**Name:** **ID Number:**

# CS 6456 — Principles of UI Software

**March 2nd 2000**

## Midterm Exam

1. (10 points) Most systems that use input events provide a hierarchy of events that contain progressively more detailed information about the event. The event at the root of such an event hierarchy contains information that is provided to all events.

a) Name at least 4 pieces of information would you expect to see in this root event?

b) What are two problems with this uniform event model?

2. (15 points) There are two dominant metaphors for computer interfaces. Describe them and highlight the differences between the two. Can we use finite state machines or grammer based approaches for these metaphors? Why or why not?

3. (20 points) In class, we discussed event handling in the context of a multi-threaded, object-oriented toolkit like Swing.

a) Setch what the event loop would look like (i.e., write it out in pseudo-code) in such a toolkit.

b) What are some reasons that we might we want to use multiple threads to handle event dispatch?

c) In class, we discussed why a single thread is typically used to handle all event dispatch. Discuss why multiple threads are generally not used to handle incoming events.

d) Why did you put the redraw where you did in the above psuedocode?

4. (20 points) Window systems provide a *virtual device* abstraction. What are some motivations for providing such an abstraction? Discuss for both the input and output side of the window system.

5. (20 points) When we talk about the output side of a toolkit, we break the process into two separate phases, “geometry management” and “redraw”. What is the difference between these two phases, and why are they normally separated?

6. (15 points) Constraints.

a) What is the different between one-way and multi-way constraints?

b) State one advantage and one disadvantage of each.

c) Which is more commonly used in user-interface toolkits? Why?