RAZZ SIMULATION

Andrea Crotti

08 Febbraio 2010

Problem

Given an initial configuration (3 initial cards for player 1 and 1 for each other player) simulate a very large number of games.
For every game get the rank obtained by player 1, where poker and flushes are discarded and the rank is computed with the rules of razz.

For example running

```
./razz 3 1 A 2 3
```

Will run 10^3 simulations with only one player which has as initial cards Ace, 2 and 3.

This would be the output

```
-1:     35
5:      77
6:      112
7:      138
8:      132
9:      159
10:     124
11:     103
12:     76
13:     44
```

Implentation

General considerations

The most important thing to get a fast and well written program is to select the right data structure. We can consider those entities:

- Card simply an integer, we don't need to keep track of the suit also.
- Deck On a deck we must be able to
  - remove given cards (from 3 up to 10 in initialization phase only)
  - select and remove random cards (4 in simplified case up to 46 in the completely fair version)
- Hand On every hand we need to be able to
  - add cards
  - compute the rank

Python

We use data structures for

C

Objectives

Design a nice structure

See how to run test units in many languages

Compare the performances and the possibilities given

Write a distributed version in erlang/python and see how they behaves

Make it fast!

A nice metric would be the #simulations / seconds The output could be equal for all the different languages and still parsable and analyzable automatically.

Speed comparison

Pure python

Cython

Pure C

Links

razz rules explained