CS1132 Spring 2016 Assignment 1a Due Feb 19th

> *Adhere to the Code of Academic Integrity.* You may discuss background issues and general strategies with others and seek help from course staff, but the implementations that you submit must be your own. In particular, you may discuss general ideas with others but you may not work out the detailed solutions with others. It is never OK for you to see or hear another student's code and it is never OK to copy code from published/Internet sources. If you feel that you cannot complete the assignment on you own, seek help from the course staff.

When submitting your assignment, follow the instructions summarized in Section 3 of this document.

Do not use the `break` statement in any homework or test in CS1132.

1 Lottery Tickets

A company lauched a new type of lottery tickets. A lottery ticket of this type costs $2 and has 4 numbers on it, each of which is an integer in [0, 9]. A drawing is held every week to produce a particular sequence of four numbers as the jackpot. The prizes are determined by comparing the numbers on a ticket with the jackpot numbers:

- Grand prize: all four numbers match the jackpot ($5000).
- First prize: three numbers match the jackpot ($100).
- Second prize: two numbers match the jackpot ($10).
- Third prize: one number matches the jackpot ($2).

Note that we compare the numbers in corresponding positions. For example, if the jackpot is [0, 2, 4, 7], then [0, 2, 5, 1] wins the second prize and [2, 0, 5, 1] wins no prize.

In this assignment, you are going to write functions for generating random lottery ticket numbers, determining prizes and investigating the chances of winning the prizes.

1.1 Generating a random number

Using the `help` command, learn about the `rand, ceil, fix`, and `floor` functions. After reading their brief descriptions, implement a random integer generator using those functions only. Implement the following function:

```
function result = myRandInt(startInt,endInt)
% Returns an integer from startInt to endInt, inclusive, with equal probability.
% If startInt > endInt, use startInt as the upper bound and endInt as the lower
% bound.
```

1.2 Generating random ticket numbers

Now you are asked to implement the following function that generates a random sequence of numbers for a lottery ticket. Make effective use of function **myRandInt** from the previous problem.

```
function result = randTicketNumbers()
% Returns a 1d array that consists of four random integers in [0, 9], representing
% the numbers on a lottery ticket.
```

1.3 Checking what prize your win

Your next task is to determine the prize you win given the numbers on a ticket and the jackpot numbers. Implement the following function as specified.

```
function [prize, amount] = checkPrize(ticketNum, jackpotNum)
% Determine what prize you win given a ticket and the jackpot numbers.
% prize is 0 for grand prize, 1 for first prize, 2 for second prize, 3 for third prize
% and 4 for no prize.
% amount is the amount of money gained by this ticket (e.g. 2 for third prize).
% ticketNum and jackpotNum are 1d arrays, each of which consists of four random integers
% in [0, 9].
% ticketNum represents the numbers on a lottery ticket.
% jackpotNum represents the numbers that win the grand prize.
```

1.4 Your chances of winning the prizes

In this part, you will use simulation to estimate the probability of winning the prizes. Write a script named **simulateLottery.m** to simulate the results of a large number (e.g. 100000) of lottery tickets, making efficient use of **randTicketNumbers** and **checkPrize**. In this script, do the following:

1. Use **randTicketNumbers** to generate a random jackpot number array.
2. Call **randTicketNumbers** for `n` times to generate `n` tickets. `n` should be a large number (e.g. 100000). For each lottery ticket, use **checkPrize** to determine what prize is won.
3. The script should keep a vector `count` of length 5, to record the number of times that each prize occurs. That is, `count(5)` is the number of times that the grand prize is won; `count(4)` for first prize, `count(3)` for second prize, `count(2)` for third prize and `count(1)` for no prize.
4. Draw a histogram showing the number of occurrence of each prize. Since you already have this data in the `count` vector, you can use MATLAB's built-in function `bar` to draw the bar graph. Calling `bar(count)` will create and show a graph with 5 bars. Add a meaningful title and y-axis label to the figure by using built-in functions `title` and `ylabel`. And use the following two lines of code to add labels for the x-axis.

   ```
   labels = {'No Prize', 'Third Prize', 'Second Prize', 'First Prize', 'Grand Prize'};
   set(gca, 'XTick', 1:5, 'XTickLabel', labels);
   ```

5. Compute the frequency of winning each prize, which is the number of occurrence over the number of total trials.
6. Compute the average amount of money won by each ticket.
7. Print the statistics of frequencies and average amount on the screen. The text should be neatly formatted and accompanied by English messages explainning what the results represent, like the following example:

   ```
   The average amount of money won by each ticket is $X.XX.
   Grand prize occurred X times in X trials. Frequency: 0.XX.
   First prize occurred X times in X trials. Frequency: 0.XX.
   Second prize occurred X times in X trials. Frequency: 0.XX.
   Third prize occurred X times in X trials. Frequency: 0.XX.
   ```

2 Self-check list

The following is a list of the minimum *necessary* criteria that your assignment must meet in order to be considered *satisfactory.* Failure to satisfy any of these conditions will result in an immediate request to resubmit your assignment. Save yourself and the graders time and effort by going over it before submitting your assignment for the first time.
Note that, although all of these are necessary, meeting all of them might still not be *sufficient* to consider your submission satisfactory. We cannot list everything that could be possibly wrong with any particular assignment!

- Δ Comment your code! If any of your functions is not properly commented, regarding function purpose and input/output arguments, you will be asked to resubmit.

- Δ Suppress all unnecessary output by placing semicolons (;) appropriately. At the same time, make sure that all output that your program intentionally produces is formatted in a user-friendly way.

- Δ Make sure your functions names are *exactly* the ones we have specified, *including* case.

- Δ Check that the number and order of input and output arguments for each of the functions matches exactly the specifications we have given.

- Δ Test Each one of your functions independently, whenever possible, or write short scripts to test them.

- Δ Check that your scripts do not crash (i.e., end unexpectedly with an error message) or run into infinite loops. Check this by running each script several times in a row. Before each test run, you should type the commands `clear all; close all;` to delete all variables in the workspace and close all figure windows.

3 Submission instructions

1. Upload files `myRandInt.m`, `randTicketNumbers.m`, `checkPrize.m` and `simulateLottery.m` to CMS in the submission area corresponding to Assignment 1a in CMS.

2. Please do not make another submission until you have received and read the grader's comments.

3. Wait for the grader's comments and be patient.

4. Read the grader's comments carefully and think for a while.

5. If you need to resubmit, fix all the problems and go back to Step 1! Otherwise you are done with this assignment. Well done!