint: Read the “Tradeoffs in Design” box on p.7.)
7. Why is it useful to have a “design rationale” at the end of a design process?
8. What is usability? How can it be measured quantitatively?
9. What was Card, Moran, and Newell’s noted contribution to human-computer interaction? Why was this an important contribution?
10. What is GOMS? (p.12 and p.235-238, but the “select insertion point” method in Figure 7.2 is wrong)
11. What is cognitive science? How can user interface design benefit from cognitive science?
12. What is participatory design?

Analyzing Requirements - Rosson & Carroll Chapter 2
1. What is requirements analysis?
2. Work can be characterized as having three dimensions: activities, artifacts, and social context. Describe each of these. Why is it important to consider each in the design of a user interface?
3. What is hierarchical task analysis (HTA)? (pp. 39-40 and Annett, 2003)
4. Which of the three dimensions of work are best captured by HTA? Which are not so well captured by HTA?
5. What is ethnography?
6. What are stakeholders? Why is it important to get all stakeholders involved in the requirements analysis phase of a system design?
7. What is “tacit” knowledge? How can you get it from someone?
8. What is a field study? What is the goal of a field study? What kinds of data should you collect in a field study? What kinds of analyses would be useful to conduct?
9. When interviewing a user or potential user of a system, why is it sometimes useful to start the interview with open-ended questions? What is an example of an open-ended question? An example of a question that is not open-ended?
10. What are hypothetical stakeholders? (Note: They are also known as personas.) Why is the role of hypothetical stakeholders in a design process?
11. In scenario-based design, a “claims analysis” is a consideration of design tradeoffs, a sort of cost-benefit analysis. Why is a consideration of design tradeoffs important in a design activity? (pp. 72-75)

Interaction Design - Rosson & Carroll Chapter 5
1. What is interaction design? Why is it difficult to represent interaction design? What are three different ways to represent an interaction design?
2. What are the seven stages of action? (Figure 5.1 on p. 160)
3. What is the “gulf of evaluation” and the “gulf of execution”? How can an interface reduce these “gulfs”?
4. What is an affordance? What is the best way to figure out if something that you think is an affordance is really an affordance?
5. What is a direct manipulation interface? What is a command language interface? What are two advantages of direct manipulation over command-based interaction? Of command-based interaction over direct manipulation? Give an example of a task for which it would be best to provide a direct manipulation interface and a task for which it would be best to provide a command based interaction. (pp. 161-163)
6. What is an action sequence? (pp. 164-171) How does a user put together an action sequence? What is the relationship between planning an action sequence and HTA?
7. What is a chunk? (p. 166; and slides)
8. What are keyboard shortcuts? How are they useful? How are they not so useful? (p. 177)
9. What is a storyboard? (p. 190) What aspect of a user interface does a storyboard help to convey?
10. Name and briefly describe the two basic types of user errors. Which are more common for novices? Which are more common for experts? Give a specific example of each, and provide a general design guideline for minimizing each type of error.

Fitts’ Law
1. What is Fitts’ Law?
2. What does it predict? For what task?
3. What is the equation?
4. How are the parameters set?
5. How is it useful?
6. What are some specific design implications of Fitts’ law?

1. What is HTA?
2. What is a task?
3. HTA addresses what shortcoming in traditional time and motion studies?
4. What is the role of the “plan” in the HTA? Where does it appear in a diagram? What are the three common types of plans discussed in the reading?
5. To what level of detail should you take an HTA?
6. What is the value in carrying out an HTA as opposed to simply listing the sequence of steps that a person needs to take to accomplish a task, such as in a flowchart?
7. How is HTA useful for building easy-to-use, easy-to-learn, and useful computer systems?