Use a full sheet of 8½x11" paper. (Half sheet? Half credit!)

Put only your last name in the far upper left hand corner of the paper, where a staple would hit it. (It's OK to write BIG, just start in the corner!)

**Mitchell**

AVOID A ½-POINT DEDUCTION!

Numbering responses may help you avoid overlooking a question; it's ok to go ahead and pre-number your sheet.

Feel free to abbreviate.

3 minutes; 3 points
Quiz 7, February 24, 2015
3 minutes; ½ point/answer; 3 points total

1. Other than the fact that Haskell lists can't be changed, what's a major difference between Ruby arrays and Haskell lists?

2. What's a major difference between strings in Ruby and Java?

3. In Ruby, given \texttt{s="abc"} what is the \textbf{TYPE} of \texttt{s[0]}?

4. In Ruby, given \texttt{s="testing"} what is the \textbf{VALUE} of \texttt{s[2,2]}?

5. Write a Ruby program that behaves like this:

   \begin{verbatim}
   $ ruby q7.rb
   ab
   cd
   \end{verbatim}

6. Haskell folding functions take two arguments. What are good names for them? (What does whm name them?) Don't worry about their order!
1. Other than the fact that Haskell lists can't be changed, what's a major difference between Ruby arrays and Haskell lists? Ruby arrays are heterogenous.

2. What's a major difference between strings in Ruby and Java? Three that come to mind: Ruby strings are mutable, accessible with indexing operators, and are indexable from the right.

3. In Ruby, given `s="abc"` what is the **TYPE** of `s[0]`?

   ```ruby
   >> s = "abc"; s[0]
   => "a"

   >> it.class
   => String
   ```

4. In Ruby, given `s="testing"` what is the **VALUE** of `s[2,2]`?

   ```ruby
   >> s = "testing"; s[2,2]
   => "st"
   ```

5. Write a Ruby program that behaves like this:

   ```bash
   $ ruby q7.rb
   ab
cd
   ```

   Short answer:
   ```ruby
   puts "ab\ncd"
   ```

   Long answer:
   ```ruby
   puts "ab"
   puts "cd"
   ```
6. Haskell folding functions take two arguments. What are good names for them?

   acm and elem (or val)

   Other perfect answers:
   a and e
   v, a
   value and accumulator