Table of Contents

ECE 498 - Matlab Homework #4

Author: Derek Haas

```
clear;
close all;
clc;
```

Question 1: Leap Year Calculator

```
% Make an array of years 1-5000.
start = 1;
last = 5000;
years = start:1:last;

% if (year is not divisible by 4) then (it is a common year).
% Meaning -> if((year % 4) != 0) it is common.
% Meaning -> if((year % 4) == 0) it is a leap year.
numLeap = mod(years,4) == 0;

% else if (year is not divisible by 100) then (it is a leap year).
% Meaning -> if((year % 100) != 0) it is a leap year.
% Subtract out the common years from the leap years to use ==.
numLeap = numLeap - (mod(years,100) == 0);

% else if (year is not divisible by 400) then (it is a common year).
% Meaning -> if((year % 400) != 0) it is common.
% Meaning -> if((year % 400) == 0) it is a leap year.
numLeap = numLeap + (mod(years,400) == 0);

% Sum all the leap years together.
numLeap = sum(numLeap);
fprintf("Number of leap years between %d and %d: %d\n", start, last,
 numLeap);
```

```
Number of leap years between 1 and 5000: 1212
```

Question 2: Table Tennis Tournament

```
% Team 1:
A = 1; B = 2; C = 3;
Team_1 = [A, B, C];

% Team 2:
```

```
X = 4; Y = 5; Z = 6;
Team_2 = [X, Y, Z];

all_permutations = perms(Team_2);
% all_permutations(i, 1) plays against A
% all_permutations(i, 2) plays against B
% all_permutations(i, 3) plays against C

% Arrays to store who played A,B, and C.
A_opponent = [];
B_opponent = [];
C_opponent = [];

% Go through all permutations of Team 2.
for i = 1:size(all_permutations, 1)

    % Who A can play.
    % A cannot play against X
    if (all_permutations(i, A) ~= X)
        % If the player hasnt been added yet, add them to A's list.
        if (ismember(all_permutations(i, A), A_opponent) == 0)
            A_opponent(end+1) = all_permutations(i, A);
        end
    end

    % Who B can play.
    % There is no restriction on who B can play.
    % If the player hasnt been added yet, add them to B's list.
    if (ismember(all_permutations(i, B), B_opponent) == 0)
            B_opponent(end+1) = all_permutations(i, B);
    end

    % Who C can play.
    % C cannot play against X and Z
    if ((all_permutations(i, C) ~= X) && (all_permutations(i, C) ~=
 Z))
        % If the player hasnt been added yet, add them to C's list.
        if (ismember(all_permutations(i, C), C_opponent) == 0)
            C_opponent(end+1) = all_permutations(i, C);
        end
    end
end

% Format and print the arrays to the console.
fmt = ['\nA''s opponents:' repmat(' %d',1, numel(A_opponent))];
fprintf(fmt, A_opponent);
fmt = ['\nB''s opponents:' repmat(' %d',1, numel(B_opponent))];
fprintf(fmt, B_opponent);
fmt = ['\nC''s opponents:' repmat(' %d',1, numel(C_opponent))];
fprintf(fmt, C_opponent);
fprintf('\n');
```

```
A's opponents: 6 5
```

```
B's opponents: 5 4 6
C's opponents: 5
```

Published with MATLAB® R2020a