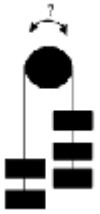
Name: **Section:**

Practice Exam2: input, output, if, loops, vectors, figures.

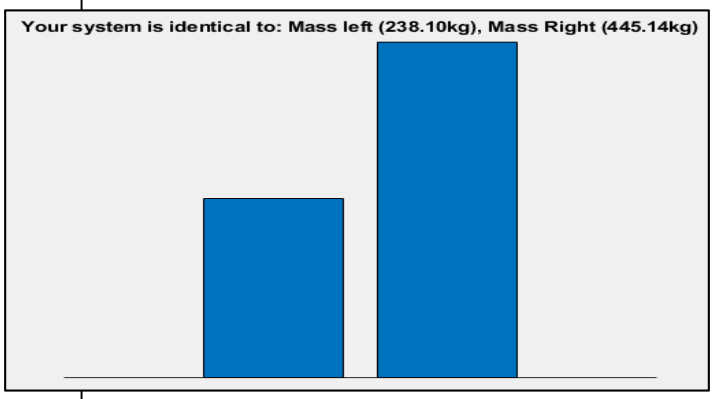
As a reminder, no credit is given to a method that has not yet been taught. Baby steps….
I do not answer questions the first hour (unless related to machine issues).



Weights are attached to a pulley and held in place by someone. Instantaneously, all weights are released and the system accelerates either rotating clockwise or counterclockwise, or does not move when the weights cancel out on each side. Calculate and show the acceleration and the direction.

```
          *<<<<< Oh how fast do you go? >>>>>>*

How many weights total are attached (>3): -1
     ERROR (whole>3). try again -->: 0
     ERROR (whole>3). try again -->: 4.4
     ERROR (whole>3). try again -->:
     ERROR (whole>3). try again -->: 3
     ERROR (whole>3). try again -->: 5

Mass #1:                  192.28 kg   (Right side)
Mass #2:     (Left side)   81.02 kg
Mass #3:                   93.29 kg   (Right side)
Mass #4:     (Left side)  157.09 kg
Mass #5:                  159.57 kg   (Right side)

          _______________ TOTAL _________________
               (238.10kg)              (445.14kg)

The system is rotating clockwise with an acceleration of 2.97m/s².
```

← A sample run



Code Requirements:

An intro message is always required.

The number of weights is a user input. **Trap** while invalid. What is considered invalid is shown in the sample run above.

The mass of each weight is automatically generated by MATLAB: a random **float** between 10kg and 200kg.

Whether the mass is on the left or right side of the pulley is also determined by MATLAB each time: generate a random **integer** 0 or 1. As the programmer, you decide: one value means left side, the other means right side.

The math: To determine the acceleration:

$$a = \frac{sum\ of\ masses\ on\ right - sum\ of\ masses\ on\ left}{sum\ of\ all\ masses} * 9.81$$

To determine the direction: based on the equation above, if a is positive, the rotation is clockwise. If a is zero, the system is in equilibrium, etc…

Requirements for the command window display:

It is crucial the output looks professional and spaced out as in all our labs.

1) The mass's number (1,2,3,…. Etc) must show.
2) A clear visual of whether the weight is on the left or the right is required (You do not have to match my sample run, but it must be OBVIOUS and somewhat a table looking output).
3) However you decide to show the values, the mass values must align at the decimal point.
4) Units are required everywhere. All values show the same amount of decimals (you chose at least >1).
5) Once calculated, the total mass of each side must also show (At your preference where and how.)
6) The final direction and acceleration show properly. Note that the acceleration must always be displayed as a positive (or zero) value, and the squared (from "meters per second squared") displays as the exponent. It is the number 178.

Requirements for the bar graph display: Use `bar(yourvector);` where the vector has both mass totals. Create a dynamic title that shows the equivalent masses on both sides, and get rid of the axis as it serves no purpose.

Check list on the back. – read before starting

The basic:

5pts	Valid Filename (up to you)
10pts	Code runs. Comment out what crashes but leave it!
6pts	Name, section, valid description
2pts each	Necessary clean up commands
10pts	**Complete algorithm as comments – requires no programming knowledge. Just English.**
10pts	Descriptive variable names. NO single letters except for the loop control variable.
10pts	Space out the code as shown in all examples/solutions. Be consistent.
5pts	Proper use of semi-colons. (0 or 5pts. 1 rule to know)
5pts	A valid intro message
10pts	Human factor: space out the output. Lots of \n\n, tabs, spaces, field widths.

The solution:

5pts	Ask user for quantity of weights
10pts	Trap for all invalid options
10pts	Loop to display each mass data (loop of your choice)
5pts	Generate random mass
5pts	Generate random side of the mass
10pts	Valid displays, numbers align, obvious side of the mass
10pts	Adding up the mass on the correct sides
5pts	Display the total mass of each side
5pts	Determine acceleration
10pts	Correct display of acceleration (decimals, positive) and direction
10pts	Create bar graph, proper axis
5pts	Dynamic title shows the equivalent masses
10pts	Leeway

The engineer:

Step7c. Twice. Located at bottom of the script file. If something looks off, write a note!

10pts	1 test must show the loop traps ALL invalid values in one run.
10pts	Another test for the rest of the code

Do not wait until the code runs to do this. Even fully testing your trap loop is 1 valid test. Crash tests are ok too if towards the end of the time (then comment what crashes before submitting).