**ENED 1090: Engineering Models I**

**Homework Assignment #4**

**Due: Week of September 28th at the beginning of your Recitation Section**

**Problem 1: Indexing into 1-d Arrays & Conditional Statements**

Create a script file which does the following:

* Creates a vector t = [ -1 3 8 7 2 ]
* Uses an **if … else … end** conditional statement to determine if the first entry in t (i.e., t(1)) is negative. If the first entry is negative, add an fprintf statement to tell the user that t(1) is indeed negative. If the first entry is not negative, add an fprintf statement to tell the user that t(1) is non-negative. *Note: Obviously, you can look at t(1) and see that it is negative but you need to construct an if …. else statement to test t(1) in MATLAB!*
* Uses an **if … else … end** conditional statement to determine if the 2nd entry in t (i.e., t(2)) is between 6 and 10. If the 2nd entry is between 6 and 10, add an fprintf statement to indicate this. If not, add an fprintf statement indicating that the 2nd entry is not between 6 and 10. *Again, since the 2nd entry is 3, it obviously is not between 6 and 10, but you need to write a conditional statement that will determine this for you.*
* Uses an **if … elseif … else … end** conditional statement to determine if the 2nd entry is greater than the 4th entry, less than the 4th entry, or equal to the 4th entry. Include fprintf statements that display the values for t(2) and t(4) and the result (greater than, less than, or equal to). *Note: in your fprintf statement, don’t hardcode the values for t(2) and t(4), instead insert them. Something like this:*

*fprintf(‘t(2) = %i and t(4) = %i so … \n’,t(2),t(4)).*

**Run your script.**

**PASTE RESULTS HERE:**

t(1) is negative.

t(2) is not between 6 and 10.

t(2)=3 and t(4)=7 therefore t(2) is less than t(4).

**PASTE SCRIPT HERE:**

%Models HW4 Problem 1

clear; clc;

t = [-1 3 8 7 2]; %input

%check sign status of t(1)

if t(1) < 0

t1\_sign = 'negative';

else

t1\_sign = 'non-negative';

end

fprintf('t(1) is %s.\n',t1\_sign);

%check if t(2) is between 6 and 10

if t(2) > 6 && t(2) < 10

t2\_location = 'between 6 and 10';

else

t2\_location = 'not between 6 and 10';

end

fprintf('t(2) is %s.\n',t2\_location);

%compare t(2) and t(4)

if t(2) < t(4)

t2t4\_compare = 'less than';

elseif t(2) == t(4)

t2t4\_compare = 'equal to';

else

t2t4\_compare = 'greater than';

end

fprintf('t(2)=%i and t(4)=%i therefore t(2) is %s t(4).\n',t(2),t(4),t2t4\_compare);

**Problem 2: I/O statements & Conditional Statements**

In Homework #3, Problem 2, you wrote a script file that did the following:

* Prompted the user for Name, Age, City, State, and Zip Code using a series of input statements
* Used fprintf statement(s) to output the information entered by the user
* Used a menu statement to ask the user if the information entered is correct.

In this problem, you will build on that script file. Add a conditional statement (**if … else**) to your script file after the menu statement that accomplishes the following:

* If the user indicates that the information entered is correct, output a statement thanking or congratulating the user for entering the information correctly.
* Otherwise (else), add a menu statement with buttons for Name, Age, City, State, and Zip Code allowing the user to indicate which item was entered incorrectly.
* Add a conditional statement (**switch or if … elseif … elseif … elseif … else**) that prompts the user to re-enter the item that was entered incorrectly the first time. *Note: if the user messed up more than one item, they are out of luck until we learn while loops.*
* Right before the **end** of this conditional statement, repeat your fprintf statement(s) to output all of the information entered by the user.

**TEST CASE #1: Run your script and when asked if the information is correct, answer yes.**

**Paste the Resulting Output here:**

Your name is Jonathan Kenney.

Your age is 18.

You live in the city Cincinnati.

You live in the state Ohio.

Your zip code is 45219.

You have entered your information correctly. Congratulations!

**TEST CASE #2: Run your script and when asked if the information is correct, answer no. When asked which item is incorrect, pick Name and enter a different (or corrected) Name.**

**Paste the Resulting Output here:**

Your name is Jonathon Kenney.

Your age is 18.

You live in the city Cincinnati.

You live in the state Ohio.

Your zip code is 45219.

SIDE NOTE: I am assuming I am not meant to get the fprintf statements here because the instructions have us put them in the last case. Otherwise, I would have put the fprintf statements after the end to make it display here. (My name is spelled Jonathan not Jonathon)

**TEST CASE #3: Run your script and when asked if the information is correct, answer no. When asked which item is incorrect, pick Zip Code and enter a different (or corrected) Zip Code.**

**Paste the Resulting Output here:**

Your name is Jonathan Kenney.

Your age is 18.

You live in the city Cincinnati.

You live in the state Ohio.

Your zip code is 45218.

Your name is Jonathan Kenney.

Your age is 18.

You live in the city Cincinnati.

You live in the state Ohio.

Your zip code is 45219.

**PASTE SCRIPT HERE:**

%Models Hw4 Problem 2

clear; clc;

%Input

Name = input('Please enter your full name: ','s');

Age = input('Please enter your age: ','s');

City = input('Please enter your city of residence: ','s');

State = input('Please enter your state of residence: ','s');

Zip\_Code = input('Please enter you zip code: ','s');

%Output

fprintf('Your name is %s.\n',Name);

fprintf('Your age is %s.\n',Age);

fprintf('You live in the city %s.\n',City);

fprintf('You live in the state %s.\n',State);

fprintf('Your zip code is %s.\n',Zip\_Code);

Verify\_Info = menu('Is this information correct?','Yes','No');

%Corrects incorrect information

if Verify\_Info == 1

incorrect\_item = 0;

else

incorrect\_item = menu('Which information is wrong?','Name','Age','City','State','Zip Code');

end

switch incorrect\_item

case 0

fprintf('You have entered your information correctly. Congratulations!\n');

case 1

Name = input('Please enter your full name: ','s');

case 2

Age = input('Please enter your age: ','s');

case 3

City = input('Please enter your city of residence: ','s');

case 4

State = input('Please enter your state of residence: ','s');

case 5

Zip\_Code = input('Please enter you zip code: ','s');

fprintf('Your name is %s.\n',Name);

fprintf('Your age is %s.\n',Age);

fprintf('You live in the city %s.\n',City);

fprintf('You live in the state %s.\n',State);

fprintf('Your zip code is %s.\n',Zip\_Code);

end

**Problem 3:**

In Engineering Foundations (1020), simple series and parallel circuits have been covered.

For two resistors (R1 and R2) in series:

* The total resistance is the sum of R1 and R2: Rtotal = R1 + R2
* The current through each resistor is the Applied Voltage divided by total resistance: IR1 = IR2 = I = Applied Voltage/Rtotal
* The voltage across each resistor is current times resistance: VR1 = R1\*I and VR2=R2\*I

For two resistors (R1 and R2) in parallel:

* The total resistance is: Rtotal = R1\*R2/(R1 + R2)
* The voltage across each resistor is the Applied Voltage: VR1=VR2=Applied Voltage
* The current through each resistor is the Applied Voltage divided by the resistance: IR1=Applied Voltage/R1 and IR2 = Applied Voltage/R2

Write a script that will do the following:

* Ask the user with a menu statement whether the circuit is series or parallel
* Prompt the user to enter the two resistor values (input statements)
* Prompt the user to enter the Applied Voltage.
* Use a conditional statement based on the menu choice (series or parallel), and calculate the total resistance, the current through each resistor, and the voltage across each resistor.
* Use fprintf statements to display the results. **Include units!**

**TEST CASE #1: Series Circuit, R1 = 810 Ohms, R2 = 560 Ohms, Applied Voltage = 15 V.**

**Paste the resulting output here:**

The total resistance is 1370.00 Ohms.

The current through R1 is 0.01 A.

The current through R2 is 0.01 A.

The voltage across R1 is 8.87 V.

The voltage across R2 is 6.13 V.

**TEST CASE #2: Parallel Circuit, R1 = 810 Ohms, R2 = 560 Ohms, Applied Voltage = 15 V.**

**Paste the resulting output here:**

The total resistance is 331.09 Ohms.

The current through R1 is 0.02 A.

The current through R2 is 0.03 A.

The voltage across R1 is 15.00 V.

The voltage across R2 is 15.00 V.

**PASTE SCRIPT HERE:**

%Models Hw4 Problem 3

clear; clc;

circuit\_config = menu('Is the circuit series or parallel?','Series','Parallel');

r1\_resistance = input('What is the resistance (Ohms) of the first resistor? ');

r2\_resistance = input('What is the resistance (Ohms) of the second resistor? ');

source\_voltage = input('What is the applied voltage (V)? ');

switch circuit\_config

case 1

total\_resistance = r1\_resistance + r2\_resistance;

r1\_current = source\_voltage / total\_resistance;

r1\_voltage = r1\_resistance \* r1\_current;

r2\_current = r1\_current;

r2\_voltage = r2\_resistance \* r2\_current;

case 2

total\_resistance = (r1\_resistance \* r2\_resistance)/(r1\_resistance + r2\_resistance);

r1\_voltage = source\_voltage;

r1\_current = source\_voltage / r1\_resistance;

r2\_voltage = source\_voltage;

r2\_current = source\_voltage / r2\_resistance;

end

fprintf('The total resistance is %0.2f Ohms.\nThe current through R1 is %0.2f A.\nThe current through R2 is %0.2f A.\nThe voltage across R1 is %0.2f V.\nThe voltage across R2 is %0.2f V.\n',total\_resistance,r1\_current,r2\_current,r1\_voltage,r2\_voltage);

**Problem 4: Under-Over 7 Dice Game**

Under-Over 7 is a very simple game played with two dice. The player simply bets on whether the sum of the two dice when rolled will be Under 7, Over 7, or Equal to 7.

Write a script that does the following:

* Includes this command: rng(‘shuffle’). This command randomly seeds the random function you will be using.
* Prompts the user (input statement) for how much money he/she wishes to wager
* Prompts the user (menu statement) to place his/her bet: Under 7, Over 7, Equals 7
* Rolls the dice: Dice1 = randi([1 6],1); Dice2 = randi([1 6],1); *Note: randi([1 6],1) creates one integer value in the range of 1 to 6.*
* Outputs to the user (fprintf) the value of each dice roll and the sum of the two rolls.
* Uses a conditional statement to determine if the user won or lost. If the user lost, use an fprintf statement indicating the loss. If the user won, use an fprintf statement indicating the win and the payout. The payout depends on the bet. If the player correctly bet Under 7 or Over 7, the payout is 1:1 (i.e., if I bet $1, I win $1). If the player correctly bet Equal 7, the payout is 4:1 (i.e., if I bet $1, I win $4).

**Run your script several times to test it.**

**PASTE ONE SAMPLE OUTPUT HERE:**

Die one reads 2 and die two reads 5.

The dice read 7.

You win! The payout is four-to-one so you receive $20.

**PASTE SCRIPT HERE:**

%Models Hw4 Problem 4

clear; clc;

%input

rng('shuffle')

wager = input('What is your wager ($)? ');

choice = menu('Under 7, 7, or over 7? ','Under 7','7','Over 7');

die1 = randi([1 6],1);

die2 = randi([1 6],1);

roll\_value = die1 + die2;

fprintf('Die one reads %i and die two reads %i.\n',die1,die2);

fprintf('The dice read %i.\n',roll\_value);

if (roll\_value < 7 && choice == 1) || (roll\_value > 7 && choice == 3)

winnings = wager;

fprint('You win! The payout is one-to-one so you receive $%i.\n',winnings);

elseif (roll\_value == 7 && choice == 2)

winnings = wager\*4;

fprintf('You win! The payout is four-to-one so you receive $%i.\n',winnings);

else

loss = wager;

fprintf('You lose. You have lost $%i.\n',loss);

end

**Turn In:**

**The word (or pdf) document with your plots, MATLAB commands, and answers to questions.**