CS200: Programming I **Instructor:** Sam Vinitsky

P5: Luck of the Draw

(25 points)

Due: Tuesday, July 9 @ 11:59pm

Part 0: Introduction

Brrrring brriiiiiing the phone rings – it's Sky Masterson, local gambling aficionado. Sky used to be a world-class gambler, but lately his luck has run out. He needs your help to get back on top!

Sky attends a weekly underground gambling[1] ring run by the math professors from the local university (as most underground gambling rings are). Sky is bad at math, and needs you to write computers programs to help him play the games and win enough money to make his nemesis, Nathan Detroit, jealous.

Sky will be running your programs on a computer hidden in his pocket, which is very small and slow. They **can** run conditionals and loops, but they cannot run any of the following: String parsing, arrays, ArrayLists, the "Math" or "Random" libraries, or anything we havent explicitly covered in class. If you have questions about what you are/not allowed to use, please ask on Piazza.

Before attempting this homework, work through all of the practice problems on loops from this week's lecture (posted on Canvas here).

Each program you turn in should include a comment at the top with (1) your full name, (2) your student ID number, (3) your netID, and (4) the name of anyone you discussed the homework with (excluding Sam and Alex).

As always: **start early, ask questions, and have fun!**

[1] My lawyers have advised me to state that gambling is **illegal**, and neither I, nor the CS department, nor the University of Wisconsin - Madison, condone illegal behaviour. This assignment is not an endorsement of breaking the law, but an academic exercise. The defense "I saw it on my CS homework" will **not** hold up in court.

Part 1: The Dice Game [7 points total]

a) Playing the dice game (3 points)

Sky needs to make money fast, and decided he would start off by playing a simple game that involves rolling dice. In this game, Sky rolls 2 dice, each of which has twenty sides (numbered 1,2,3,4...20). Sky wins if the two numbers he rolls differ by **3 or less**. Sky wants to make sure that playing this game is worth his time, so he would like you to calculate his probability of winning.

Write a program called `FairDice.java` that prints out the probability that Sky wins this game. Your printout should be formatted as follows:[2]

Sky has a 45.555353% chance of winning.

Hint: *The probability that Sky wins is the* ***number of winning rolls*** *divided by the* ***total number of rolls****. To calculate this, you will need to loop through every possible roll (each of which has two die), and keep track of how many rolls Sky would win on. To get the percentage, multiply by 100. Do not round your answer.*

Remember, you may not use the Java "Random" library, and it would even not help you in any way. You need to *calculate* the probability, but if you simulated this game with the Random library, you would only get an *approximation*. If this paragraph meant nothing to you, that's okay.

b) Cheating at dice (4 points)

Sky is tired of losing. He is not an honest man, so he has decided to cheat by bringing along a **third** die. Sky will roll all three dice at once, and then check to see if **any two** of the dice differ by 3 or less (he will then quickly remove the third die, so the people he's playing with don't notice that there was an extra die). Sky wants to know the probability that he wins this game by cheating.

Write a program called `CheatyDice.java` that prints out the probability that Sky wins this game (by cheating). Your printout should be formatted as follows: [3]

Sky has a 45.555353% chance of winning if he cheats.

[2] Note: this may not be the correct percentage

[3] Note: this *still* may not be the correct percentage

Part 2: The Boat Game [10 points total]

Sky got caught cheating at dice, and has been relegated to a game that he couldn't possibly cheat at: **The Boat Game**. This game involves a room full of water with a toy boat in it. Game-play proceeds as follows:

- First, Sky bets an integer number of dollars **greater then 1**.
- The water level is set to a height equal to the number of dollars Sky bet.
- Then, the follow happen **repeatedly**:
  - If the water level is exactly 1, then the game ends.
  - If the water level is **even**, then the water level is changed to be **half** of what it was before.
  - If the water level **odd**, then the water level is changed to be **three times** what it was before, **plus one**.
- Sky earns an amount of money equal to the **total number of times the boat's height was changed** over the course of the game.

In summary: (1) Sky **pays** X dollars to set the water level to a height of X. (2) The water rises and falls the above pattern until it reaches a height of exactly 1. (3) Sky **earns** an amount of money Y equal to number of times the water level was changed. Note that Sky's total profit here is $Y - X$

a) Determining the water levels (5 points)

Since Sky is dishonest, he believes that other people try to cheat as well. He would like a way to figure out if the people running the boat game are changing the water level as expected, or if they are cheating and using a different scheme. He needs you to write a program to **compute the sequence of water levels**. Since Sky might make a variety of different bets, he needs you to do this **for each initial bet he could make**. Since Sky has been losing a lot, he only has $200 left on him – meaning you only need to compute the sequence for every integer from 2 to 200.

Write a program called `BoatLevels.java` that prints out, for each amount Sky could bet, the sequence of levels you would expect from the boat game.

Your program should print out the sequences formatted in this fashion:

```
Initial Bet: $2
Level Sequence: 2, 1

Initial Bet: $3
Level Sequence: 3, 10, 5, 16, 8, 4, 2, 1

Initial Bet: $4
Level Sequence: 4, 2, 1
```

Your print formatting should match this example **exactly** (except it doesn't need to be bolded...) Note the comma and space after each number (except the 1).

Hint: *You might want to write a function called `printBoatLevels(int n)` that prints out the level sequence for an initial bet of `n`, and call it repeatedly for each possible amount Sky could bet...*

Hint 2: *`printBoatLevels` should use a **while** loop...*

b) What's the best bet? (5 points)

Sky doesn't like losing, so he wants you to figure out which bets he should make. Write a program called `BoatBestBet.java` that, for each amount of money that Sky could bet, prints out the **length** of the boat sequence and **the amount of profit Sky would make** on such a bet. Remember that Sky only has $200 on him! **At the end, include a single line stating which bet would give him the most profit, and what that profit would be.**[4]

For example, if Sky only had $4 on him, the output would be:

```
Initial Bet: $2
Number of Height Changes: 1
Total Profit: $-1

Initial Bet: $3
Number of Height Changes: 7
Total Profit: $4

Initial Bet: $4
Number of Height Changes: 2
Total Profit: $-2

Sky's best bet is $3, which would earn him a profit of $4
```

[4] If there are two bets that give the same profit, choose the bet that is smaller.

Your print formatting should match this example **exactly**. You do not need to include commas in your numbers when they are more than 3 digits. Notice that if he were to bet $4, he would was a **negative** profit (i.e. he would lose money).

Hint: *You might want to write a function called `computeNumberOfChanges(int n)` that returns an integer equal to the number of height changes for an initial bet of `n`.*[5] *Then you can call this function for each possible amount Sky could bet, and do something to find the largest...*

Part 3: Finding the Casino [5 points total]

After winning an inordinate amount of money at the boat game with your help, Sky is in trouble! The powers that be have caught on to his tricks, and have kicked him out of the casino for good. However, he has plans to sneak his way back into the casino. The issue, however, is that the casino moves location each night, in order to prevent the police from catching them.

Normally, the gamblers are told where the casino will be each night in a mass email. Since Sky is no longer welcome at the casino, they have removed him from the mailing list! Luckily, the math professors in charge of the casino are lazy, and use an **algorithm** to determine the location of the casino each night. Sky has figured out that algorithm, and needs you to use it to calculate where the casino will be in the future!

The algorithm is as follows: first, the entire city is mapped to the x-y coordinate plane. This means that each location is associated with two numbers – an x position, and a y position. Each day, the coordinates for tonight's casino are determined as follows:

- First, tonight's x position is determined:
  - If last night's x position is divisible by last night's y position, then tonight's x position is equal **to last night's x divided by last night's y**.
  - Otherwise, tonight's x position is equal to **last night's x times last night's y**.
- Now, tonight's y position is determined:
  - If last night's y position is **less than 10**, then tonight's y position is that **plus 1**.
  - Otherwise, tonight's y position is last night's y position **minus 9**.

The last day Sky was at the casino, it had coordinates $x = 11$ and $y = 5$. Write a program called `FindCasino.java` that prints out the location of the casino for an entire year (i.e. 365 days) after Sky was kicked out.

[5] This will be similar to `printBoatLevels()`, but with minor modifications.

Your printout must be formatted as follows:

```
Day 1: x=55, y=6
Day 2: x=330, y=7
Day 3: x=2310, y=8
Day 4: x=18480, y=9
Day 5: x=166320, y=10
Day 6: x=16632, y=1
Day 7: x=16632, y=2
Day 8: x=8316, y=3
Day 9: x=2772, y=4
```

Notice that Day 1 is the night **after** Sky was kicked out. Make sure the print 365 day's worth of locations, so Sky can keep attending the casino for a whole year!

Part ∞: Feedback Form [3 points]

Fill out this feedback form **after you submit** this assignment. Completion of this will count towards your grade, but your responses themselves will not affect your grade in any way (so be honest!).

What to turn in

On Canvas, turn in a zip folder named `<your_net_id>_P5.zip` containing the files:

- `FairDice.java` [3 points]
- `CheatyDice.java` [4 points]
- `BoatLevels.java` [5 points]
- `BoatBestBet.java` [5 points]
- `FindCasino.java` [5 points]

Also complete the feedback form. [3 points]
