try {
    QifParser parser = new QifParser.
    parser.parseFullFile(new File("Sample\accountlist.txt"));
    Assert.assertNotNull(parser.accounts);
    Assert.assertNotNull(parser.categories);
    Assert.assertEquals(parser.accounts.size(), parser.categories.size());
} catch (IllegalArgumentException ex) {
    Assert.fail(ex.getMessage());
} catch (IOException ex) {
    Assert.fail(ex.getMessage());
}
What is Programming?

“The process of transforming a mental plan of desired actions for a computer into a representation that can be understood by the computer”

-- Jean-Michel Hoc and Anh Nguyen-Xuan
Topic List

- Program comprehension
  - information needs, code navigation, working sets, code search
- Software evolution
  - refactoring, program differencing, reverse engineering
- Development tools & environments
  - team awareness, delta debugging, visualization, DOI models, task-centric development
- Quantitative and qualitative means of evaluating software engineering research
  - experiments, case studies
Topic List

› Program comprehension
  › information needs, code navigation, working sets, code search

› Software evolution
  › refactoring, program differencing, reverse engineering

› Development tools & environments
  › team awareness, delta debugging, visualization, DOI models, task-centric development

› Quantitative and qualitative means of evaluating software engineering research
  › experiments, case studies
Assessment

- Paper presentation(s): 30%
- Paper reviews: 20%
- Project: 50%
Presentations

- ~2 presentations
  - ~30 minute talk
  - ~30 minute discussion
- ~2 presentation backup
  - Help the presenter lead the discussion
- I will go first (next week)
Paper Reviews

- Assess projects like a program committee
  - Everyone will read and review several papers
  - Reviews organized via easychair
    - [http://easychair.org](http://easychair.org)
- Program committee meeting in the last class
  - Up to you whether we ‘accept’ papers
    - ‘acceptance’ has no bearing on your grade
Projects

- Three options:
  - Build a development tool
  - Perform a literature survey
  - Evaluate a development tool
- Each will be accompanied by a paper (~6-10 pages), this will be the artifact
- Groups are encouraged (up to 3 people)
- Project proposal will be due soon (Sept 24)
  - 1 page description
- Project presentations (10 minutes)
Software Tool

- Identify a real problem faced by developers
- Model a solution
- Implement the tool that addresses your model of the problem
- Evaluate (preliminary) the tool
  - Users would be great here, but given time constraints qualitative scenarios would work
Literature Survey

- Develop a complete understanding of a specific relevant topic
  - Synthesis is key:
    - What are the relevant papers?
    - How do they contribute to our understanding?
    - What shortcomings do they identify?
    - How can future work address these?
  - This option will be evaluated the most rigorously
Evaluate an Existing Tool

- Validate a previously-existing development tool
- The tool must be relevant to the course
- Validation should involve non-author users
- Talk to me before the proposal, if you have questions
To Do

1) Get an easychair account (free)
2) Choose 2 papers you would like to present
   ‣ Insert into Google doc (by Sept 19 @ 0800)
3) Start thinking about projects
Next Week

- Two very high-level papers:
- Both available online