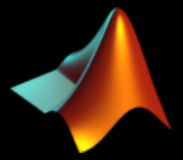

Programming with MATLAB

While-loops

Dimitris Voudouris, PhD
d.s.voudouris@gmail.com

while-loops

Why use *while*?

% To execute a statement for an un-defined number of times
% e.g.,
% ***while*** subject holds down the button
% …present XY on the monitor
%
% ***while*** the computed variable is larger than X
% … keep on calculating
%
% For-loops run for a fixed number of iterations
% While-loops *can* run for an undetermined number of iterations

while-loops

Structure of loop

```
while logical expression
    statement
end

% A logical expression can be true or false
% While it is true…
% …the statement is executed
% The loop always closes with an end
```

while-loops

With a *while-loop* you can repeat an operation several times

```
cnt = 1;
while cnt = 1:10
    disp(cnt)
    cnt = cnt + 1;
end

% When using a while-loop, the counter-variable (here cnt)
% may need to be defined beforehand.
% If you want your loop to proceed to the next iteration, you
% need to update the variable (cnt = cnt + 1)

% In contrast, when running a for-loop, you define the variable at the first
% line and this variable is updated automatically
```

break

You can exit a *while* loop using the command *break*

```
target = 0.9;
iter = 1;
while 1                 % Matlab understands '1' as true (and '0' as false)
    temp = rand;
    if temp >= target;
            break    % If variable temp is equal or larger than 0.9, terminate
    end
end

% The function break terminates the current while-loop. If the while-loop is
% nested, then Matlab will return control to the outer loop.
% break can also be used in for-loops
```

continue

You can skip an iteration using the command *continue*

```
trial = 1;
while trial < 181
    MT(trial) = randi([500 700], 1, 1);
    if MT(trial) <= 530
            trial = trial + 1;
              continue            % If MT(trial) is <= 530, skip iteration
    end
    amplitude(trial) = randi([25 35], 1, 1); % executed only if MT > 530
    trial = trial + 1;
end

% Once MT(trial) is <= 530, amplitude won't be calculated,
% so amplitude(trial) will be assigned a zero!
% Don't forget that Matlab fills up skipped indices with zeros!!
```

Have fun!

Do not forget ***keyboard***

It can help you :
- Understand why your loops behave oddly
- Build a (seemingly) complicated loop by seeing its state after every iteration