Chapter 2 Review by ZHANG Zhengyuan

Matlab is

- Interpreted
- Dynamically typed

Control Flow

Logicals

Produced by

- primitives: true, false
- Comparative operators: >, <, ==, \~=, etc
- Casting:
  - explicit: logical()
  - implicit: non-zero values are casted to true, zero is casted to false

Chain logicals

- Comparative operators: &, |, \~, xor(a, b). Both operands would be evaluated.
- Short-circuit operators: &&, ||. Second operand is evaluated only if necessary.

Conditional Branches

if

```
if expression
    statements
elseif expression
    statements
else
    statements
end
```

switch

```
switch switch_expression
   case value1
      statements
   case {value2, value3, value4}
      statements
   }
   otherwise
      statements
end
```

Loops

while

```
while expression
    statements
end
```

for

```
hello = 'hello world';
for i = hello
   disp(i)
end
```

additional control

- break: exit the nearest loop
- continue: jump to the next iteration of the nearest loops
- return: exit the current function, if any, or script
- exit: exit the Matlab Program

Data Structure: Matrix

- The default data structure in Matlab is matrix
- Cannot contain elements with different types

- The matrices are stored as array in memory
    - Construct:

```
a = 5                  % a is a 1*1 matrix
b = [1 2 3; 4 5 6]     % b is a 2*3 matrix
c = 1 : 6              % c is a 1*6 matrix
d = linspace(1, 6, 6)  % d is identical to c

% construct using other matrix
c'                     % [1; 2; 3; 4; 5; 6]
c + 1                  % [2 3 4 5 6 7]
c.^2                   % [1 4 9 16 25 36]
[[a; 6] [a; 7]-5]      % [5 0; 6 2]
reshape(b, 3, 2)       % [1 5; 4 3; 2 6]
% etc
```

    - Visit:

```
% single access
b(4)                   % display 5
b(2, 3)                % equivalent to b(2 + (3-1)*size(b,1)), display 6

% block access
b(2, :)                % return the 2nd row
b(:, [1 3])            % return a matrix containing 1st & 3rd column

% errors
b(3, 3)                % error: index exceeds array bound
b(0, 2)                % error: index is invalid. Array indices must be positive integers or logical values
```

    - Modify:

```
% change elements: similar to visit
b(:, 2:end) = [7 8; 9 10]   % b becomes [1 7 8; 4 9 10]
c(:) = 5                    % c becomes [5 5 5 5 5 5]

% extend matrix
b(7) = 7                    % error: Attempt to grow array along ambiguous dimension
b(3, 2) = 8                 % No error, b becomes [1 7 8; 4 9 10; 0 8 0]

% shink matrix
b(2, :) = []                % delete 2nd row, b becomes [1 7 8; 0 8 0]
b(4) = []                   % b is flattened to [1 0 7 8 0]
```

    - Get information about a matrix

```
size(b)
sum(b)
```

```
numel(b)
b > 5                % return a logical array
b(b > 5) = 0             % change the elements that are greater than 5 to 0
% etc
```