Fundamentals of High Discipline TDD

# Introduction

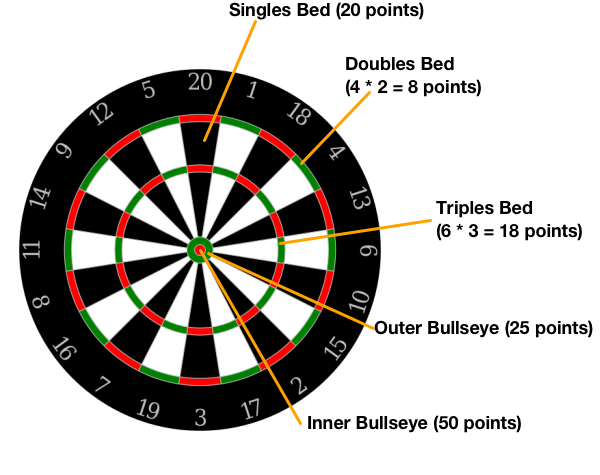
As a part of the course, you are to implement the scoring aspect of the game of darts. No knowledge of the game is assumed, scoring is explained below.

# Game Description

Darts is a throwing game in which small pointed objects are thrown at a circular board on a wall. The board is composed of 20 numbered segments and a bullseye. Each segment and the bullseye is divided into regions known as beds:

* The segments are divided into a wider, outer and narrower, inner singles bed
* The inner ring separating the inner and outer singles beds is the triples bed
* The outer ring separating the outside of the board is the doubles bed
* The bullseye is composed of two beds:
  + Inner Bullseye: 50 points
  + Outer Bullseye: 25 points

Refer to the following diagram for an example of sample scores:



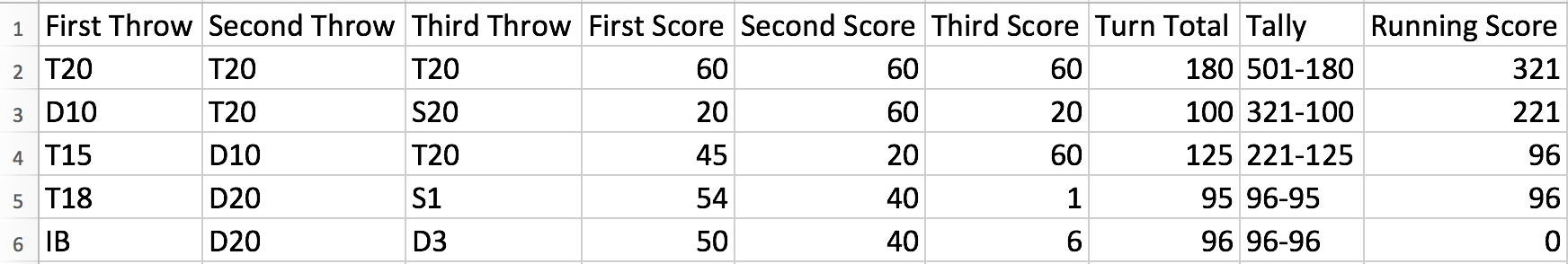
# Rules

* A game of darts begins with a score of 501
* During a turn a player must play 1, 2 or 3 darts, any darts not throw during a turn are considered to be a pass and does not score
* The game ends when the score reaches exactly zero:
  + The final “winning” throw that reaches zero, must land on either a double or the inner bullseye
  + When the score reaches exactly zero and the final throw lands on neither a double or the bullseye the score resets to the score at the beginning of the turn
* When the score reaches exactly one or passes below zero the score resets to the score at the beginning of the turn
* A dart that does not land on any bed is considered a Miss and does not score

# Notes on Scoring

* 170 is the highest score that can be won from and is reached with the following throws:
  + Triple 20 + Triple 20 + Inner Bullseye = 60 + 60 + 50 = 170
* The smallest score from which to win is two. Remember the final throw must land on a double or the inner bullseye. A final throw on the inner bullseye would put the score below zero, this leaves the only scoring possibility at double 1 for two points.

# Example Game



Note: row 5: resets to 96 because the score reached is 1 because the game cannot be won.

# The Story

As a Player I want to be able to keep track of my score after each turn

You are provided with skeleton Java and Ruby code bases, containing the following:

* Test driven model objects representing the various beds, including Misses and Passes
* A Player class which you are to implement with an API interface of the following form:
  + score(): Returns Integer
    - returns the current score
  + turn(firstThrow: ThrowObject, secondThrow: ThrowObject,  
     thirdThrow: ThrowObject): Returns Nothing
    - represents the throws taken during a turn and adjusts the score appropriately
* You do not need to implement an entire game of darts
* Consider the course facilitator to be the business owner of the story (ask them clarifying questions