



Iterative Refinement



Today's Lecture

- A more generic recipe for data analysis:
 - Start from a “reasonable” type
 - Enrich it with more information as problem is better understood
 - Repeat
- What makes a type “reasonable”
 - Remember our discussion of naturals?



Modeling File Directories

- We've all seen them
- Lots of programs operate on them, like:
 - List, copy, move, delete, find, etc.
- How do we go about programming this concept?



A First Attempt

- A file is a symbol
- An entry is
 - a file or
 - a directory
- A directory is list of entries



A Second Attempt (Refine)

- A file is a symbol
- An entry is
 - a file or
 - a symbol and a directory
- A directory is a list of entries



A Third Attempt (Refine)

- A file is a symbol, a number, and data
- An entry is
 - a file or
 - a symbol and a directory
- A directory is a list of entries



A Fourth Attempt (Re-org)

- A file is a symbol, a number, and data
- A directory is
 - a symbol, a list of directories, and a list of files
- Why is this last step not a refinement?
 - In some sense, it is not a refinement
 - In some sense, it is a backward refinement