Walter Cazzola

- Home Page
- ADAPT Lab.
- Curriculum Vitae
- Research Topic

Didactics

Publications

Funded Projects

Research Projects

Related Events

Exam of Advance in Programming

03 June 2013

Disclaimer. Note that to have a running solution for an exercise is not enough: you need a well-cooked solution that proves your ability to use what explained during the classes. The worth of the 2 exercises is the same (15 points). To pass the exam you have to do both exercises. The submissions with only one exercise will not be evaluated at all.

Exercise 1: Becoming a Palindrome.

If you don't know that, a palindrome string is a string that can be read in the same way from left to right and from right to left. Of course not all the strings are palindrome but any of them can be transformed into a palindrome by adding a minimum (depending on the string) number of characters.

In this exercise, you have to write a function `palin` that given a string returns how many characters are necessary to render it palindrome and which are those characters. Please note that the correct answer minimizes the number of introduced characters and list them in (temporal) order of introduction (characters can be introduced everywhere).

The following is an example of the expected computation.

```
if __name__ == "__main__":
  print(palin("casa"))
  print(palin("otto"))
  print(palin("palindromo"))
  print(palin("posero"))
```

```
[18:02]cazzola@surtur:~/pa/es1>python3 palin.py
The word «casa» needs 1 insertions to become palindrome: ['c']
The word «otto» needs 0 insertions to become palindrome
The word «palindromo» needs 7 insertions to become palindrome:
                    ['p', 'a', 'l', 'i', 'n', 'd', 'r']
The word «posero» needs 3 insertions to become palindrome: ['p', 'r', 'e'
The word «coccinella» needs 7 insertions to become palindrome:
```

Exercise 2: Drools for the Poors.

Drools is a production rule system whose main goal is to solve constrained systems describing problems like this.

A foursome of golfers is standing at a tee, in line from left to right; each golfers wears different colored pants

- one golfer is wearing red pants;
- the golfer to Fred's right is wearing blue pants;
- Joe is second in line;
- Bob is wearing plaid pants; and
- Tom isn't in position one or four and he isn't wearing orange pants;

In what order will the four golfers tee off and what color are each golfer's pants?

The aim of this exercise is to write a class `Drools` (general) that fed with the rules and elements of the problem can (via a brute force approach) find a solution to the problem. This class, among the others, will have a method `eval` that validates all the possible solutions against the constraint the problem imposes.

Hints Currying, itertools, recursion lambdas and closures will be your best friends.

The following is how the application should work.

```
if __name__ == "__main__":
  d = Drools(rules,
     ['bob', 'joe', 'fred', 'tom'],
     ['red', 'orange', 'blue', 'plaid'],list(range(1,5)))
```

```
Golfer bob is in position 4 and wears some plaid pants
Golfer fred is in position 1 and wears some orange pants
Golfer joe is in position 2 and wears some blue pants
Golfer tom is in position 3 and wears some red pants
```

Last Modified: Fri, 14 Jun 2013 12:24:30 ADAPT Lab.