Computer Algebra and Technical Computing (MTH1006)

B. Vorselaars
bvorselaars@lincoln.ac.uk

School of Mathematics and Physics, University of Lincoln

Module structure

- Contents:
  - MATLAB (Technical Computing): 1st term
  - MAPLE (Computer Algebra): 2nd term.
- Total credits 15: on average 150 hours in total. This implies on average per week:
  - 2 hours lectures/practicals
  - 4 hours 15 min time outside scheduled hours.
- Grading
  - 20% logbook
  - 80% coursework (one assignment, one in-class test per term).

Any issues

If there are any issues:

- Contact the lecturer (me): `bvorselaars@lincoln.ac.uk`
- Or contact your course rep

Marks MATLAB logbook and in-class test

The marks for the MATLAB logbook and the MATLAB in-class test can be found online, including feedback.

Assessment dates - MAPLE

- Assessment dates
  - 28-2-2024: coursework handed-out
  - 13-3-2024: hand-in coursework
  - 24-4-2024: hand-in logbook
  - 1-5-2024: Final in-class test
- They can also be found on blackboard (heading Assessments)
- Add them to your diary/calendar!

Aim module part 2: Computer Algebra with MAPLE

Usage of MAPLE (outline syllabus):

- Numerical calculations, built in functions & constants, basic algebraic operations, naming and evaluating expressions, defining and evaluating functions
- Plotting graphs (Cartesian, polar and parametric, 3D, contour, surfaces)
- Differentiation, 1st and 2nd derivatives, integration, approximating integrals, area, solids of revolution, partial differentiation.
- Polynomial arithmetic, manipulating polynomials, factorisation of polynomials
- Limits, sequence and series, convergence tests, Taylor & Maclaurin series
- Simple differential equations
- Matrices, matrix operations, inverse, determinant. Linear systems of equations, row operations, eigenvalues and eigenvectors

MAPLE

What is MAPLE used for?

- Symbolic manipulations (integration, differentiation, ...), linear algebra (Matrix calculations)
- Numerical mathematics and computation
- Programming
- Plotting and processing data

More information on MAPLE

- E-book: Maple by example - Abell and Braselton, https://library.lincoln.ac.uk/items/135577?query=Abell+and+Braselton&resultsUri=items%3Fquery%3D%2BAbell%2Band%2BBraselton%26target%3Dcatalogue%26facet%255B0%255D%3Dfulltext%253Ayes&facet%5B0%5D=fulltext%3Ayes&target=catalogue (unlimited number of e-books)
- User manual: https://www.maplesoft.com/documentation_center/maple2020/UserManual.pdf
- Programming guide: https://www.maplesoft.com/documentation_center/maple2020/ProgrammingGuide.pdf
- Various tutorials within MAPLE (see exercises)
- Googling

MAPLE alternatives

A license is needed. The university has a license for on-campus and online off-campus. There is also a student version available to buy. No entirely compatible alternatives as with MATLAB. Programs with similar functionality:

- Mathematica: also not free. Very powerful, slightly less intuitive. See also `http://www.wolframalpha.com` and `http://sandbox.open.wolframcloud.com/` for online versions
- MuPAD: another computer algebra system; part of MATLAB since 2008.
- REDUCE: another free computer algebra system, started in 1963 (`http://reduce-algebra.sourceforge.net/`)
- SymPy: a free computer algebra library, used within Python (`http://www.sympy.org`). Much weaker than Maple, in terms of capabilities, but basic functionality is available.

MAPLE alternatives

The previous list is not complete, and there are other alternatives (e.g., the free Maxima (`http://maxima.sourceforge.net`) or Xcas (`https://www-fourier.ujf-grenoble.fr/~parisse/giac.html`)), and many more online tools. However, most of the times they are not as powerful as MAPLE.

Accessing MAPLE

The full version of MAPLE can be accessed:

- In the computer lab INB2305. This is open 24/7 and you can enter outside of lecture hours, and if the lecturer allows it also inside lectures of other modules.
- From home using the Cloud Desktop. For this, go to `https://client.wvd.microsoft.com/arm/webclient/index.html` and log in. See `https://ict.lincoln.ac.uk/?p=17583` for connection instructions. See Blackboard -> module content -> Accessing Matlab and Maple for more detailed information.
- In the library (24/7 open), on the 1st floor: some of the Windows workspaces 22–25 next to room UL108 have Maple installed. See `https://findapc.lincoln.ac.uk/?building=ul&tab=first` for the layout and availability

If any of the PCs do not work, send an email to Digital Technologies (`ict@lincoln.ac.uk`) and cc me in.

Accessing limited versions of MAPLE

Some functionality of MAPLE is for free:

- The freely available MAPLE player can be installed on a home computer, for *reading* MAPLE documents and worksheets. See `https://www.maplesoft.com/products/maple/mapleplayer`.
- Some simple commands can be entered online. See `https://www.maplesoft.com/products/StudentApps/index.aspx`. In particular, see `https://maplecloud.maplesoft.com/application.jsp?appId=16155001`. Simple commands can be entered and some of them may be executable by this web interface.

Accessing full versions of MAPLE

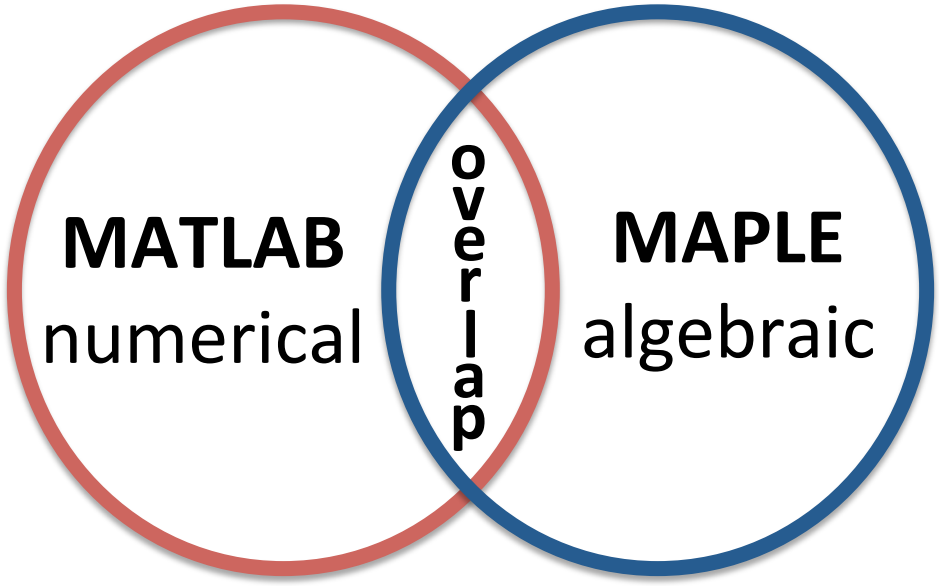
In case you can't be on the university *and* your internet connection is really bad

- You can download a student trial for Maple. This trial is valid for approximately 2 weeks, and if needed adviced to do this *only two weeks before the test* so that you can complete the logbook and still practise for the in-classs test. See `https://www.maplesoft.com/products/maple/free-trial/`

Comparison with MATLAB

Following slides show a comparison between the two packages. MAPLE commands presented here will be discussed in more detail later in the module.

Overlap in packages



Script for n! (normal factorial)

Maple

```
n:=5:
product(k, k = 1 .. n)
```

MATLAB

```
n=5;
prod(1:n)
```

Function for n! (normal factorial)

Maple

```
my_factorial := proc (n)
product(k, k = 1 .. n)
end proc
```

MATLAB

```
function fac=my_factorial(n)
fac=prod(1:n);
end
```

Function for n!! (double factorial)

Maple

```
my_double_factorial := proc (n)
 local dfac, k;
 dfac := 1;
 for k from n by -2 to 1 do
  dfac := dfac*k
 end do;
end proc;
```

MATLAB

```
function dfac=my_double_factorial(n)
dfac=1;
 for k=n:-2:1
  dfac=dfac*k;
 end
end
```

Calling the functions

Maple

my_factorial(5)

my_factorial(6)

my_double_factorial(5)

my_double_factorial(6)

MATLAB

my_factorial(5)

my_factorial(6)

my_double_factorial(5)

my_double_factorial(6)

Testing the factorial function for large values

Maple

factorial(100)

933262154439441526816992388562667004907159682643816214685929638952175999932299156089414639761565182862536979208272237582511852109168640000000000000000000000000

MATLAB

>> factorial(100)

ans =

9.3326e+157

MATLAB: approximate solution (fits within a floating point number)
Maple: exact solution

Testing the factorial function for large values

Maple

factorial(200)

```
788657867364790503552363213932
185062295135977687173263294742
533244359449963403342920304284
011984623904177212138919638830
257642790242637105061926624952
829931113462857270763317237396
988943922445621451664240254033
291864131227428294853277524242
407573903240321257405579568660
226031904170324062351700858796
178922227896237038973747200000
000000000000000000000000000000
000000000000000
```

MATLAB

```
>> factorial(200)

ans =

   Inf
```

MATLAB: ‘overflow’ of number (does not fit within a floating point number)

Maple: exact solution

Exercise with double loop

Maple

```
S1 := proc (N)
 local S, d, n, m;
 S := 0;
 for n to N do
  for m to N do
   if n = m then
    d:=1
   else
    d:=0
   end if;
   S := S+d*n
  end do;
 end do;
end proc;
```

MATLAB

```
function S=S1(N);
S=0;
for n=1:N
    for m=1:N
        if n==m
            d=1;
        else
            d=0;
        end
        S=S+d*n;
    end
end
```

Running S1(2000) takes 0.055 seconds with MATLAB and 4.4 seconds with Maple.

→**MATLAB is 80 times quicker**

Comparison

Maple

Really good with symbolic calculations: exact

MATLAB

Really good with numerical calculations: fast

Comparison of Maple and Matlab syntax

	Maple	Matlab
Addition	`+`	`+`
Exponentiation	`^`	`^`
Suppress output	`:`	`;`
π (the number 3.141...)	`Pi`	`pi`
i (for which $i^2 = -1$)	`I`	`i`
$n!$ (factorial)	$n!$ or `factorial(n)`	`factorial(n)`
$n!!$ (double factorial)	$n!!$ or `doublefactorial(n)`	N/A
help on function $\log(x)$	`? log`	`help log`
Boolean NOT	`not`	$\sim$
Boolean OR	`or`	`|`
Boolean AND	`and`	`&`
Clear all variables	`restart`	`clear`
Clear the variable a	`a:='a'`	`clear a`
Set the variable $a2$ equal to 5	`a2:=5`	`a2=5`
Set the variable v to a column vector	$v := \langle 1, 2, 3 \rangle$	`v=[1; 2; 3]`
Row vector	$\langle 1 \vert 2 \vert 3 \rangle$	`[1, 2, 3]`
Matrix	$\langle \langle 1 \vert 2 \vert 3 \rangle, \langle 4, 5, 6 \rangle \rangle$ or $\langle \langle 1, 4 \rangle \vert \langle 2, 5 \rangle \vert \langle 3, 6 \rangle \rangle$	`[1, 2, 3; 4, 5, 6]`
Matrix multiplication	`A.B`	`A*B`
Element-wise multiplication	`A*~B`	`A.*B`
Transpose of a matrix A	A^+	`A.'`
Store user input in b	`b:=readstat("Give a number")`	`b=input('Give a number')`
Access 2nd row, 3rd column of matrix A	`A[2,3]`	`A(2,3)`

Access 2nd till 3rd row, 1st and second column of matrix A	`A[2..3,1..2]`	`A(2:3,1:2)`
Access 2nd till end row, 1st and second column of matrix A	`A[2..,1..2]` or `A[2..-1,1..2]`	`A(2:end,1:2)`

Maple code of a previous line can be converted to Matlab within Maple by

```
CodeGeneration[Matlab](%)
```

Note on Maple vs Matlab

Maple can do numerical calculations, and Matlab can do symbolic calculations. However, Matlab is better in numerical and Maple in symbolic.

Two modes

- Worksheet mode (basic)
- Document mode: similar to a word document, but then with active elements such as equations. Could be used as a logbook.

Basics

- Palette: hover over symbol to see the MAPLE command.
- keyboard shortcut CTRL-L: to refer to another equation.
- Imaginary number i: `I` (notice: capital I)
- The number $\pi = 3.141\ldots$: `Pi` (notice: capital P)
- Differentiate an expression: `diff`, e.g. `diff(x^2,x)`
- Integrate an expression: `int`, e.g., `int(x^2,x=a..b)`
- Obtain numerical value: `evalf`, e.g., `evalf(Pi)`
- Difference light-blue background (Math mode) and white background (text mode)
- Context Panel

Exercise/homework for this session

- Open the *Getting Started*, then *Tutorial: Talking to Maple* tutorial and complete this. To do this, open a new blank document by clicking on the menu bar *File*$\rightarrow$ *New* $\rightarrow$ *Document Mode*. Store your `.mw` file on your OneDrive for your logbook.
- Do the same for the second tutorial, *Tutorial: Putting Ideas Together*.