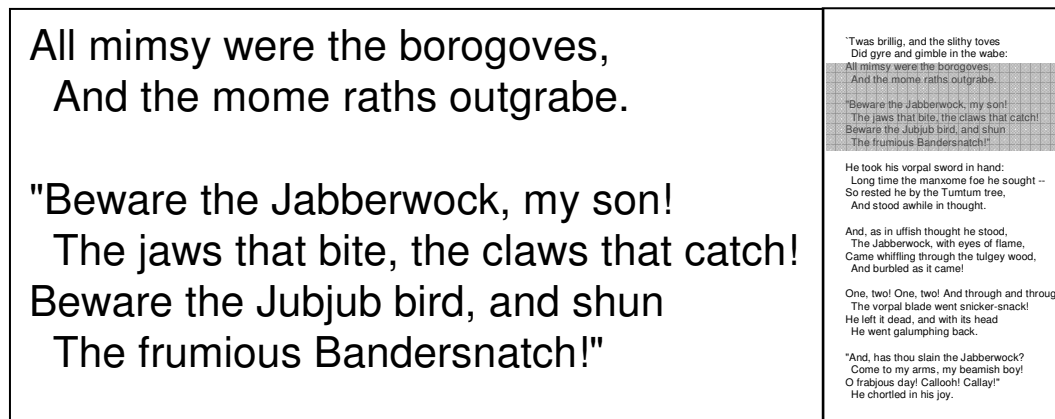


Quiz 2

This quiz is closed book, closed notes. You have 80 minutes to complete it. Each question has enough space for a complete, concise answer, but an additional blank page is included at the end if you find it necessary to continue your answer.

Your name: _____

Suppose you're implementing a new kind of scrolling pane widget. Instead of displaying a conventional scrollbar for moving around its child component, the widget should display a miniature view of the entire child, with a **25% gray** rectangle showing which part of the child is currently visible. The example below shows how the scrolling widget might look with a text widget as its child. The input to the miniature view is like a scrollbar, so (for example) clicking and dragging the gray rectangle should behave the same as clicking and dragging a scrollbar thumb.



All the parts of the widget should be implemented as components, named as follows:

- **MiniScrollPane:** the entire widget itself
- **Viewport:** the part that shows the child component at normal size, on the left
- **Track:** the miniature view of the child component, on the right
- **Thumb:** the gray rectangle
- **Child:** the child component being scrolled (which happens to be a text editor in the example above, but might be another component).

3. (8 points) Fill in the paintComponent methods for **Track** and **Thumb** below. (In Swing, paintComponent() paints just the component itself; any children of the component are automatically painted next.) Some useful Swing methods are shown at the bottom of the page.

```
// references to other components
JComponent miniScrollPane, viewport, child, track, thumb;
```

```
class Track extends JComponent {
    public void paintComponent(Graphics g) {
        Graphics2D g2 = (Graphics2D) g;
```

```
        // Your code goes here
```

```
    }
}
```

```
class Thumb extends JComponent {
    public void paintComponent(Graphics g) {
        Graphics2D g2 = (Graphics2D) g;
```

```
        // Your code goes here
```

```
    }
}
```

```
// Useful Swing methods
g2.translate(x, y)
g2.rotate(angle)
g2.scale(x,y)
g.setColor(new Color(r, g, b, a))
    where r,g,b,a are doubles in the range [0,1]
g.setComposite(rule)
    where rule can be SRC_OVER_DST, DST_OVER_SRC,
                    SRC_IN_DST, DST_IN_SRC,
                    SRC_OUT_DST, DST_OUT_SRC
g.fillRect(x,y,width,height)
```

Form Title -- (appears above URL in most browsers and is used by WWW search)		Background Color:	...
Q&D Software Development Order Desk		FFFBF0	
Form Heading -- (appears at top of Web page in bold type)		Text Color:	...
Q&D Software Development Order Desk <input checked="" type="checkbox"/> Center		000080	
E-Mail responses to (will not appear on)	Alternate (for mailto forms only)	Background Graphic	...
dversch@q-d.com			
Text to appear in Submit button	Text to appear in Reset button	<input type="radio"/> Mailto	
Send Order	Clear Form	<input checked="" type="radio"/> CGI	
Scrolling Status Bar Message (max length = 200 characters)			
WebMania 1.5b with Image Map Wizard is here!			
<input type="button" value="Prev Tab"/>		<input type="button" value="Next Tab"/>	

4. (6 points) List at least 3 different violations of **graphic design rules** in the user interface above.

5. (6 points) Do a heuristic evaluation of the user interface above, finding at least 3 different problems (which must be different from problems listed in the previous question).

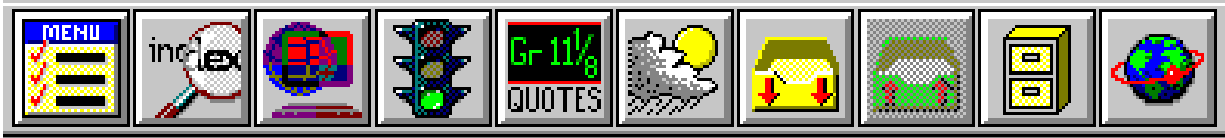
6. (6 points) Suppose you are developing a stock trading application, in which delaying a few seconds can result in money lost. Thinking about the most important aspect of usability for this problem, discuss the applicability of each evaluation method listed below. Mention at least one limitation of each method that's relevant to the application you're developing.

(a) Heuristic evaluation

(b) Formative evaluation

(c) Predictive evaluation

7. (6 points) Suppose you're redesigning the appearance of an important part of your interface, so that the important part passes the squint test. Which visual variables will help, and which ones won't?



8. (6 points) Discuss the usability of the color choices in the toolbar above.

9. (6 points) Discuss the internationalization of the toolbar above.

10. (6 points) Suppose you have the MIT logo (shown below on the left) loaded into an Image object. The red and gray pixels are opaque, and the white background is transparent. You want this logo in all blue letters instead, but still with a transparent background (as shown on the right). Fill in the code below to make it happen. Some useful methods are shown at the bottom of the page.



Original logo



You want this

```
// Variables you can use
Image logo; // the MIT logo
int w = logo.getWidth();
int h = logo.getHeight();

// Create a new Image for the result.
// The blue logo should be in the result image at the end.
Image result = new BufferedImage(w, h);
Graphics2D g = (Graphics2D) result.getGraphics();

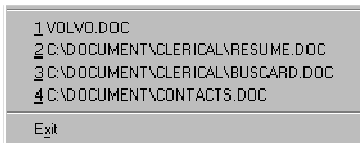
// Your code follows

}

// Useful Swing methods
g.drawImage(image, x, y)
g.setColor(new Color(r, g, b, a))
    where r,g,b,a are doubles in the range [0,1]
g.setComposite(rule)
    where rule can be SRC_OVER_DST, DST_OVER_SRC,
                    SRC_IN_DST, DST_IN_SRC,
                    SRC_OUT_DST, DST_OUT_SRC
g.fillRect(x,y,width,height)
```

11. (6 points) Make keystroke-level models of two methods for opening a file in a typical word processor. Show your work by annotating each step with the specific action the user takes. You don't have to estimate the actual time for each method. Assume the user has been **typing** prior to this method.

(a) Using **only the mouse** to choose volvo.doc from a Recent Files list at the bottom of the File menu (like the one below).



(b) Using **only the keyboard** to invoke the Open File dialog box using a shortcut (Ctrl-O or Command-O) and then typing volvo.doc explicitly.

12. (8 points) Here is part of a Java program intended to animate a rectangle:

```
// instance variables
int x, y, w, h; // current position & size of rectangle

// paint method
public void paint(Graphics g) {
    g.setColor(Color.RED);
    g.drawRect(x, y, w, h);
}

// animation method
public void animate(int newX, int newY) {
    int oldX = x, oldY = y;
    for (int i = 1; i <= 100; ++i) {
        Thread.sleep(10); // sleep for 10 msec
        x = (int) (oldX*(i/100.0) + newX*(1 - i/100.0));
        y = (int) (oldY*(i/100.0) + newY*(1 - i/100.0));
        paint(getGraphics());
    }
}
```

Suppose (x,y,w,h) is currently $(0, 0, 10, 10)$, and that the window is 200 pixels square. How will the program behave when it calls `animate(100, 100)`? Consider both **input** and **output** behavior, and draw pictures of the start, middle, and end of the animation to illustrate the output part of your answer.

13. (6 points) List three different things you should say to a user before a formative evaluation.
14. (6 points) Louis Reasoner has 10 boolean preferences to display on a Preferences dialog box (preferences like Open Application Maximized and Save Automatically On Exit). He says, "Let's use a standard multiple-selection listbox containing all 10 choices." Give two reasons why his suggestion is a bad idea, and propose a better design.
15. (6 points) Consider this Javascript code that generates HTML:

```
div.innerHTML = "Good " + (hourOfDay < 12 ? "morning" : (hourOfDay < 18 ?  
"afternoon" : "evening")) + ", " + username + "!";
```

Aside from the fact that the string literals are English, what assumptions does this code make that might need internationalization?

16. (6 points) In a certain toolkit, undoable edits implement an interface that has 3 methods:

```
void doEdit();
```

```
void undo();
```

```
void redo();
```

Why are there both doEdit() and redo()? That is, should redo() not always be the same as doEdit()?

END OF QUIZ

Extra space if needed

Extra space if needed