CMSC 433 – Programming Language Technologies and Paradigms
Spring 2003

May 13, 2003

Project 6

- Test oracle available
- 1. Peering and accepting messages
- 2. Handling slow clients
- 3. Symmetric disconnects
- 4. Does your portal forward a probe packet
- 5. Does your portal generate and forward response
- 6. Does your portal pass through a response msg
- 7. Can downloaded code connect to port 80
- 8. Can downloaded code not connect elsewhere

- Don’t forget to test with p5 oracle

Final Exam

- Pugh’s section
  - Tuesday, May 20, 10:30am-12:30pm
- Foster’s section
  - Thursday, May 22, 10:30am-12:30pm
- Topics:
  - Everything
  - More emphasis on material after midterm
  - Looks at sample midterm 2’s, finals

Java Basics

- Objects, inheritance, and interfaces
  - Mutability, equality, casts (dynamic types)
- Method dispatch
  - Overriding, overloading
  - Dynamic, static
- Exceptions
  - try, catch, finally

Project 1

- Build a toy Java web server
  - MiniServlet interface between web server and servlets
  - A little bit of reflection
  - Some generic container classes (e.g., Map)
  - Some sockets and I/O
    - If you need any java.* APIs, we’ll give them to you on the exam

Testing and Specifications

- Benefits and limitations of testing
- Black box vs. white/glass box
  - Tips on generating tests from specs
  - Statement, branch, condition coverage
  - Tradeoffs, design choices (e.g., unit vs. integration)
- Specifications
  - Issues (tradeoffs between formal and informal etc.)
  - What makes a good specification
  - Javadoc
Project 2

- Use JUnit to test Graphs
  - JUnit philosophy: full automated, test-as-you-go
  - Test case components: set up, run, tear down
  - Test suites, test runner
  - Testing against a specification

Abstraction, Types, Polymorphism

- Data abstraction
  - Abstraction function, rep invariant
- Subtypes and overloading methods
- Polymorphism
  - Subtype polymorphism
  - Parametric polymorphism with GJ

Design Patterns

- Object Modeling Technique
- Iterators
- Singleton, typesafe enum
- Abstract Factory
- Adapter
- Proxy
- Decorator
- Template
- State
- Observer
- Composite
- Strategy
- Bridge
- Command
- Visitor

Project 3

- GoogleMap (and LazyList)
  - Generic Java
  - GJ Container classes
  - Iterators and laziness
  - Abstraction

Threads

- What are threads?
- Data races and synchronization
- Wait and notify(All)
  - Producer/consumer example
- Thread cancellation
- Deadlock

Project 4

- Thread pool
  - Programming with threads in Java
  - Avoiding data races (checkSync, findBugs)
  - Avoiding deadlock
**Distributed Programming**

- Java RMI
  - Remote interfaces
  - RMI registry
  - Stubs and downloading code
  - Gnutella
- Security
  - Security managers
  - Policy files

**Project 5**

- Distributed peer-to-peer network
  - Using RMI
  - Application-level network abstraction
  - Peer-to-peer network architecture, design choices

**Special Topics**

- Graphical user interfaces
  - Using Swing, do’s and don’ts
- Reflection
  - Example applications
- Garbage collection
  - Ref counting, mark and sweep, stop and copy
  - Soft and weak pointers
- XML and ant

**Project 6**

- Enhancing project 5
  - Adding robustness
    - Slow portals, symmetric failures
  - Reverse path forwarding
  - Downloading code
    - Installing security manager, adding policy file
  - Finding routes on the network