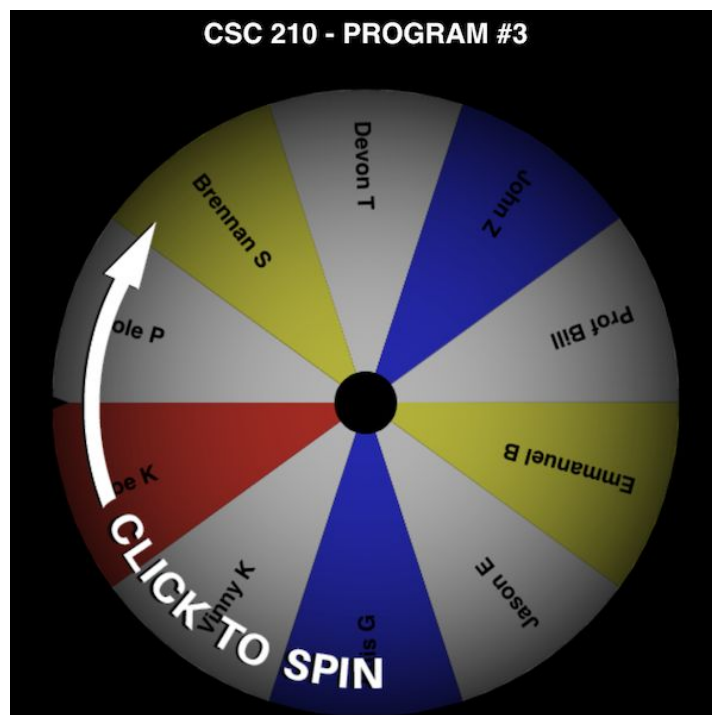# Program #3 - Wheel gui

Prof Bill - Mar 2020

Program #3 logistics:

- Due: **Fri Apr 3, 2020** at the beginning of class

- Worth: **8 points** (8% of your grade)

- Learn: JavaFX, event-driven coding, UX design, Eclipse IDE, Java inheritance

1. Description

Add a gui to your Wheel of decision from Program #1. You can use the real deal for some good gui ideas, wheeldecide.com.

thanks… yow, bill

2. Design discussion

Please use your console wheel from Program #1. If you need to make changes to get him to work, then do that. But your console wheel should still run throughout this process.

Info on the original wheel, Program #1, is at the Programs page. wtkrieger.faculty.noctrl.edu/csc210-spring2020/programs.html

Here's a TODO list for your Wheel gui:

❏ **Spin** the wheel (special effects up to you)

❏ **Reload** wheel items after spinning

❏ **Add item** to the wheel

❏ **Name** the wheel (show this name somewhere in your gui)

❏ **Clear** the wheel, removing all items and leaving the wheel empty

❏ **Reverse** items in wheel

❏ **Report** on wheel; text report that lists items, num items, first item, last item, etc

You have three BIG tasks:

1. Design your gui: draw on paper first!

2. Learn our new environment: Getting/installing JavaFX might be a drag.

3. Code 'er up: the fun part

3. Requirements

Program #3 requirements are:

➢ Write your program in **Java**.

➢ I will only accept **quality code**: [Java coding guidelines](#).

How to succeed (writing any program):
1. Start early!
2. Don't be shy. Ask a question in class. Email me. Come to office hours.
3. Small bites. Divide and conquer your program into small, manageable tasks.
4. ABW. Always be working. Your program should always compile and run. Never leave your work in disarray.

4. Grading

Special: I want each of you to demo your Wheel Gui for me in person. I'll setup a 10-15 min meeting with each of you where we'll grade your work...together! (gasp)

Still do this though...create a **program3** folder on your k: drive. It should contain:
- All your Java source files
- A **README.txt** file...that follows my template

Remember our **plagiarism** guidelines as well. Getting help from google or stackoverflow or a friend is OK, but:

1. You must acknowledge any help you receive with a comment in your code.
2. You must understand any code in your solution.
3. Get help on program components, not the assignment (tic tac toe philosophy).
4. Questions about this...contact me **before** you turn in your work, not after.

thanks… yow, bill