1.4 The Representation of Numbers

Maple, as like every computer algebra-system, tries to represent the integers, rational, real and complex numbers exactly. But what does this dubious statement mean? This is what we are going to discuss in this chapter.

Let's start with the integers.

```
> 40!
```

$$815915283247897734345611269596115894272000000000 \quad \textbf{(1)}$$

A brief remark is needed at this point. The integer above is too big to be represented in any numeric systems in the usual way, even in double word. Indeed, Maple represents the integers in a so-called dynamic data vector, namely in a 10^4 based number system. And because it uses 17 bits to represent the length of the data vector, the number of the digits of the biggest number that can be represented is: $2^{17}-1$. The base of the number system is 10^4 thus the number of the digits of the biggest number that can be represented is $4\left(2^{17}-1\right)=524284$. Don't misunderstand it! This is not the biggest representable number. This is only the number of its digits.
Maple offers several procedures for the operations executed on the integers. It is impossible to go through all of them. Here we will confine our discussion only to some examples.

```
> ifactor(%)
```

$$(2)^{38}\,(3)^{18}\,(5)^{9}\,(7)^{5}\,(11)^{3}\,(13)^{3}\,(17)^{2}\,(19)^{2}\,(23)\,(29)\,(31)\,(37) \quad \textbf{(2)}$$

```
> expand(%)
```

$$815915283247897734345611269596115894272000000000 \quad \textbf{(3)}$$

```
> nextprime(%);
```

$$815915283247897734345611269596115894272000000053 \quad \textbf{(4)}$$

```
> prevprime(%%);
```

$$815915283247897734345611269596115894271999999917 \quad \textbf{(5)}$$

```
>
```

The **ifactor** procedure creates the prime factor decomposition of the integers, which is the product of the powers of the primes. We had the appointed operations executed with the **expand** procedure. Notice that we have got back the results above word for word.

The **nextprime** procedure determines the first prime subsequent to the integer given as a parameter, while the **prevprime** procedure determines the last prime prior to the integer given as a parameter. Based on the instructions above we can come to the conclusion that $40!+53$ and $40!-83$ are prime numbers.

The quotient and the remainder of the division executed on the integers and the greatest common divisor of two integers can be created by the **iquo**, **irem** and **igcd** procedures.

```
> iquo(40, 6), irem(40, 6), igcd(40, 6)
```

$$6, 4, 2 \quad \textbf{(6)}$$

Let's continue our investigations with the rational numbers.

> $-\frac{4}{18}$

(7)

$$-\frac{2}{9} \quad (7)$$

> $op(\%)$

$$-2, 9 \quad (8)$$

Notice that Maple did not execute the division. It did not respond that the result is -0.22. Instead it simplified with the greatest common divisor of the numerator and the denominator and then it represented the result as a pair of integers in which the denominator is positive. The result of the $op(\%)$ command also shows this.

The **op** procedure shows the components of its parameter. In this case the two operands are -2 (the numerator) and 9 (the denominator). This representation is called the representation of the rational numbers in the canonical form. We want to emphasise that with this solution the system gives the exact representation of the rational number above. If the specified division had been executed then this exactness would have lost.

In the case of real numbers the situation is more difficult. Let's see.

> $s := \sqrt{3}$

$$s := \sqrt{3} \quad (9)$$

> $op(s)$

$$3, \frac{1}{2} \quad (10)$$

> $evalc\left(\sqrt{-18 + I}\right)$

$$\frac{1}{2}\sqrt{-36 + 10\sqrt{13}} + \frac{1}{2} \mathrm{I} \sqrt{36 + 10\sqrt{13}} \quad (11)$$

The solutions of Maple are great because they are obvious. This is also true in this case: instead of executing the appointed operations and after carrying out some conversions, the system shows the real number with the expression that created it. In this way the representation is exact. The formulas are shown on the screen as it is usual in mathematics.

The imaginary unit I ($I^2 = -1$) gives an example of the constants built in the system. More information can be gained with the help of the *help*("constant") command. But now let's look at some examples.

> Pi, *true*, *false*, *infinity*

$$\pi, \mathit{true}, \mathit{false}, \infty \quad (12)$$

> $\sin\left(\frac{\mathrm{Pi}}{4}\right)$

$$\frac{1}{2}\sqrt{2} \quad (13)$$

> $testeq\left((a + b)(a - b) = a^2 - b^2\right)$

$$\mathit{true} \quad (14)$$

>

Naturally Maple provides the approximation of the real numbers with arbitrary exactness.

> $s := \sqrt{2}$

$$s := \sqrt{2} \tag{15}$$

> *evalf*(*s*, 4); *evalf*(*s*, 40); *evalf*(*s*)

$$1.414$$
$$1.414213562373095048801688724209698078570$$
$$1.414213562 \tag{16}$$

>

The **evalf** procedure creates the real approximation of its argument, namely with the exactness given by the second argument. In case we do not give a second parameter then the value of the **Digits** environment variable determines the exactness of the real approximation.

> *Digits*

$$10 \tag{17}$$

> *Digits* := 120; *evalf*(*s*); *evalf*(Pi)

$$Digits := 120$$

1.4142135623730950488016887242096980785696718753769480731766797379907324784621\
0703885038753432764157273501384623091229702

3.1415926535897932384626433832795028841971693993751058209749445923078164062862\
0899862803482534211706798214808651328230665 **(18)**

> *Digits* := 10

$$Digits := 10 \tag{19}$$

The default of **Digits** is 10. After we set it to 120 the system gives the values of $\sqrt{2}$ and π with the exactness of 120 significant digits.

What Have You Learnt About Maple?

- Maple is able to represent extremely large, approximately 500 000 digit integers. It represents exactly the rational numbers as integer pairs in which the numerator and the always positive denominator are relative primes. To represent the real and complex numbers it uses the expressions which have generated them. Consequently, the operations executed on the numbers represented in this way are supported by software arithmetic.
- The **evalf** procedure provides for the approximation of real numbers, namely with the arbitrary exactness that can be set runtime. We can specify the exactness of the approximation as the second parameter of **evalf**. If we do not give a second parameter then it is the value of the **Digits** environment variable that determines the exactness of the approximation.
- The **evalc** procedure executes the operations selected for the complex numbers and it creates the canonical form of the result, that is, the complex number.
- The **testeq** procedure examines the equivalence of expressions and it gives a logical value (**true**, **false**) as a result.

Exercises

1. Execute the following operations.

- 2^{200}
- $1000!$
- $ifactor(10^{10})$
- $2^{\left(2^{\left(2^{\left(2^{\left(2^{2}\right)}\right)}\right)}\right)}$
- $(2^2)^{\left((2^2)^{(2^2)}\right)}$
- $\left(\left(\left((2^2)^2\right)^2\right)^2\right)^2$

2. Determine the biggest primes smaller than 10, 100, 1000, 10000 and 100000.

3. Determine the greatest common divisor and the least common multiple of 54321 and 12345. Give the ratio and the remainder when the two numbers are dividing with remainder.

4. Calculate the area and the volume of the circle and respectively the sphere with radius 2.3 cm. Calculate the area of the square and the volume of cube with side length 3.2 cm for 24 significant digits.

5. Calculate the following expressions:

- $\sqrt{2\,I}$
- $\sqrt{-8\,I}$
- $\sqrt{4+I}+\sqrt{4-I}$
- $(-1)^{\left(\frac{1}{4}\right)}$
- $\sqrt{1-I\sqrt{3}}$

6. . Assume that a:=2. Check the following equalities with the **testeq** procedure.

- $a^3-4=a^2$
- $a^3-4=a+2$
- $a^3=\sin(a)$
- $a^3-b^3=(a-b)\left(a^2+a\,b+b^2\right)$
- $\tan(a)=\dfrac{\sin(a)}{\cos(a)}$