Before we begin

**HW1**  Solutions posted
**HW2**  Due Monday, July 15, 6pm
  **PL**  Prelim Date
Today

- Pseudocode
- Algorithms
  - Prime Sieve
  - Number Guessing
  - Sorting Numbers
- Switch/Case
Definitions

Algorithm

is a step-by-step description of a calculation, like a recipe.
Examples: Prime Sieve, Binary Search, Bubble Sort.

Pseudocode

is a high-level description of a program or algorithm. It can be converted to a program easily.
Primes Function

Question
What are all prime numbers $\leq N$?

Using what we know

```
function p = primes1 (N)
    p = []; % creates an empty array
    for j = 1:N
        if isprime(j) % built-in isprime
            p = [p, j]; % expands the array
        end
    end
end
```
A Better Primes Function

Add knowledge
All prime numbers, except 2, are odd numbers.

Updated code

```matlab
function p = primes2 (N)
    if N>1, p = [2]; else p = []; end
    % check only odd numbers
    for j = 3:2:N
        if isprime(j)
            p = [p, j];
        end
    end
end
```
Measuring Performance

tic/toc

tic starts the timer, toc returns the elapsed time.

Comparing primes functions

```
N = input('Enter N: ');

tic % Start timer
p0 = primes(N); % Call built-in primes
t0 = toc; % Stop timer and store elapsed time

% Let's also measure our functions
tic; p1 = primes1(N); t1 = toc;
tic; p2 = primes2(N); t2 = toc;
```
Why is it slow?

- We check isprime for 3, 5, 7, 9, 11, 13, 15, ...

Current Version

```matlab
function p = primes2 (N)
    if N>1, p = [2]; else p = []; end
    % there are unnecessary checks
    for j = 3:2:N
        if isprime(j)
            p = [p, j];
        end
    end
end
```
Sieve of Eratosthenes

Prime Sieve

- Idea: Eliminate the multiples of a number ahead of time, so that we don't need to check it.

Algorithm

```plaintext
% Create an array X of all 1's of length N
% Set X(1) to 0
% Find position k of next 1 in the X array
% If k is less than or equal to sqrt(N)
%   Set X(2*k), X(3*k), X(4*k) ... to zero
%   Go back to finding k
% Else
%   Find the indices of all 1's in X array
% These indices are prime numbers
```
function p = primes3 (N)
    X = ones(1,N); % An array of N 1's
    X(1) = 0; % 1 is not a prime number
    m = floor(sqrt(N)); % The maximum number upto
                       % which we have to work
    k = 2; % The next available 1 in X array
           % if X(2) exists :)

    while k <= m

        % Set X(2*k) X(3*k) etc to zero
        for j = 2*k:k:N
            X(j) = 0;
        end

        % Find the next 1 in X array
        k = k + 1;
        while X(k) ~= 1
            k = k + 1;
        end

    end

    p = find(X == 1); % Find all indices of elements
                       % which are equal to 1 in X array
end
Too many possibilities

Handling with if/elseif/else/end

```
n = input('enter a digit: ');
if n == 1
    disp('one');
elseif n == 2
    disp('two');
elseif n == 3
    disp('three');
elseif n == 4
    disp('four');
% and many more
else
    disp('something else');
end
```
Switch/Case

switch
branches based on several cases

Usage

```plaintext
switch switch_expr
  case case_expr1
    do_something1
  case case_expr2
    do_something2
  case {case_expr3, case_expr4}
    do_something3
  otherwise
    do_some_otherThing
end
```
Switch/Case - Examples

n = input('enter a digit: ');
switch n
    case 1
        disp('one');
    case 2
        disp('two');
    case 3
        disp('three');
    case 4
        disp('four');

% and many more

    otherwise
        disp('something else');
end
Switch/Case - Examples

```matlab
hall = 'Morrill';

switch hall
    case {'Morrill','Goldwin Smith'}
        disp('Arts Quad');
    case {'Warren','Roberts'}
        disp('Ag Quad');
    case {'Upson','Duffield','Kimball'}
        disp('Engineering Quad');
    otherwise
        disp('Unknown Hall');
end
```
number = fix(10*rand);
guess = input('enter a digit: ');
if number == guess
    disp('that is my number!');
else
    if number > guess
        disp('my number is greater');
    else
        disp('my number is smaller');
    end
end
Number Guessing - 2

numbergame2.m

```matlab
number = fix(10*rand);
guess = -1;

while guess ~= number
    guess = input('enter a digit: ');
    if number == guess
        disp('that is my number!');
    else
        if number > guess
            disp('my number is greater');
        else
            disp('my number is smaller');
        end
    end
end
```
disp('Pick a number between 1−100!');
pause
high = 100; low = 1; trial = 0;

while 1
    guess = floor((high+low)/2);
    trial = trial + 1;
    fprintf('(Trial %d) Is it %d ?', trial, guess);
    response = input('(y -> Yes, d -> Go down, u -> Go up) ','s');
    switch response
        case {'y','Y'}
            disp('Yay!'); break;
        case {'u','U'}
            low = guess;
        case {'d','D'}
            high = guess;
        otherwise
            disp('Please enter ''y'', ''d'' or ''u''');
            response = input('(y -> Yes, d -> Go down, u -> Go up) ','s');
    end
end
function x = bubblesort(x)

disp('================================== INPUT ARRAY ===========================')
fprintf('(Array x) '); disp(x)

disp('================================================================================')

n = length(x);
step = 0;
for i=1:n
  for j=1:n-1
    if x(j) > x(j+1)
      temp = x(j+1);
      x(j+1) = x(j);
      x(j) = temp;
    end
    step = step + 1;
  end
  fprintf('(Step %2d) ',step); disp(x); pause
  fprintf('(Pass %2d) ',i); disp(x); pause
end