2 More about MATLAB

2.1 Arithmetic

MATLAB has the expected arithmetic operators including \(^\) for exponentiation. Precedence is as expected. There are also inbuilt constants such as \(\pi\), and standard functions such as \(\sin\), \(\ln\), \(\exp\), and so on. The trigonometric functions and their inverses use radians.

Some calculations can produce two special values: \(\text{Inf}\) (standing for infinity) can be produced by for example \(1/0\); while \(\text{NaN}\) (standing for Not-a-Number) can be produced by for example \(0/0\).

2.2 Input and Output

Strings are enclosed in single quotes: \(A = ‘Hello world’\). (Well technically this is a character vector and there is another version of strings—more on this later.) The user can be prompted for input using the \(\text{input}\) command; e.g.

\[
\text{>> tol = input(’Enter tolerance ’);}
\]

One can use the command \(\text{format}\) to adjust the display. For example

\[
\text{>> format long;}
\text{>> 2*acos(0)}
\]

will print out \(\pi\) to 15 decimal places. In general one can use the command \(\text{disp}\) to display messages to the user. A more general output command is \(\text{fprintf}\). This command is given a “format-string” where the percent-sign is used to indicate a slot to be filled by a value, followed by the values in order. There are many options. In the following example the \(%.4f\) slot prints out its contents as decimal to 4 decimal places (f for float); the \(\backslash n\) part causes a new-line.

\[
\text{>> fprintf( ’The square-root of %.4f is %.4f\n’, X, sqrt(X) );}
\]
2.3 More User Stuff

It is good programming practice to lay out the code using indentation and white space to help the reader. But note that the command executor does not care.

One can use the \texttt{help} command to get some information about a command. One can use \texttt{clear} to forget calculations. In the Command Window, the up-arrow accesses previous commands. There are extra commands in \texttt{toolboxes}, which we do not use at present.

\begin{verbatim}
Sample code

Here is code to convert temperatures.

\% script: farenheit
\% goddard 2021
\n\texttt{disp(‘This will convert from Farenheit to Celsius’)};
\texttt{F = input( ‘Input Farenheit temp (whole number): ’)};
\texttt{C = (F-32)*5/9;}
\texttt{fprintf(‘\%i in Farenheit is \%i in Celsius\n’, F, round(C) )};
\end{verbatim}