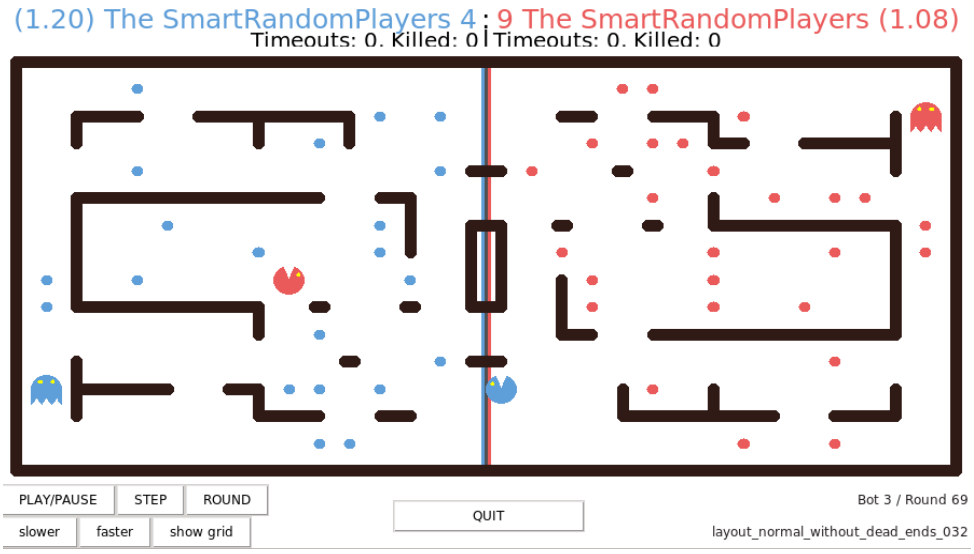
University of Zurich^UZH

Department of Physics

Scientific Programming with Python

# Pelita – Getting Started

January 22, 2015



(1.20) The SmartRandomPlayers 4 : 9 The SmartRandomPlayers (1.08)

Timeouts: 0, Killed: 0 | Timeouts: 0, Killed: 0

Bot 3 / Round 69

layout_normal_without_dead_ends_032

Pelita is a fun game, where you program an Artificial Intelligence for two Pacman bots. This document should help you to get started quickly.

1. create a directory `pelita_root` for this project
2. enter `pelita_root` and clone Pelita from github: https://github.com/ASPP/pelita.git
3. form `pelita_root` run a first example game:

   ```
   ./pelita/pelitagame
   ```

   The game will randomly choose two AI's from the `players` sub-directory. These are usually not very smart, so the game will probably end when 300 rounds are played an the time runs out. You can choose the players with command line arguments:

   ```
   ./pelita/pelitagame FoodEatingPlayer RandomPlayer
   ```

4. in `pelita_root` create a new folder (e.g. `newai`) with a suitable name for your own AI.
5. enter `newai` and initialise a git directory
6. copy one of the players into your new directory and give the file a suitable name (e.g. `NewPlayer.py`)
7. edit `NewPlayer.py` and replace the name of the class as well as the name of your team (in the `factory` method).
8. run your bots against a random player

   ```
   ./pelita/pelitagame newai/NewPlayer.py
   ```

9. commit your AI to git and start to improve the behaviour of your new AI. Below you find some hints and rules that should help you.

## Rules

- The board is divided into two halfs. Each half (and the pellets on it) belongs to one team.
- Each team consists of two bots.
- While on their own half, bots are ghosts and can eat enemy bots.
- While in the enemy half, bots are Pacmans and can eat enemy pellets (but get killed by enemy bots).
- If a Pacman gets eaten, he will respawn at the starting position.
- The goal of each team is to earn the most points. Points are awarded
  1. for eating a pellet (1 point)
  2. for eating an enemy bot (5 points)
- The game ends once one of the following happens:
  - a team eats all of the enemy pellets.
  - 300 rounds have been played.

  The team with more points wins.
- The game is turn based and the bots move one after the other, alternating between the two teams.
- Bots can only observe an enemies exact position if they are near.
- If the AI of a team needs more then 3s to calculate its next move, a random move is performed and the team gets a timeout.
- If a team has more then 3 timeouts it looses immediately.
- If the AI of a team raises an exception, the team looses immediately.

## Hints

- `pelitagame -h` prints a lot of helpful output. Read it. (For development and debugging, `--seed` and `--no-timeout` are especially useful.
- `http://aspp.github.io/pelita/` contains extensive documentation including a section explaining how to write a player.
- When writing tests, the `test` directory is very instructive.
- On the lower left of the GUI you can control the execution of the game. This might be useful to see what is going on.
- A strong strategy is to ensure not to get eaten and eat the enemy pellets as fast as possible. The basic defence against a fast eater is to sit on one of your own pellets and eat any bot that tries to get it.

## Tournament

We plan to run a Pelita tournament at the end of the Spring semester. You are invited to write your own AI, either alone or in a team, and join the tournament. If you want to join, please send an email to python@physik.uzh.ch, so we can keep you informed. We will also update the school website once we fixed the details for the tournament.

If you want to work in teams and need a git server to collaborate, please get in touch with us and we will try to help you.