Homework 1

Part 1:

Implement the dining philosophers problem within a DiningPhilosophers.java file

Requirements:

- Each philosopher is represented by a thread
- Assign a name to each philosopher
- Each philosopher has access to left fork and right fork objects
- When a philosopher gets both forks, sleep for 1 second to simulate eating time
- After a philosopher is done eating, the philosopher places both forks down and their name is printed in the command line
- Implement a solution to prevent deadlocks from happening in the program

Example output:

```
Philosopher 1
Philosopher 4
Philosopher 3
Philosopher 2
Philosopher 5
```

Part 2:

- Calculate the RTP of a roulette game where a user only bets on black.
- A roulette wheel has numbers 0 - 36 as possible slots the ball can land in
- When the ball lands on black, the player receives the amount he bet x 2. If the ball lands on 0 or red, the player loses his bet amount
- RTP % calculation = credited winnings/wagered amount * 100
- In order for our simulation to be usable, the simulation needs to be run 10 million times
- To start the simulation use a command line argument to define how many threads we will use for our simulation.
    - Example: java Simulation 10
      Will start our simulation program with 10 threads

After the simulation is done running, the program will output the time it took for the simulation to run along with the calculated RTP value

Note: the Random nextInt() method is synchronous. A new Random object needs to be made for each thread

Example output:

```
Simulation Time: 100 seconds
RTP: 95%
```

Homework submission:

Upload a folder of all java files to Moodle.

Required files will be:
DiningPhilosophers.java, Roulette.java, Names.txt (Required if pair programming)
Homework grading:
If your code compiles, runs, and matches the requirements - 100%

If you code does not compile or run - You can come see me in office hours to walk through your code OR leave it up to me to assign a grade based on how many of the requirements I feel you have met in your code.