

Monty Hall Problem

Description:

Here's the description of the Monty Hall Problem: https://en.wikipedia.org/wiki/Monty_Hall_problem

You can write your solution in any language, just please use git and when you start, begin with an "Initial commit" ex:

```
jmillier ~/src/monty_hall
[ > git init
Initialized empty Git repository in /Users/jmillier/src/monty_hall/.git/
jmillier ~/src/monty_hall (master)
[ > touch trials.rb
jmillier ~/src/monty_hall (master)
[ > touch results.jpg
jmillier ~/src/monty_hall (master)
[ > git add .
jmillier ~/src/monty_hall (master)
[ > git commit -m "Initial commit."
[master (root-commit) f643947] Initial commit.
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 results.jpg
 create mode 100644 trials.rb
jmillier ~/src/monty_hall (master)
[ > git log
commit f643947d93b96cb5e328ea98d271c941c3dc0af1 (HEAD -> master)
Author: Jeff Miller <jeffmiller80@gmail.com>
Date: Thu Dec 28 11:33:15 2017 -0600

    Initial commit.
```

The goal is to run a simulation of the problem 10,000 times and demonstrate that it is indeed a 2/3 chance of winning by switching your door choice vs a 1/3 chance of winning by keeping your original choice. Put these results in either results.txt or results.jpg depending on if it's text based or a screen shot

Format:

This is a "take home problem" that should take the candidate 45-60 mins to complete.

What we're looking for:

- Does the code run?
- Code design & structure (OOP, functional, etc)
- Good naming practices
- Comments on anything that's confusing
- Reasonable commit size and messages (all the code in one commit isn't great, commit descriptions should be understandable)
- How easily is the code changed? For example, is it easy to adapt the code to 4 doors vs 3? What about adding a consolation prize?
- Are exceptions handled?
- BONUS: If the code comes with tests that is great!