# Purely Functional Algorithm Specification
# Exercises Day 1

Jan van Eijck
CWI & ILLC, Amsterdam

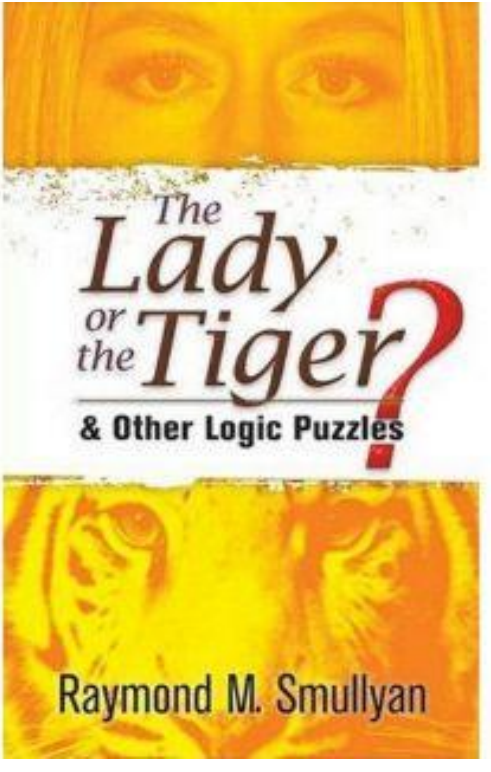
ESSLLI, Opole, August 6, 2012

homepages.cwi.nl/~jve/courses/12/esslli12/

```
module Exerc1

where
```

## Puzzles from Smullyan [1]



## The First Puzzle

There are two rooms, and a prisoner has to choose between them. Each room contains either a lady or a tiger. In the first test the prisoner has to choose between a door with the sign “In this room there is a lady, and in the other room there is a tiger”, and a second door with the sign “In one of these rooms there is a lady and in the other room there is a tiger.” A final given is that one of the two signs tells the truth and the other does not.

## Haskell Statement of the Puzzle

```
data Creature = Lady | Tiger
     deriving (Eq,Show)

sign1, sign2 :: (Creature,Creature) -> Bool
sign1 (x,y) = x == Lady && y == Tiger
sign2 (x,y) = x /= y
```

## Haskell solution

```
solution1 :: [(Creature,Creature)]
solution1 =
 [ (x,y) | x <- [Lady,Tiger],
           y <- [Lady,Tiger],
           (sign1 (x,y) && not (sign2 (x,y)))
           || (not (sign1 (x,y)) && sign2 (x,y))]
```

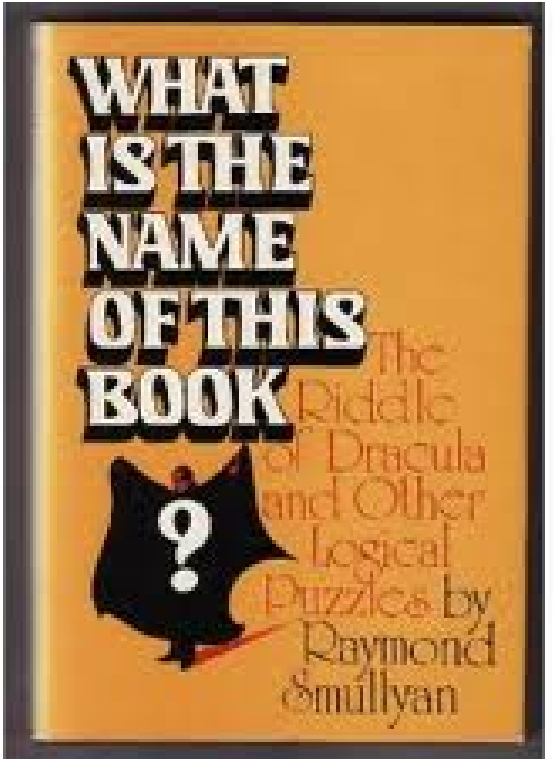
Running this reveals that the first room has a tiger in it, and the second room a lady:

```
*WLH> solution1
[(Tiger,Lady)]
```

## The Second Puzzle

The second puzzle of the book runs as follows. Again there are two signs. The sign on the first door says: “At least one of these rooms contains a lady.” The sign on the second door says: “A tiger is in the other room.” This time either the statements are both true or both false. Give a Haskell implementation of `solution2` that solves the puzzle. You will also have to write functions for the new signs, of course.

## Knights and Knaves



## The First Puzzle

On the island of knights and knaves made famous in another logic puzzle book by Raymond Smullyan [2], there are two kinds of people. Knights always tell the truth, and knaves always lie. Of course, if you ask inhabitants of the island whether they are knights, they will always say "yes."

Suppose John and Bill are residents of the island. They are standing next to each other, with John left and Bill right. John says: "We are both knaves." Who is what?

## Haskell Solution

```
data Islander = Knight | Knave deriving (Eq,Show)

john :: (Islander,Islander) -> Bool
john (x,y) = (x,y) == (Knave,Knave)

solution3 :: [(Islander,Islander)]
solution3 = [(x,y) | x <- [Knight,Knave],
                     y <- [Knight,Knave],
                     john (x,y) == (x == Knight) ]
```

This reveals that John is a knave and Bill a knight:

```
*WLH> solution3
[(Knave,Knight)]
```

## Another Knights and Knaves Puzzle

In this puzzle, again John is on the left, Bill on the right. John says: “We are both of the same kind.” Bill says: “We are both of different kinds.” Who is what? Implement a Haskell solution.

## A Puzzling Program

Use a minute or so to analyze the following program.

```
main = putStrLn (s ++ show s)
  where s = "main = putStrLn (s ++ show s) \n  where s = "
```

This has the following ingredients that may still be unfamiliar to you:

- `show` for displaying an item as a string (if the item to be displayed is already a string, then this string is quoted);
- `\n` for the newline character.

Now that this was explained to you, reflect again, and tackle the exercises below.

## Exercises

- Predict what will happen when the function `main` is executed. Next write down your prediction, and check it by executing the function.
- (Only for those who know some logic.) What does this have to do with logic? Hint: think of Kurt Gödel's famous proof of the incompleteness of the first order theory of arithmetic.

## References


[1] Raymond M. Smullyan. The Lady or the Tiger?: and Other Logic Puzzles. Dover, 2009. First edition: 1982.

[2] Raymond M. Smullyan. What is the name of this book? Dover, first edition 1990 edition, 2011.