Smalltalk Assignment
(AddressBook design and implementation)

- Grading (70 total points possible)
  - Presentation — 10 points
  - Design — 30 points
  - Smalltalk conventions — 5 points
  - Implementation — 15 points
  - Test cases — 10 points
- Mean was a 65/70 — Nice work!

General thoughts

- Spell check!
- Read what you turn in—some printouts used the default font which is nearly impossible to distinguish between () and []
- Label all arrows on pictures

Design first… Implement later!

- **Not**: "Because we use the dictionary we do not support multiple cards with the same name"
- **Instead**: "We feel it is imperative that the user has cards with unique names, and it is easy enough to add a distinguishing middle initial or name, so we chose to use the dictionary."

Overuse of strings

- Fields could have been symbols instead of strings
- Values could have been any object, not just a string
- If it's not a free-form, arbitrary string, consider using a symbol instead: #name, #address, #phone (comparing strings is expensive)

More overuse of strings

- Return true and false (not 'yes' or 'no', and not True or False)
- Do not return strings to indicate error conditions — remember the question about dynamic typing on the midterm. Use:
  self error: 'Card for that name not found'

Printing a value

- Returning a string is **not** printing it
- Overriding the printOn: method teaching the Smalltalk system how to display objects of a class that you write — argument is a Stream object
- Never use a global variable unless you absolutely must — instead, let the user pass you the global variable (e.g., Transcript)