Week 8 - Programming III

- Today:
  - Another loop option
  - A programming example: tic-tac-toe

- Textbook chapter 7, pages 187-190, 193-201 (sections 7.4.2, 7.7)

Longer Running Loops

For loops repeat a fixed number of times:

```
for variable = {array of length n}
  {commands}
end
```

and `break` can be used to stop earlier.

**Question:** How about repeating “until done”? Run as long as is needed.

**Answer:** MATLAB’s “while” loop:

```
while expression
  {commands to be repeated as long as expression is true}
end
```

Prior example – compounded interest until the amount doubles:

```
value = 1000;
for year = 1:1000
  value = value * 1.08;
  disp([num2str(year),' years: $ ',num2str(value)])
  if value > 2000
    break
  end
end
```

**Expected output:**

1 year: $ 1080.00
2 years: $ 1166.40
3 years: $ 1259.71
4 years: $ 1350.49
5 years: $ 1448.33
6 years: $ 1556.07
7 years: $ 1663.82
8 years: $ 1771.83
9 years: $ 1880.00
10 years: $ 1989.32

**while version**

```
format bank
value = 1000;
while value < 2000
  value = value * 1.08;
  disp(value)
end
```
Example – Collecting and storing data until a zero is entered:

```matlab
x = []; % empty array
new = 1; % initialize
while new ~= 0
    new = input('enter value ');
    x = [x, new];
end
x = x(1:end-1) % to drop the zero
```

Example – Getting valid keyboard input:

E.g. forcing the user’s input to be between 0 and 10:

```matlab
x = -3;
while (x < 0) || (x > 10)
    x = input('type a value ');
end
```

or:

```matlab
x = input('enter value ');
while (x<0) || (x>10)
    disp('invalid choice');
    x = input('enter value ');
end
```

Example – computing pi:

```
π = 4 - \frac{4}{3} - \frac{4}{5} - \frac{4}{7} - \frac{4}{9} - \frac{4}{11} - \frac{4}{13} - ...
```

```matlab
% while loop to compute pi
p = 0;
den = 1;
sign = 1;
while abs(p-pi) > 0.0001
    p = p + 4*sign/den;
    sign = -sign;
    den = den + 2;
end
```

Example – “infinite” Hi-Lo:

```matlab
numb = round (10*rand(1));
done = 0;
while ~done
    guess = input('guess');
    if guess == numb
        disp('You got it !!!');
    elseif guess > numb
        disp('too high');
    else
        disp('too low');
    end
    done = 1;
end
```

Nesting of while loops

```matlab
while expression1
    (outer loop commands)
    while expression2
        (inner loop commands)
    end
    (more outer loop commands)
end
```

these can also be more than 2 levels deep
Example of Programming: tic-tac-toe

- Play tic-tac-toe, human against human:
  - Track moves
  - Show board
  - Recognize end of game

Program outline:

```matlab
board = zeros(3,3);
plot(.........)
% get O move
% update board and plot
for turn = 1:4
  % get O move
  % update board and plot
  % check for victory by O
  %
  % get X move
  % update board and plot
  % check for victory by X
end
% check for draw
```

Program flow:

1. Initialization, including graphics
2. Get X’s first move
3. Loop 4 times:
   - Get O’s move, check for win
   - Get X’s move, check for win
   - Break on victory
4. Check for draw

Flowchart:

- Initialization
- Get X move
  - Get O move
    - O wins?
    - No
      - Get X move
      - Game Over
  - X wins?
    - Yes
      - Game Over
    - No
      - Draw?
      - Yes
        - Game Over
      - No
        - Game Over

Program Details: Initialization

```matlab
% tic-tac-toe PFS March 2007

% initialization
clear
close all
board = zeros(3,3);
```
Board Graphic

```matlab
% draw a board
figure
plot([-1.5 1.5],[-1.5 1.5],'k','linewidth',2);
hold on
plot([1.5 3.5],[-1.5 1.5],'k','linewidth',2);
plot([1.5 3.5],[1.5 1.5],'k','linewidth',2);
plot([2.5 3.5],[-1.5 1.5],'k','linewidth',2);
hold off
axis off
```

Get and Show First Move (X)

```matlab
% get first move
move = input('enter player 1 move [r,c] ');
r = move(1);
c = move(2);
board(r,c) = 1;
text(c,-r,'X','HorizontalAlignment','center','FontSize',20)
```

Start Loop with the Second Player

```matlab
% start loop with second player
for turn = 1:4
    % player 2 move
    move = input('enter player 2 move [r,c] ');
r = move(1);
c = move(2);
board(r,c) = -1;
text(c,-r,'O','HorizontalAlignment','center','FontSize',20)
end
```

Check for Victory by O

```matlab
% check for victory by O
result = [sum(board), sum(board')];
result(1) = board(1,1) + board(2,2) + board(3,3);
result(2) = board(1,3) + board(2,2) + board(3,1);
if any(result == -2)
disp('player 2 wins!')
break
end
```

Finish Loop with the First Player

```matlab
% repeat for player 1
move = input('enter player 1 move [r,c] ');
r = move(1);
c = move(2);
board(r,c) = 1;
text(c,-r,'O','HorizontalAlignment','center','FontSize',20)
% check for victory by X
result = [sum(board), sum(board')];
result(1) = board(1,1) + board(2,2) + board(3,3);
result(2) = board(1,3) + board(2,2) + board(3,1);
if any(result == 3)
disp('player 1 wins!')
break
end
end
```

Check for a Draw

```matlab
% check for draw
if all(result == 0) && (sum(sum(abs(board))) == 9)
disp('Nobody wins!')
end
```
Typical Output

Command Window

enter player 1 move [r,c] [2,2]
enter player 2 move [r,c] [1,3]
enter player 1 move [r,c] [1,1]
enter player 2 move [r,c] [1,2]
enter player 1 move [r,c] [1,3]