for loops

Looping Statements

- Loops are used to repeat actions.
- Conditional Loops
  - while
- Counted Loops
  - for

```
for loopvar = range
    action
end
```

```
for i = 1:6
  fprintf('hello\n');
end
```

```
for i = [ 1 2 3 4 5 6 ]
  fprintf('hello\n');
end
```

```
for i = 1:6; fprintf('hello\n'); end
```

```
for i = 1:6
  disp(i);
end
```

```
for i = [ 1 2 3; 4 5 6]
  disp(i)
end
```

Common use cases of for loops

A) for i= 1:n
- Do something that doesn't care what i is.

B) for i= 1:n
- Do something with i itself.

C) for x= v
- Do something with x

for i= 1:numel(v)
- Do something with v(i)

D)
for i= 1:R
- for j= 1:C
  - Do something that does not care what i and j are.

E)
for i= 1:R
- for j= 1:C
  - Do something with i and j.

F)
[R,C]=size(m)
- for i= 1:R
  - for j= 1:C
    - Do something with m(i,j)

Exercise: runningsum

- Write a function runningsum.m that takes an integer n, and returns the sum of numbers from 1 to n. Do not use the sum() function.

- Change the function runningsum so it can take 2 arguments "start" and "finish" and returns the sum of numbers "start" to "finish".
  - If only "start" is provided, return the sum of numbers from 1 to "start".
  - Important programming concept: specifying default function arguments.
- Change the function runningsum so that the arguments "start" and "finish" can be specified in any order.

Exercise: myprod

- Write a function myprod(v) that returns the product of the elements in v. Do not use the built-in prod() function.

Combining loops with ifs.

- Exercise: Write a function mymin(v) that returns the minimum value in the vector v. Use a for loop. Do not use min() function.

Combining loops with ifs.

- Exercise: Write a function mymax(v) that returns the maximum value in the vector v. Use a for loop. Do not use max() function.

Nested for loops

```
for loopvarone = rangeone   ←   outer loop

    % actionone includes the inner loop

    for loopvartwo = rangetwo     ←     inner loop
        actiontwo
    end
end
```

- Exercise: write a function printrectangle(R,C) that prints a box of stars, width height R and width C.

```
>> printrectangle(3,4)
****

****

****
```

Exercise

- Write a function printtriangle(R) that prints a triangle of height R, with 1 star in the first row, and R stars in the last row. Do not use if statements or matrices or indexing.

```
>> printtriangle(4)
*
**
***
****
```

Exercise

- What will the following code print?

```
for i = 1:3
  for j = 1:2
    fprintf('i=%d, j=%d\n', i, j);
  end
end
```

Exercise: multtable

- Write a function multtable(R,C) that returns a matrix m where m(i,j) is equal to i*j.

Exercise: combining nested loops and if

- Write a function mymatsumifpos(m) that returns the sum of positive elements in the matrix m.

Exercise

- Write a function mymatsum(m) that returns the sum of all elements in the matrix m. Use nested for loops.
- Create a random 10000x10000 matrix m in command window. calculate mymatsum(m). How long does it take matlab to calculate this?
  - Programming concept: tic, toc
- Can you re-write your function to run faster?
  - Programming concept: proximal/linear memory indexing
- Can you re-write your function to contain a single for loop?

Tips for Speed:
"Preallocate" and/or "Avoid loops"

```
a=[];
for i=1:10000
        a(i)=log(i);
end
```

```
a=zeros(1,10000);
for i=1:10000
        a(i)=log(i);
end
```

```
a=log(1:10000);
```

```
tic; a=[]; for i=1:10000; a(i)=log(i); end; toc
tic; a=zeros(1,10000); for i=1:10000; a(i)=log(i); end; toc
tic; a=log(1:10000); toc
```