CS227B GGP Code Base
Obtaining the Code

git clone
ssh://<yourSuNetID>@myth/afs/ir/class/cs227b/ggp-spr1112.git

You probably want to keep your code under git control in case we need to patch something.
Running the Apps

- Add JUnit4 to your build path
- Add a cache folder to the `<root>/games` folder if it doesn't exist already (this may be patched by the time you read this)
- Configure the working directory for each app to be the root folder of the project.
Basic Components

• Applications:
  • Kiosk, Tiltyard, ServerPanel, PlayerPanel, ConsoleMatchServer, ConsolePlayer

• Important classes:
  • Gdl, Game, Gamer, Statemachine, MachineState + Wrapper classes (Role, Move)
Applications

• Kiosk – lets you play against a player
• ServerPanel – sets up a server on your machine and allows you to start a match between two remote players (ConsoleMatchServer is an alternative)
• PlayerPanel – sets up a player on your machine (ConsolePlayer is an alternative)
• Tiltyard – plays several matches of the same game locally between two players.
Class: Gamer & Player

- Gamer holds the logic for playing games
- Player wraps gamer with GGP protocol
- A gamer anywhere in the project will be detected by the tools
- Override the getName() method to give your gamer a friendlier name. =D
Gamer Hierarchy

Gamer

Can not be invoked by PlayerPanel

SingleGameGamer

Can be invoked by PlayerPanel.

StateMachineGamer
Class: Gdl

- Immutable – use replace(...) if you want to generate new GDL based on existing GDL
- Immutability also implies that you can compare references directly for equality as long as you make sure you keep all of your Gdl in the GdlPool.
- getChildren() method and GdlPreorderIterator are options for iteration.
Class: Statemachine

• Abstracts computing the state, legal actions, and goals of the game
• There is an existing ProverStateMachine implementation that you want to use for this assignment
• Works the classes Role, Move, MachineState
  • Role and move are just wrappers for Gdl
• Does not cache – you are responsible for caching intelligently if you want
Adding Local Games

- At a minimum you must have the following:
  - A directory in the <root>/games/games/ folder
  - A JSON METADATA file in that folder that gives the name of the game and the relative path to the rulesheet. (Properties “name” and “rulesheet”)
  - A KIF rulesheet in the folder
- For reference there are games in the local repository that show you how to do this
tiltyard.ggp.org

• Online arena written by Sam Schreiber you can (and should) leave your player running on as much as possible.

• Part of your grade will depend on your performance on Tiltyard over the course of the quarter!
Reporting Issues

- Tiltyard bugs should be directed to action@ggp.org
- Code base bugs should be directed at course staff (Piazza or ptpham@stanford.edu)