**Chapter 11 Short Summary**

\* *Imaginary* unit of *Complex Numbers*

-> Write a complex number as usual (i.e. ).

-> The imaginary unit is any of the special symbols  in the Common Symbols palette or I on the keyboard. By default Maple always outputs the imaginary unit as .

-> To have Maple output the imaginary unit as and accept  on the keyboard as input execute

**> **



(In below examples since the  function above has been evaluated, Maple will always display the imaginary unit as .)

\* Fundamental *Complex Number Functions*

->  =  and  = 

(They can be input as  and  using the Common Symbols palette, e.g.  = ,  = .)

->  = 

(Can be input as  using the Accents palette, e.g.  = )

->  = 

(Can be input as  using the Layout palette or the keyboard, e.g.  = .)

(Note that )

->  = 

(returns value in radians, e.g.  = ).

-> 

returns a complex number  in modulus-argument form as which represents  where e.g.  = .

 can also be used as input as an alternative to .

\* *Simplifying* Complex Numbers and *evalc()*

To obtain an explicitly numerical result of a complex number, you can use numerical approximation, by inputting decimal points or using evalf().

e.g.

**> **



**> **



-> (To manipulate complex expressions containing indeterminates you must decide whether the indeterminates represent real or complex values.) By default, Maple assumes that indeterminates represent complex values. This leads to results that are not very useful, e.g.

**> **

**> **



**> **



**> **



An advantage of representing a complex number as  rather than  is that the arguments of the  function are assumed to be real, although  is not assumed to be positive, e.g.

**> **



**> **



**> **



**> **



**> **



-> One way to tell Maple to regard indeterminates as real is to apply the Maple complex evaluation function *evalc()*, which regards all indeterminates as real, e.g.

**> **



**> **



**> **



**> **



**> **



(Note that *evalc()* also attempts to split a complex expression into its real and imaginary parts whenever possible.)

e.g. Euler's formula can be constructed as follows:

**> **



-> Remark: Maple knows **** = **** and automatically simplifies complex number expressions so that the imaginary unit appears at most once and never in a denominator.

\* *complexplot()*

The function  in the  package is essentially a complex analogue of the  function. It accepts a list of complex numbers as its first argument and plots them on an Argand diagram. Like , it accepts ranges for the real and imaginary axes as its second and third arguments respectively, and by default it joins the points with straight lines.

e.g.

**> **

**> **

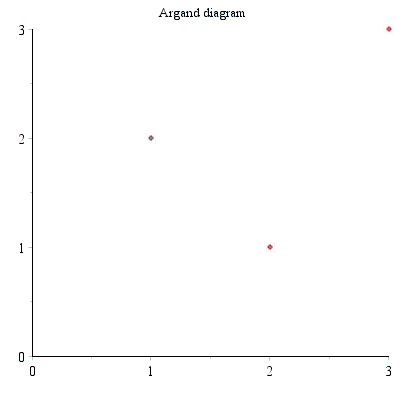






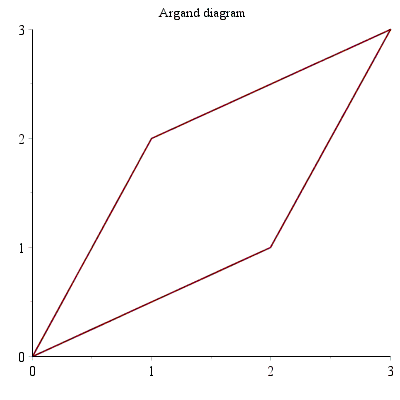
**> **

**> **



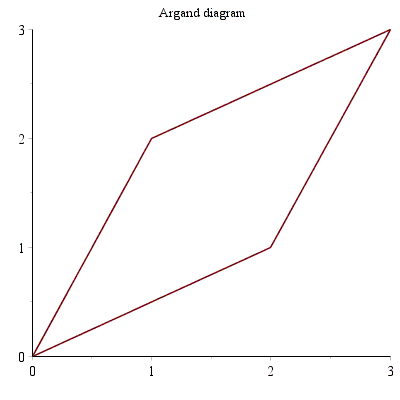
Remark: However, unlike ,  does not accept multiple input data, so to plot the parallelogram representing this complex sum we can either plot it as a polygon:

**> **



or we can use  to combine two complex plots, (which is more natural and flexible). This produces an identical plot to that above:

**> **

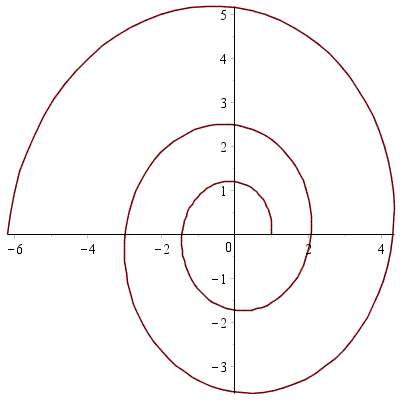


-> *complexplot()* allows you to plot functions from ℝ to ℂ (as curves in the Argand diagram). E.g.

**> **



**> **



\* Aside: *De Moivre’s Thoerem*

**> **

**> **



**> **

 **(1.3.2)**

Example: (Constructing formulae for  and for a value of n)

**> **



Expanding only the right-hand side of this equation gives:

**> **

**(1.3.3)**

Taking the real and imaginary parts give:

**> **



**> **



Now simplifying using  using a "side relation" (see the Maple help for details):

**> **



**> **

