



- ▶ Getting Started Overview
- ▶ W0M: Overview and Introduction to Software Engineering (Week 0, Monday Aug. 31)
- ▶ W0W: Beautifully Engineered Software, Plan & Document vs Agile (Week 0, Wednesday Sept. 2)
- ▶ W1W: Introduction to Ruby (Week 1, Wednesday Sept. 9)
- ▶ W2W: More Ruby and Intro to BDD & TDD (Week 2, Wednesday Sept. 16)
- ▼ W3M: SaaS

## HW 1-3: ROCK PAPER SCISSORS (100/100 points)

Specs: `spec/rock_paper_scissors_spec.rb`

In a game of rock-paper-scissors, each player chooses to play Rock (R), Paper (P), or Scissors (S). The rules are: Rock breaks Scissors, Scissors cuts Paper, but Paper covers Rock.

In a round of rock-paper-scissors, each player's name and strategy is encoded as an array of two elements

```
[ ["Armando", "P"], ["Dave", "S"] ] # Dave would win since S > P
```

### 1. Game Winner:

Create a `RockPaperScissors` class with a class method `winner` that takes two 2-element arrays like those above, and returns the one representing the winner:

```
RockPaperScissors.winner(['Armando', 'P'], ['Dave', 'S']) # => ['Dave', 'S']
```

If either player's strategy is something other than "R", "P" or "S" (case-SENSITIVE), the method should raise a `'RockPaperScissors::NoSuchStrategyError'` exception and provide the message: "Strategy must be one of R,P,S"

If both players use the same strategy, the first player is the winner.

### 2. Tournament:

A rock-paper-scissors tournament is encoded as an array of games - that is, each element can be considered its own tournament.

**Architecture  
and REST  
(Week 3,  
Monday Sept.  
21)**

ESaaS Ch. 2.1-2:  
The Web as a  
Client-Server  
System; TCP/IP  
intro (13:25)

ESaaS Ch. 2.3:  
HTML+CSS (9:33)

ESaaS Ch. 2.4:  
3-tier shared-  
nothing  
architecture &  
scaling (11:53)

ESaaS Ch. 2.5:  
Model-  
View-Controller  
(8:06)

**Homework 1:  
More Ruby (Due  
Tues. 9/29 at  
Midnight)**

Homework 1 due Oct  
06, 2015 at 05:00 UTC

W3.0M - Goals  
and Activities for  
Week 3, Monday

W3.2M -  
Background:  
Introduction to  
Git and HTML

W3.3M - Activities

W3.4M -  
Preparation for  
Monday, Sept. 28

► W4M: SaaS

```
[  
  [  
    [ ["Armando", "P"], ["Dave", "S"] ],  
    [ ["Richard", "R"], ["Michael", "S"] ],  
  ],  
  [  
    [ ["Allen", "S"], ["Omer", "P"] ],  
    [ ["David E.", "R"], ["Richard X.", "P"] ]  
  ]  
]
```

In the tournament above Armando will always play P and Dave will always play S. This tournament plays out as follows:

Under this scenario, Dave would beat Armando ( $S > P$ ) and Richard would beat Michael ( $R > S$ ), so Dave and Richard would play (Richard wins since  $R > S$ ); similarly, Allen would beat Omer, Richard X. would beat David E., and Allen and Richard X. would play (Allen wins since  $S > P$ ); and finally Richard would beat Allen since  $R > S$ . That is, pairwise play continues until there is only a single winner.

Write a method `RockPaperScissors.tournament_winner` that takes a tournament encoded as an array and returns the winner (for the above example, it should return `['Richard', 'R']`). You can assume that the array is well formed (that is, there are  $2^n$  players, and each one participates in exactly one match per round).

HINT: Formulate the problem as a recursive one whose base case you solved in part 1.

No files selected.

Architecture  
and REST  
(Week 4,  
Monday Sept.  
28)

► W4W: Rails  
Intro (Week 4,  
Wednesday  
Sept. 30)

► W5M: Rails  
cont. (Week 5,  
Monday Oct.  
5)

► W5W:  
Enhancing  
SaaS with  
JavaScript  
(Week 5,  
Wednesday  
Oct. 7)

► W6M: Agile  
Methodology:  
Working with  
the Customer  
(Week 6,  
Monday Oct.  
12)

► W6W: BDD  
with  
Cucumber and  
Capybara  
(Week 6,  
Wednesday  
Oct. 14)

```
On Time
RockPaperScissors
  should raise NoSuchStrategyError if strategy isn't R, P, or S [10
points]
  game
    rock breaks scissors [10 points]
    scissors cut paper [10 points]
    paper covers rock [10 points]
    first player wins if both use same strategy [10 points]
  tournament
    base case [20 points]
    recursive case [30 points]

Finished in 0.00442 seconds
7 examples, 0 failures
```

---

## SUBMIT URL TO PAIRING VIDEO (SCREENCAST)

(10 points possible)

Please submit the URL to an unlisted youtube video recording (screencast) of your pairing session on this assignment below.

?

If you are unable to access YouTube and/or G+, feel free to submit a link to a video hosted on some other service such as Zoom).

Note: we are hoping to see screencasts with screen sharing plus text chat, or even better, audio chat, but video from webcams are not required.

---

---

- ▶ W7M: TDD  
with RSpec  
(Week 7,  
Monday Oct.  
19)
- ▶ W7W: TDD  
with RSpec  
cont. and  
Review So Far  
(Week 7,  
Wednesday  
Oct. 21)
- ▶ W8M: Wrap Up  
and  
Assessment of  
Part 1 (Week 8,  
Monday, Oct.  
26)
- ▶ W8W: Project  
Poster Session
- ▶ W9M:  
Introduction to  
Part 2 and  
Advanced Rails  
(Week 9,  
Monday Nov.  
2)
- ▶ W9W:  
Advanced Rails  
(Week7,  
Wednesday,  
Nov. 4)
- ▶ W10M:

Refactoring &  
Legacy (Week  
10, Monday  
Nov. 9)

- ▶ W10W:  
Refactoring &  
Legacy (Week  
10, Nov. 11)
- ▶ W11M: Project  
Management  
(Week 11,  
Monday Nov.  
16)
- ▶ W11W: Project  
Management  
(Week 11,  
Wednesday,  
Nov. 18)
- ▶ W12M: More  
Enhancing  
SaaS with  
Javascript  
(Week 12,  
Monday Nov.  
23)
- ▶ W13M: Design  
Patterns for  
SaaS (Week 13,  
Monday Nov.  
30)
- ▶ W13W: Design  
Patterns for  
SaaS (Week 13,

Wednesday,  
Dec. 2)

- ▶ W14M:  
Practical  
DevOps:  
Deployment,  
Upgrades,  
Performance,  
Security (Week  
14, Monday  
Dec. 7)
- ▶ W14W:  
Practical  
DevOps:  
Deployment,  
Upgrades,  
Performance,  
Security (Week  
14,  
Wednesday  
Dec. 9)
- ▶ W15M:  
EarlyBird  
Project Demos  
(Monday, Dec.  
14)
- ▶ W15W-W16T:  
Project Demos  
and Final  
Exam  
(Wednesday-  
Friday and  
Monday and  
Tuesday Dec.  
16-18 and Dec  
21-22)

▶ Bonus Videos

© All Rights Reserved