MATLAB for Finance

10
Language Syntax

Programming Languages

- Programming languages generally fall into one of two categories: compiled or interpreted.
- With a compiled language, code you enter is reduced to a set of machine-specific instructions before being saved as an executable file.
- With interpreted languages, the code is saved in the same format that you entered.

Advantages and Disadvantages

- Compiled programs generally **run faster** than interpreted ones, because interpreted programs must be reduced to machine instruction at runtime (i.e. during the execution).
- However, with an interpreted language you can do things that cannot be done in a compiled language.
  - For example, interpreted programs can modify themselves by adding or changing functions at runtime.
  - It is also usually easier to develop application in an interpreted environment because you don't have to recompile your application each time you want to test a small section.
- Even though interpreted programs are readable (by any other) and must be translated in machine language during any execution

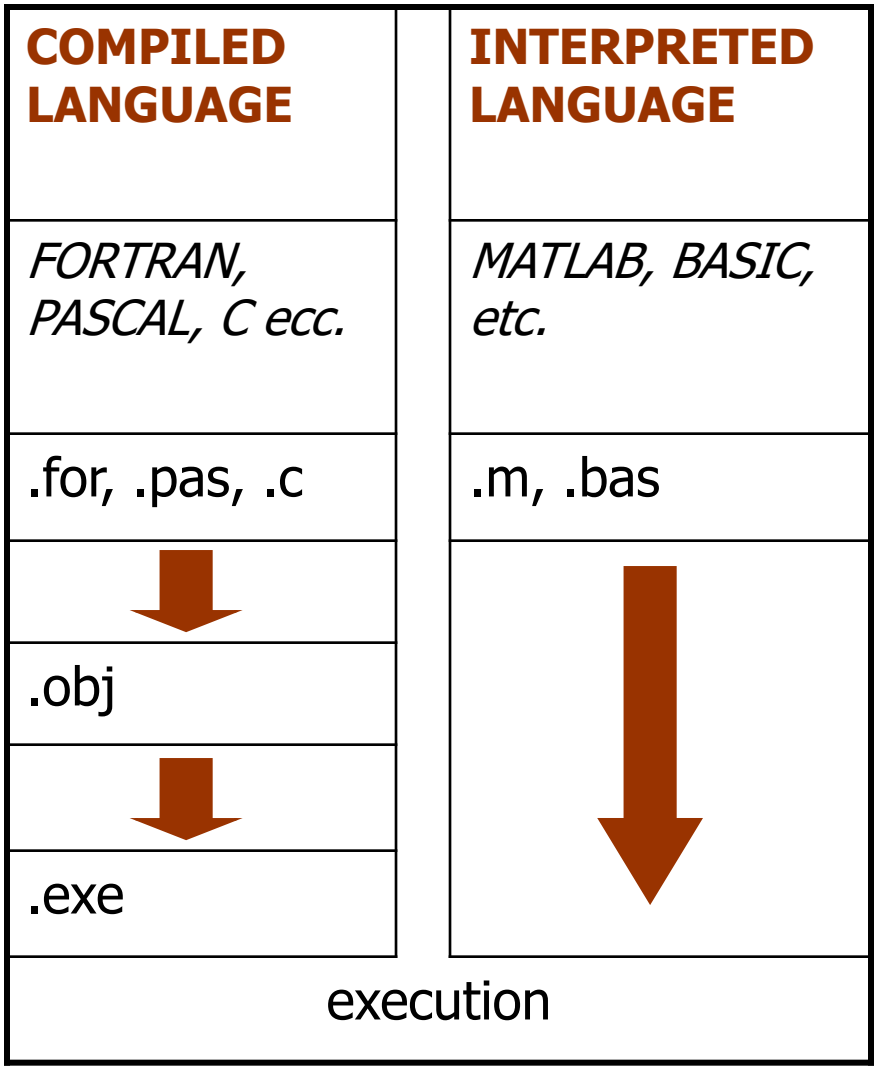
edit in execution

copyright

Compiled vs. Interpreted

- Problem
  - ↓Analysis
- Algorithm
  - ↓Implementation
- Source Code (program)
  - ↓Compiling (+ linking)
- Executable (program)

  Execution (running)



Exercise

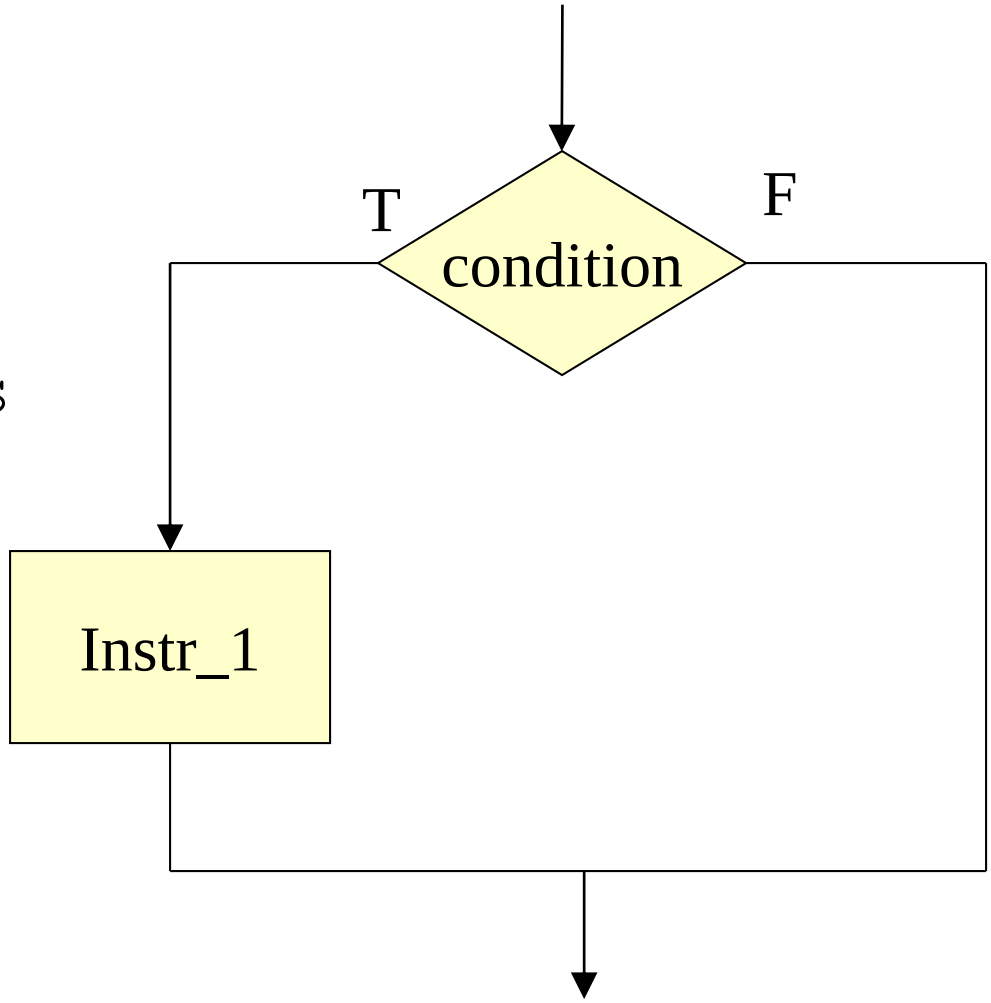
- Create function f that makes the following computation:

$$f(x) = \begin{cases} x^2 + 2 & \text{if } x > 0 \\ -3x & \text{otherwise} \end{cases}$$

Selection (one way)

- If ... Then

```
if condition
    instructions
end
```



Exercise

- Create a function that divides *a* by *b* only if $b \neq 0$

Condition

item `logical_operator` *item*

constant or variable

- Operator:
  - `>` greater than
  - `<` less than
  - `>=` greater than or equal to
  - `<=` less than or equal to
  - `==` equal to
  - `~=` not equal
  - `~` not (unary operator)

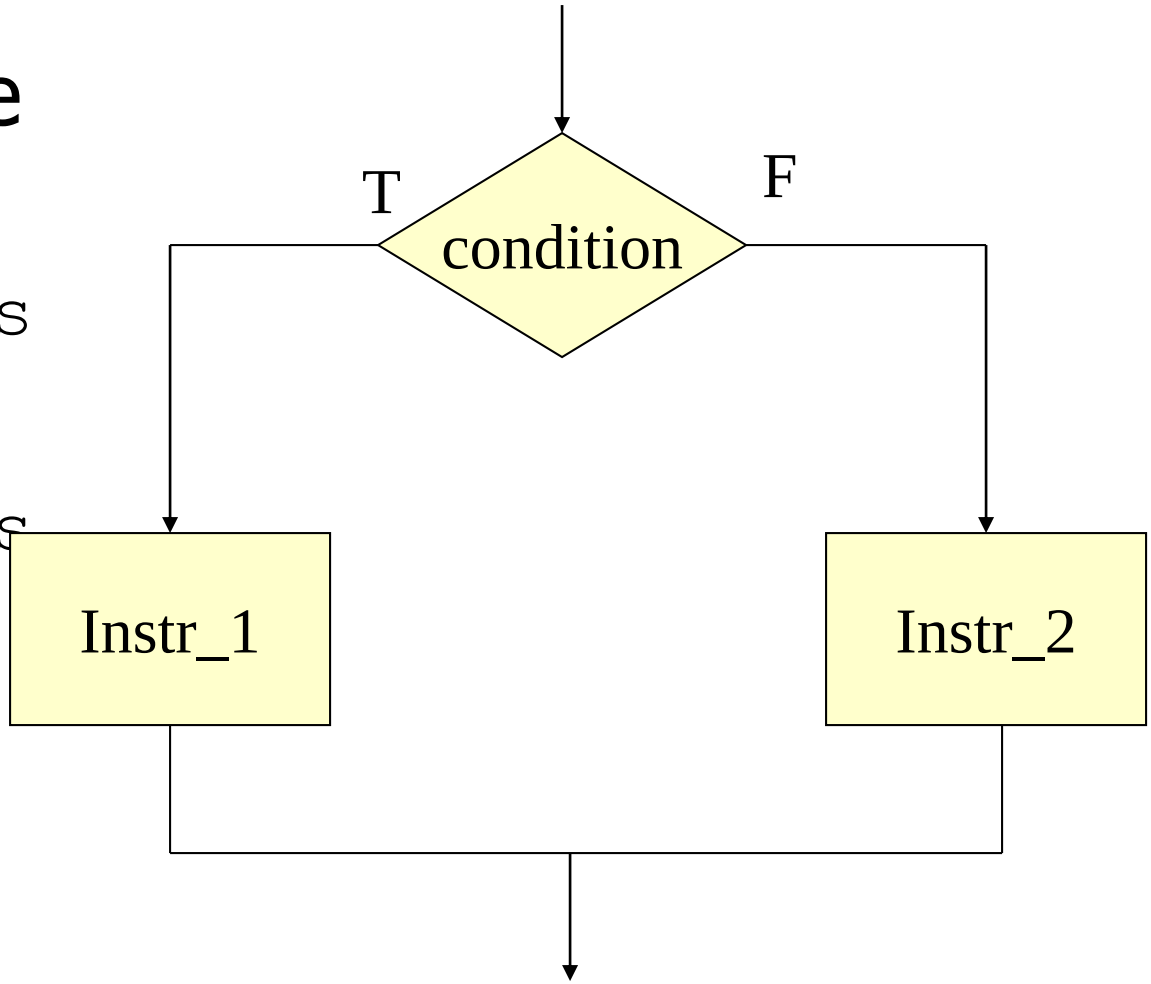
any()

all()

Selection *(two ways)*

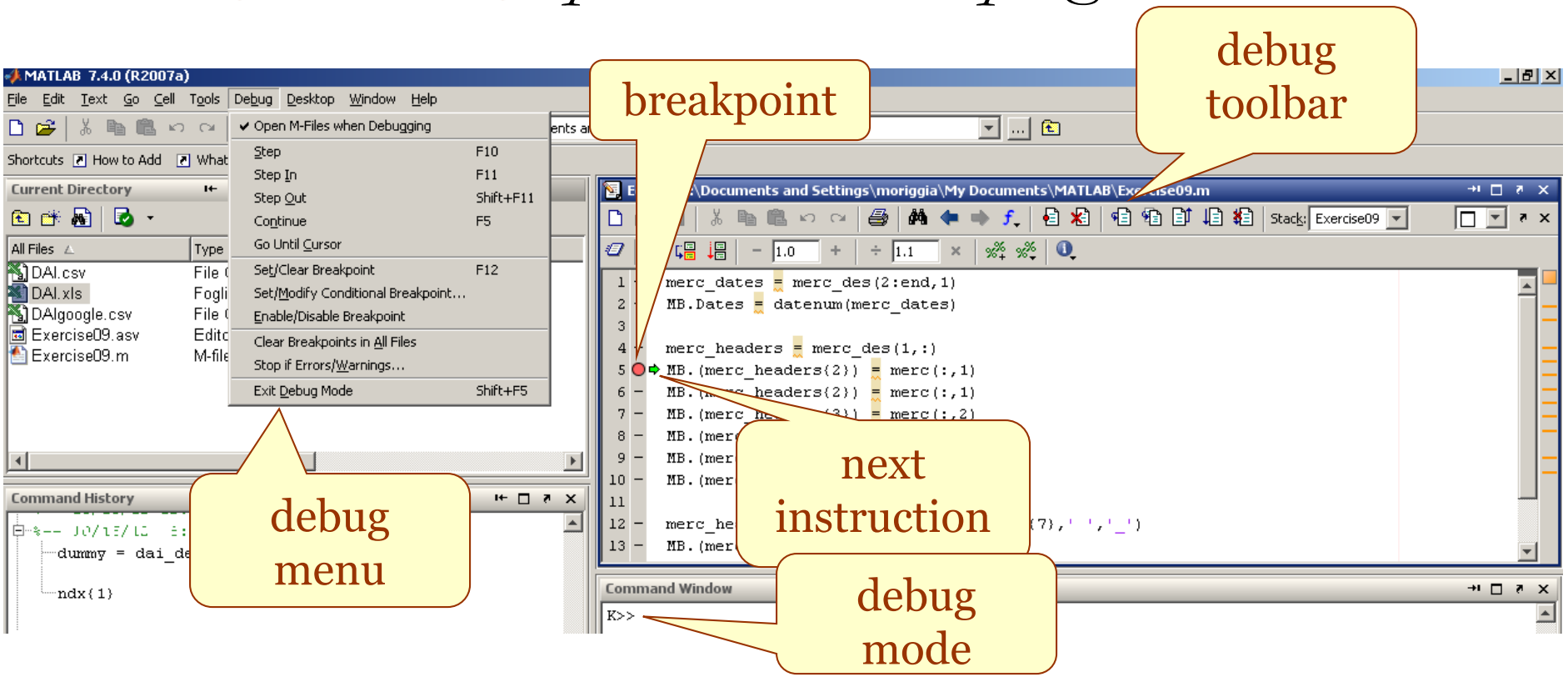
- If… Then… Else

```
if condition
  instructions
else
  instructions
end
```



Debugger

- *Program used to test and "debug" other programs*
- **Debugging:** *process finding 'bugs' in a prgm*
- **Program bug:** *problem with a program*



for

- Loop for represents the mathematic symbol ∀
- The syntax is:

```
for counter = initial:last
    instructions
end
```

- Find the solutions of a quadratic equation

- lookfor spline
- help spline

Polygonal Chain

- "A connected series of line segments" (*)
- The equation of the line passing through two different points

*) Wikipedia

Bond Pricing

With real data

- Download from web
  - EUR yield curve
  - a BTP
- create the bond cashflow vector
- create the bond ttm vector
- match the bond ttm to the yield curve ttm
- compute the fair price

Linear Interpolation

Exercise

- Generalize the first M-file of MB structure exercise for using only one statement for all the repeated statements:

```
MB.(headers{2}) = merc(:,1)
```

- Can we put in loop also

```
headers{7} = strrep(headers{7},' ','_')
```

or not?