Gaining Experience, Abstraction

We continue where we left off in the previous chapter. The excitement is rising. After doing some preparations (in connection with algebraic expressions) we are going to use Maple in whole induction proofs. The exercises solved first start like this: “Prove that...” In these cases the task itself contains the statement to be proved.

In the two tasks of the worksheet 3.3 the statement to be proved is also unknown so it is our task to define the conjectures. And we can define conjectures by gaining experiences. And since people are originally lazy, they strive for creating witty solutions with which they can gain maximum experience with minimal input. We hope that after reading this chapter the number of those people who agree that Maple is one of the most remarkable tools of humankind will rise.

3.1 A Complex Number Exercise

Calculate the value of the following expression:

$$\frac{1 + \mathrm{I}\tan(\mathrm{alpha})}{1 - \mathrm{I}\tan(\mathrm{alpha})}$$

where I is an imaginary unit.

As a start it is enough to know that Maple knows the Greek letters. Let’s illustrate it with some examples.

```
> alpha, Beta, epsilon, pi, Omega
```

$$\alpha, \mathrm{B}, \varepsilon, \pi, \Omega \qquad \textbf{(1)}$$

The Greek letters are written in English into the input line. The Greek capital letters are written with capital letters. Naturally the palettes are available as well. From the Greek palette we can insert Greek letters to the place where the cursor stands with one click. And in this case we see the Greek letter itself on the input line instead of the phonetic English transcription.

```
> alpha, Beta, epsilon, phi, Omega
```

$$\alpha, \mathrm{B}, \varepsilon, \phi, \Omega \qquad \textbf{(2)}$$

After this give the following command to enter the expression of the task . The sign of the imaginary unit in Maple is I. You can enter it on the keyboard but it can be found in the Common Symbols palette as well.

$$> \ \frac{1 + \mathrm{I}\tan(\mathrm{alpha})}{1 - \mathrm{I}\tan(\mathrm{alpha})}$$

$$\frac{1 + \mathrm{I}\tan(\alpha)}{1 - \mathrm{I}\tan(\alpha)} \qquad \textbf{(3)}$$

```
> evalc(%)
```

$$\frac{1}{1+\tan(\alpha)^2} - \frac{\tan(\alpha)^2}{1+\tan(\alpha)^2} + \frac{2\,\mathrm{I}\tan(\alpha)}{1+\tan(\alpha)^2} \qquad \textbf{(4)}$$

Maple did note execute the operations of the first command. We had to call the evalc procedure to do this. Do you still remember that the % sign is the result of the previous operation? This evaluates the evalc procedure as a complex expression and gives the canonical form of the complex number as a result.

Let's continue with that we substitute tan(alpha) with $\frac{\sin(\text{alpha})}{\cos(\text{alpha})}$.

> $subs\left(\tan(\text{alpha}) = \frac{\sin(\text{alpha})}{\cos(\text{alpha})}, \%\right)$

$$\frac{1}{1+\frac{\sin(\alpha)^2}{\cos(\alpha)^2}} - \frac{\sin(\alpha)^2}{\cos(\alpha)^2\left(1+\frac{\sin(\alpha)^2}{\cos(\alpha)^2}\right)} + \frac{2\,\mathrm{I}\sin(\alpha)}{\cos(\alpha)\left(1+\frac{\sin(\alpha)^2}{\cos(\alpha)^2}\right)} \qquad (5)$$

> $normal(\%)$

$$-\frac{-\cos(\alpha)^2+\sin(\alpha)^2-2\,\mathrm{I}\sin(\alpha)\cos(\alpha)}{\cos(\alpha)^2+\sin(\alpha)^2} \qquad (6)$$

> $combine(\%, trig)$

$$\cos(2\,\alpha)+\mathrm{I}\sin(2\,\alpha) \qquad (7)$$

The result of the substitution is a pretty ugly expression which the normal procedure simplified. We applied the combine procedure with the trig option for this. Notice that the system knows the trigonometric identities and it executed the possible simplification. We are done with this. Let's continue the calculation of the complex numbers with another task.

Determine the complex numbers the conjugate of which is the cube of the original number

Enter a complex number and cube it.

> $x := a+\mathrm{I}\,b$

$$x := a+\mathrm{I}\,b \qquad (8)$$

> x^3

$$(a+\mathrm{I}\,b)^3 \qquad (9)$$

> $evalc(\%)$

$$a^3-3\,a\,b^2+\mathrm{I}\,(3\,a^2\,b-b^3) \qquad (10)$$

The system only selects the cube of the $x = a + I\,b$ complex number and it has to be forced by the evalc procedure to do the operation. In this case we get the result immediately in canonical form. The situation is similar in the case of the calculation of the real and imaginary unit of the complex number and of its conjugate supposing we calculate symbolically.

> $\mathrm{Re}(x)$

$$\Re(a+\mathrm{I}\,b) \qquad (11)$$

> $evalc(\%)$

$$a \qquad (12)$$

The $\Re$ symbol, which created the real unit of the complex number given to its argument, can be found in the Common Symbols palette. The form of the instruction in 1-D Math is Re(x). It is not surprising that to create the result of the operation the evalc procedure was needed. The same is true for the usage of the conjugate procedure when we want to get the conjugate of the complex number.

> *conjugate*(*x*)

$$\overline{a+\mathrm{I}\,b} \qquad \textbf{(13)}$$

> *evalc*(%)

$$a-\mathrm{I}\,b \qquad \textbf{(14)}$$

In the task in hand we are looking for those x complex numbers for which $x^3=\overline{x}$.

> $x^3 = conjugate(x)$

$$(a+\mathrm{I}\,b)^3=\overline{a+\mathrm{I}\,b} \qquad \textbf{(15)}$$

> *evalc*(%)

$$a^3-3\,a\,b^2+\mathrm{I}\,(3\,a^2\,b-b^3)=a-\mathrm{I}\,b \qquad \textbf{(16)}$$

Notice that the evalc procedure had no problems with the fact that it got an equality as an argument. It executed the complex evaluation of the expressions on the right and the left sides and it gave the result in the form of another equality..

The equality of two complex numbers is equivalent with the equalities of its real and imaginary units . We chose the $\Re$ and $\Im$ ssymbols from the Common Symbols palette in the next two commands. The form of $\Im(x)$ alakja **1-D Math** is *Im(x)*.

> $valós := evalc(\mathrm{Re}(lhs(\textbf{(16)})) = \mathrm{Re}(rhs(\textbf{(16)})))$

$$valós := a^3-3\,a\,b^2=a \qquad \textbf{(17)}$$

> $képzetes := evalc(\mathrm{Im}(lhs(\textbf{(16)})) = \mathrm{Im}(rhs(\textbf{(16)})))$

$$képzetes := 3\,a^2\,b-b^3=-b \qquad \textbf{(18)}$$

We applied the lhs (left hand side) and rhs (right hand side) procedures to the equality (16) which create the expressions on the right and left side of the equality received as their argument. So we took the real unit of the right and left side of the equality and we joined these with an equal sign. The equality created this way became the value of the variable valós. In the case of the second command we did the same with the imaginary units.

The valós and képzetes provide a system of equation for unknowns a and b. Although the system of equation is a third degree in both of its variables, Maple can solve it.

> $solve(\{képzetes, valós\}, \{a, b\})$

$$\{b=0, a=0\}, \{b=1, a=0\}, \{b=-1, a=0\}, \{b=0, a=1\}, \{a=-1, b=0\}, \{a=\mathrm{RootOf}(1+2\,\_Z^2), b=\mathrm{RootOf}(1+2\,\_Z^2)\} \qquad \textbf{(19)}$$

We get the solutions of the system of equation as a sequence of sets. Each set contains two equalities and each of these provides the values of a and b. Now we can understand why solve does not give the solution calculated as a value to the unknowns of the equation. In the case of more solutions only the user can decide which one is needed. For us a and b are real numbers so those solutions are interesting in which both of them got a real value. Pick the sets providing the solutions from the (19) sequence and calculate the complex numbers determined by them.

```
> M:=NULL:
  for h in (19) do
```

```
  unassign('a','b'): assign(h);
  if Im(a)=0 and Im(b)=0 then M:=M,a+b*I fi
od;
M;
```

$$0, I, -I, 1, -1 \qquad (20)$$

We have solved the task. But the instructions above need some explanations.

We met a new form of the for repetition statement

for *x* **in** *X* **do** *ciklusmag_utasításai* **end do**;

in which x is a name (the name of a variable) and X is a set or a list. The body of the loop is executed to each element of X.

The conditional instruction was also new to us and its syntax is:

if *feltétel* **then** *utasítások* **end if**;

- The execution is done in a way that the system evaluates the condition and if a logical true value is created then it executes the instructions standing after the then key word. In the opposite case the system does not execute any instructions.
- Notice that in the previous versions of Maple the instructions of the body of the loop had to be closed between the do and od key words. However, the latest versions prefer the do and end do pair but due to compatibility considerations, both versions can be used. The same is true for the usage of the if-then-fi and the newer if-then-end if keywords used in the conditional instructions.
- We have already met the unassign procedure which, according to its name, has the opposite effect as the assign procedure which gives value to the variables. According to this the unassign frees the bounded variables.
- We have already met library procedures, packages and the short and long names of the procedures in the packages. It is time to summarise what has to be known about the construction of Maple. The procedures of Maple can be divided into two groups:
- • the procedures of the kernel and
- • the library procedures.
- The Maple.lib file contains the so-called top level procedures and the procedures of the packages. The procedures of the kernel were prepared in language C while the library procedures were created in the own language of Maple. It is also important to know that the procedures of the kernel and the top level procedures are available at the time of the loading of the system. The procedures in the packages have to be made available with the [képlet] command as the with[képlet] alternative solution. In the first case it is only the appointed procedures, and in the second case all the procedures of the package become available.
- What Have You Learnt About Maple?
- • We can use the Greek letters in Maple. The Greek letters have to be written with English spelling into the input line. If we want to use a Greek capital letter, then we only have to start to write the English name of it with a capital letter. The Greek letters can also be inserted from the Greek palette to the input line.
- • We have met the following procedures that operate on the complex numbers: the evalc procedure executes the evaluation of an expression and it gives the result in the canonical form. The Re and Im procedures create the real and the imaginary unit of the complex numbers,

while the conjugate procedure calculates the conjugate of a complex number. The common symbols [képlet] and [képlet] used for the calculation of the real and imaginary unit can be inserted from the Common Symbol palette.

- • The expressions on the left and right side of an equality can be created by the lhs and rhs instructions.
- • The normal procedure provides the simplification of the rational expressions.
- • The combine procedure transforms the product of the expressions containing trigonometric functions to the sum of expressions containing trigonometric functions with the trig option.
- • The syntax of the for instruction which we have met here:

for *x* **in** *X* **do**

...ciklusmag utasításai...

od,

where X is a set or a list. It makes the body of the loop be executed to all the components of the X, to all the elements of the set and the list.

- • The syntax of the conditional instruction

if *feltétel* **then** *utasítások_1* **else** *utasítások_2* **fi**.

The condition is evaluated and if it is true then the instructions_1, and if it is not true then the istrunctions_2 instruction sequence is executed. The else branch of the if instruction can be omitted. The

if *feltétel* **then** *utasítások* **fi**

executes the instructions instruction sequence only if the condition is true, otherwise it does nothing.

- Az alábbi táblázat a Maple felépítését illusztrálja.

Categories	Language	Availability
the procedures of the kernel	C	automatically at loading
top level procedures	Maple	automatically at loading
the procedures of the packages	Maple	**with the with procedure, e.g.** *with*(*plots*, *textplot*), *with*(*plots*)

Exercises

1. Assume that omega=–1/2+I*sqrt(3)/2. Calculate the following expressions:

- $(a+b)\,(a+b\,\mathrm{omega})\,(a+b\,\mathrm{omega}^2)$
- $(a+b\,\mathrm{omega}+c\,\mathrm{omega}^2)\,(a+b\,\mathrm{omega}^2+c\,\mathrm{omega})$
- $(a\,\mathrm{omega}^2+b\,\mathrm{omega})\,(b\,\mathrm{omega}^2+a\,\mathrm{omega})$

2. Create the conjugate, the real and imaginary unit of the following complex numbers

- $1 + 2\,\mathrm{I}$;
- $(1 + 2\,\mathrm{I})^7$
- $(3 - \mathrm{I})^3 - (3 + \mathrm{I})^3$;
- $\frac{1}{2} + \sqrt{3}\,\mathrm{I}$;
- $\frac{a + b\,\mathrm{I}}{a - b\,\mathrm{I}}$;

Determine that in which cases we needed to use the evalc procedure to get the desired solution.

3. Solve the following equations

- $x^2 - (2 + I)\,x + (-1 + 7\,I) = 0$
- $x^2 - (3 - 2\,I)\,x + (5 - 5\,I) = 0$

4. Solve the following system of equations

- $(1 + \mathrm{I})\,x + (1 + 2\,\mathrm{I})\,y + (1 + 3\,\mathrm{I})\,z + (1 + 4\,\mathrm{I})\,t = 1 + 5\,\mathrm{I}$
- $(3 - \mathrm{I})\,x + (4 - 2\,\mathrm{I})\,y + (1 + \mathrm{I})\,z + 4\,\mathrm{I}\,t = 2 - \mathrm{I}$

5. We recommend those readers who are familiar with programming to try out the following commands

$$interface(verboseproc = 2)$$

$$print(isprime)$$

The first command sets the verboseproc environment variable to 2 (the default is 1) then the print [képlet] command displays the source text of the isprime procedure on the screen. Try out other procedures as well.