CHAPTER 3

Numbers and Numeral Systems

Numbers play an important role in almost all areas of mathematics, not least in calculus. Virtually all calculus books contain a thorough description of the natural, rational, real and complex numbers, so we will not repeat this here. An important concern for us, however, is to understand the basic principles behind how a computer handles numbers and performs arithmetic, and for this we need to consider some facts about numbers that are usually not found in traditional calculus texts.

More specifically, we are going to review the basics of the decimal numeral system, where the base is 10, and see how numbers may be represented equally well in other numeral systems where the base is not 10. We will study representation of real numbers as well as arithmetic in different bases. Throughout the chapter we will pay special attention to the binary numeral system (base 2) as this is what is used in most computers. This will be studied in more detail in the next chapter.

3.1 Terminology and Notation

We will usually introduce terminology as it is needed, but certain terms need to be agreed upon straightaway. In your calculus book you will have learnt about natural, rational and real numbers. The natural numbers $\mathbb{N}_0 = \{0,1,2,3,4,\dots\}$[1] are the most basic numbers in that both rational and real numbers can be constructed from them. Any positive natural number n has an opposite number

[1] In most books the natural numbers start with 1, but for our purposes it is convenient to include 0 as a natural number as well. To avoid confusion we have therefore added 0 as a subscript.

$-n$, and we denote by $\mathbb{Z}$ the set of natural numbers augmented with all these negative numbers,

$$\mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}.$$

We will refer to $\mathbb{Z}$ as the set of integer numbers or just the integers.

Intuitively it is convenient to think of a real number x as a decimal number with (possibly) infinitely many digits to the right of the decimal point. We then refer to the number obtained by setting all the digits to the right of the decimal point to 0 as the *integer part* of x. If we replace the integer part by 0 we obtain the *fractional part* of x. If for example $x = 3.14$, its integer part is 3 and its fractional part is 0.14. A number that has no integer part will often be referred to as a fractional number.

Definition 3.1. *Let $x = d_k d_{k-1} \cdots d_2 d_1 d_0 \, . \, d_{-1} d_{-2} \cdots$ be a decimal number. Then the number $d_k d_{k-1} \cdots d_1 d_0$ is called the integer part of x while the number $0 \, . \, d_{-1} d_{-2} \cdots$ is called the fractional part of x.*

For rational numbers there are standard operations we can perform to find the integer and fractional parts. When two positive natural numbers a and b are divided, the result will usually not be an integer, or equivalently, there will be a remainder. Let us agree on some notation for these operations.

Notation 3.2 (Integer division and remainder)**.** *If a and b are two integers, the number $a \,/\!/\, b$ is the result obtained by dividing a by b and discarding the remainder. The number $a \,\%\, b$ is the remainder when a is divided by b.*

For example $3 \,/\!/\, 2 = 1$, $9 \,/\!/\, 4 = 2$ and $24 \,/\!/\, 6 = 4$, while $3 \,\%\, 2 = 1$, $23 \,\%\, 5 = 3$, and $24 \,\%\, 4 = 0$.

We will use standard notation for intervals of real numbers. Two real numbers a and b with $a < b$ define four intervals that only differ in whether the end points a and b are included or not. The closed interval $[a, b]$ contains all real numbers between a and b, including the end points. Formally we can express this by $[a, b] = \{x \in \mathbb{R} \mid a \le x \le b\}$. The other intervals can be defined similarly,

Definition 3.3 (Intervals)**.** *Two real numbers a and b defined the four intervals*

$$(a,b) = \{x \in \mathbb{R} \mid a < x < b\} \text{ (open)}; \qquad (a,b] = \{x \in \mathbb{R} \mid a < x \le b\} \text{ (half open)};$$
$$[a,b] = \{x \in \mathbb{R} \mid a \le x \le b\} \text{ (closed)}; \qquad [a,b) = \{x \in \mathbb{R} \mid a \le x < b\} \text{ (half open)}.$$

With this notation we can say that a fractional number is a real number in the interval $[0,1)$.

3.2 Natural Numbers in Different Numeral Systems

We usually represent natural numbers in the decimal numeral system, but in this section we are going to see that this is just one of infinitely many numeral systems. We will also give a simple method for converting a number from its decimal representation to its representation in a different base.

3.2.1 Alternative Numeral Systems

In the decimal system we use a positional convention and express numbers in terms of the ten digits 0, 1, ..., 8, 9, and let the position of a digit determine how much it is worth. For example the string of digits 3761 is interpreted as

$$3761 = 3 \times 10^3 + 7 \times 10^2 + 6 \times 10^1 + 1 \times 10^0.$$

Numbers that have a simple representation in the decimal numeral system are often thought of as special. For example it is common to celebrate a 50th birthday in a special way or mark the centenary anniversary of an important event like a country's independence. However, the numbers 50 and 100 are only special when they are written in the decimal numeral system.

Any natural number can be used as the base for a numeral system. Consider for example the *septenary* numeral system which has 7 as the base and uses the digits 0-6. In this system the numbers 3761, 50 and 100 become

$$\begin{aligned}
3761 &= 13652_7 = 1 \times 7^4 + 3 \times 7^3 + 6 \times 7^2 + 5 \times 7^1 + 2 \times 7^0, \\
50 &= 101_7 = 1 \times 7^2 + 0 \times 7^1 + 1 \times 7^0, \\
100 &= 202_7 = 2 \times 7^2 + 0 \times 7^1 + 2 \times 7^0,
\end{aligned}$$

so 50 and 100 are not quite as special as in the decimal system.

These examples make it quite obvious that we can define numeral systems with almost any natural number as a base. The only restriction is that the base must be greater than one. To use 0 as base is quite obviously meaningless, and if we try to use 1 as base we only have the digit 0 at our disposal, which means that we can only represent the number 0. We record the general construction in a formal definition.

Definition 3.4. *Let β be a natural number greater than 1 and let n_0, n_1, ..., $n_{\beta-1}$ be β distinct numerals (also called digits) such that n_i denotes the number $n_i = i$. A natural number representation in base β is an ordered collection of digits $(d_k d_{n-1} \dots d_1 d_0)_\beta$ which is interpreted as the natural number*

$$d_k\beta^k + d_{k-1}\,\beta^{k-1} \,+ d_{k-2}\,\beta^{k-2} + \cdots + d_1\beta^1 + d_0\beta^0$$

where each digit d_i is one of the β numerals $\{n_i\}_{i=0}^{\beta-1}$.

Formal definitions in mathematics often appear complicated until one gets under the surface, so let us consider the details of the definition. The base β is not so mysterious. In the decimal system $\beta = 10$ while in the septenary system $\beta = 7$. The beginning of the definition simply states that any natural number greater than 1 can be used as a base.

In the decimal system we use the digits 0–9 to write down numbers, and in any numeral system we need digits that can play a similar role. If the base is 10 or less it is natural to use the obvious subset of the decimal digits as numerals. If the base is 2 we use the two digits $n_0 = 0$ and $n_1 = 1$; if the base is 5 we use the five digits $n_0 = 0$, $n_1 = 1$, $n_2 = 2$, $n_3 = 3$ and $n_4 = 4$. However, if the base is greater than 10 we have a challenge in how to choose numerals for the numbers $10, 11, \dots, \beta-1$. If the base is less than 40 it is common to use the decimal digits together with the initial characters of the latin alphabet as numerals. In base $\beta = 16$ for example, it is common to use the digits 0–9 augmented with $n_{10} = a$, $n_{11} = b$, $n_{12} = c$, $n_{13} = d$, $n_{14} = e$ and $n_{15} = f$. This is called the *hexadecimal* numeral system and in this system the number 3761 becomes

$$eb1_{16} = e \times 16^2 + b \times 16^1 + 1 \times 16^0 = 14 \times 256 + 11 \times 16 + 1 = 3761.$$

Definition 3.4 defines how a number can be expressed in the numeral system with base β. However, it does not say anything about how to find the digits of a fixed number. And even more importantly, it does not guarantee that a number can be written in the base-β numeral system in only one way. This is settled in our first lemma below.

Lemma 3.5. *Any natural number can be represented uniquely in the base-β numeral system.*

Proof. To keep the argument as transparent as possible, we give the proof for a specific example, namely $a = 3761$ and $\beta = 8$ (the octal numeral system). Since

$8^4 = 4096 > a$, we know that the base-8 representation cannot contain more than 4 digits. Suppose that $3761 = (d_3d_2d_1d_0)_8$; our job is to find the value of the four digits and show that each of them only has one possible value. We start by determining d_0. By definition of base-8 representation of numbers we have the relation

$$3761 = (d_3d_2d_1d_0)_8 = d_3 8^3 + d_2 8^2 + d_1 8 + d_0. \tag{3.1}$$

We note that only the last term in the sum on the right is not divisible by 8, so the digit d_0 must therefore be the remainder when 3761 is divided by 8. If we perform the division we find that

$$d_0 = 3761 \,\%\, 8 = 1, \qquad 3761 \,//\, 8 = 470.$$

We observe that when the right-hand side of (3.1) is divided by 8 and the remainder discarded, the result is $d_3 8^2 + d_2 8 + d_1$. In other words be must have

$$470 = d_3 8^2 + d_2 8 + d_1.$$

But then we see that d_1 must be the remainder when 470 is divided by 8. If we perform this division we find

$$d_1 = 470 \,\%\, 8 = 6, \qquad 470 \,//\, 8 = 58.$$

Using the same argument as before we see that the relation

$$58 = d_3 8 + d_2 \tag{3.2}$$

must hold. In other words d_2 is the remainder when 58 is divided by 8,

$$d_2 = 58 \,\%\, 8 = 2, \qquad 58 \,//\, 8 = 7.$$

If we divide both sides of (3.2) by 8 and drop the remainder we are left with $7 = d_3$. The net result is that $3761 = (d_3d_2d_1d_0)_8 = 7261_8$.

We note that during the computations we never had any choice in how to determine the four-digits, they were determined uniquely. We therefore conclude that the only possible way to represent the decimal number 3761 in the base-8 numeral system is as 7261_8. ∎

The proof is clearly not complete since we have only verified Lemma 3.5 in a special case. However, the same argument can be used for any a and β and we leave it to the reader to write down the details in the general case.

Lemma 3.5 says that any natural number can be expressed in a unique way in any numeral system with base greater than 1. We can therefore use any such numeral system to represent numbers. Although we may feel that we always use the decimal system, we all use a second system every day, the base-60 system. An hour is split into 60 minutes and a minute into 60 seconds. The great advantage of using 60 as a base is that it is divisible by 2, 3, 4, 5, 6, 10, 12, 15, 20 and 30 which means that an hour can easily be divided into many smaller parts without resorting to fractions of minutes. Most of us also use other numeral systems without knowing. Virtually all electronic computers use the base-2 (binary) system and we will see how this is done in the next chapter.

We are really only considering natural numbers in this section, but let us add a comment about how to represent negative numbers in the base-β numeral system. This is not so difficult. There is nothing particularly decimal about the minus sign, so the number $-a$ may be represented like a, but preceded with $-$. Therefore, we represent for example the decimal number -3761 as -7261_8 in the octal numeral system.

3.2.2 Conversion to the Base-β Numeral System

The method used in the proof of Lemma 3.5 for converting a number to base β is important, so we record it as an algorithm.

Algorithm 3.6. *Let a be a natural number that in base β has the $k+1$ digits $(d_k d_{k-1} \cdots d_0)_\beta$. These digits may be computed by performing the operations:*

```
$a_0 = a$;
for $i = 0, 1, \ldots, k$
    $d_i = a_i \,\%\, \beta$;
    $a_{i+1} = a_i \,//\, \beta$;
```

Let us add a little explanation since this is our first algorithm apart from the examples in section 1.4. We start by setting the variable a_0 equal to a, the number whose digits we want to determine. We then let i take on the values 0, 1, 2, ..., k. For each value of i we perform the operations that are indented, i.e., we compute the numbers $a_i \,\%\, \beta$ and $a_i \,//\, \beta$ and store the results in the variables d_i and a_{i+1}.

Algorithm 3.6 demands that the number of digits in the representation to be computed is known in advance. If we look back on the proof of Lemma 3.5, we note that we do not first check how many digits we are going to compute, since when we are finished the number that we divide (the number a_i in Algorithm 3.6) has become 0. We can therefore just repeat the two indented statements in the algorithm until the result of the division becomes 0. The following version of the algorithm incorporates this. We also note that we do not need to keep the results of the divisions; we can omit the subscript and store the result of the division $a \mathbin{/\!/} \beta$ back in a.

Algorithm 3.7. *Let a be a natural number that in base β has the $k+1$ digits $(d_k d_{k-1} \cdots d_0)_\beta$. These digits may be computed by performing the operations:*

```
$i = 0$;
while $a > 0$
    $d_i = a \,\%\, \beta$;
    $a = a \mathbin{/\!/} \beta$;
    $i = i + 1$;
```

Recall that the statement 'while $a > 0$' means that all the indented statements will be repeated until a becomes 0.

It is important to realise that the order of the indented statements is not arbitrary. When we do not keep all the results of the divisions, it is essential that d_i (or d) is computed before a is updated with its new value. And when i is initialised with 0, we must update i at the end, since otherwise the subscript in d_i will be wrong.

The variable i is used here so that we can number the digits correctly, starting with d_0, then d_1 and so on. If this is not important, we could omit the first and the last statements, and replace d_i by d. The algorithm then becomes

```
while $a > 0$
    $d = a \,\%\, \beta$;
    $a = a \mathbin{/\!/} \beta$;
    print $d$;
```

Here we have also added a print-statement so the digits of a will be printed (in reverse order).

The results produced by Algorithm 3.7 can be conveniently organised in a table. The example in the proof of Lemma 3.5 can be displayed as

3761	1
470	6
58	2
7	7

The left column shows the successive integer parts resulting from repeated division by 8, whereas the right column shows the remainder in these divisions. Let us consider one more example.

Example 3.8. Instead of converting 3761 to base 8 let us convert it to base 16. We find that 3761 // 16 = 235 with remainder 1. In the next step we find 235 // 16 = 14 with remainder 11. Finally we have 14 // 16 = 0 with remainder 14. Displayed in a table this becomes

3761	1
235	11
14	14

Recall that in the hexadecimal system the letters a–f usually denote the values 10–15. We have therefore found that the number 3761 is written eb1_{16} in the hexadecimal numeral system. ∎

Since we are particularly interested in how computers manipulate numbers, let us also consider an example of conversion to the binary numeral system, as this is the numeral system used in most computers. Instead of dividing by 16 we are now going to repeatedly divide by 2 and record the remainder. A nice thing about the binary numeral system is that the only possible remainders are 0 and 1: it is 0 if the number we divide is an even integer and 1 if the number is an odd integer.

Example 3.9. Let us continue to use the decimal number 3761 as an example, but now we want to convert it to binary form. If we perform the divisions and record the results as before we find

3761	1
1880	0
940	0
470	0
235	1
117	1
58	0
29	1
14	0
7	1
3	1
1	1

In other words we have $3761 = 111010110001_2$. This example illustrates an important property of the binary numeral system: Computations are simple, but long and tedious. This means that this numeral system is not so good for humans as we tend to get bored and make sloppy mistakes. For computers, however, this is perfect as computers do not make mistakes and work extremely fast. ∎

3.2.3 Conversion between base-2 and base-16

Computers generally use the binary numeral system internally, and in chapter 4 we are going to study this in some detail. A major disadvantage of the binary system is that even quite small numbers require considerably more digits than in the decimal system. There is therefore a need for a more compact representation of binary numbers. It turns out that the hexadecimal numeral system is convenient for this purpose.

Suppose we have the one-digit hexadecimal number $x = a_{16}$. In binary it is easy to see that this is $x = 1010_2$. A general four-digit binary number $(d_3 d_2 d_1 d_0)_2$ has the value

$$d_0 2^0 + d_1 2^1 + d_2 2^2 + d_3 2^3,$$

and must be in the range 0–15, which corresponds exactly to a one-digit hexadecimal number.

Observation 3.10. *A four-digit binary number can always be converted to a one-digit hexadecimal number, and vice versa.*

This simple observation is the basis for converting general numbers between binary and hexadecimal representation. Suppose for example that we have the eight digit binary number $x = 1100\ 1101_2$. This corresponds to the number

$$\begin{aligned}1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3 + 0 \times 2^4 + 0 \times 2^5 + 1 \times 2^6 + 1 \times 2^7 \\ = (1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3) + (0 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3)2^4.\end{aligned}$$

The two numbers in brackets are both in the range 0–15 and can therefore be represented as one-digit hexadecimal numbers. In fact we have

$$\begin{aligned}1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3 = 13_{10} = d_{16}, \\ 0 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3 = 12_{10} = c_{16}.\end{aligned}$$

But then we have

$$\begin{aligned}x &= (1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3) + (0 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3)2^4 \\ &= d_{16} \times 16^0 + 16^1 \times c_{16} = cd_{16}.\end{aligned}$$

The short version of this detailed derivation is that the eight digit binary number $x = 1100\ 1101_2$ can be converted to hexadecimal by converting the two groups of

Hex	Bin	Hex	Bin	Hex	Bin	Hex	Bin
0	0	4	100	8	1000	c	1100
1	1	5	101	9	1001	d	1101
2	10	6	110	a	1010	e	1110
3	11	7	111	b	1011	f	1111

Table 3.1. Conversion between hexadecimal and binary representation.

four binary digits separately. This results in two one-digit hexadecimal numbers, and these are the hexadecimal digits of x,

$$1100_2 = c_{16}, \quad 1101_2 = d_{16}, \quad 1100\ 1101_2 = cd_{16}.$$

This works in general.

Observation 3.11. *A hexadecimal natural number can be converted to binary by converting each digit separately. Conversely, a binary number can be converted to hexadecimal by converting each group of successive binary digits into hexadecimal, starting with the least significant digits.*

Example 3.12. Let us convert the hexadecimal number $3c5_{16}$ to binary. We have

$$5_{16} = 0101_2,$$
$$c_{16} = 1100_2,$$
$$3_{16} = 0011_2,$$

which means that $3c5_{16} = 11\ 1100\ 0101_2$ where we have omitted the two leading zeros. ▮

Observation 3.11 means that to convert between binary and hexadecimal representation we only need to know how to convert numbers in the range 0–15 (decimal). Most will perhaps do this by going via decimal representation, but all the conversions can be found in table 3.1.

3.3 Representation of Fractional Numbers

We have seen how integers can be represented in numeral systems other than decimal, but what about fractions and irrational numbers? In the decimal system such numbers are characterised by the fact that they have two parts, one to the left of the decimal point, and one to the right, like the number 21.828. The part to the left of the decimal point — the integer part — can be represented in base-β as outlined above. If we can represent the part to the right of the decimal point — the fractional part — in base-β as well, we can follow the convention from the decimal system and use a point to separate the two parts. Negative rational and irrational numbers are as easy to handle as negative integers, so we focus on how to represent positive numbers without an integer part, in other words numbers in the open interval $(0,1)$.

3.3.1 Rational and Irrational Numbers in Base-β

Let a be a real number in the interval $(0,1)$. In the decimal system we can write such a number as 0, followed by a point, followed by a finite or infinite number of decimal digits, as in

$$0.45928\ldots$$

This is interpreted as the number

$$4\times 10^{-1} + 5\times 10^{-2} + 9\times 10^{-3} + 2\times 10^{-4} + 8\times 10^{-5} + \cdots .$$

From this it is not so difficult to see what a base-β representation must look like.

Definition 3.13. *Let β be a natural number greater than 1 and let n_0, n_1, ..., $n_{\beta-1}$ be β distinct numerals (also called digits) such that n_i denotes the number $n_i = i$. A fractional representation in base β is a finite or infinite, ordered collection of digits $(0.d_{-1}d_{-2}d_{-3}\ldots)_\beta$ which is interpreted as the real number*

$$d_{-1}\beta^{-1} + d_{-2}\,\beta^{-2} + d_{-3}\,\beta^{-3} + \cdots \tag{3.3}$$

where each digit d_i is one of the β numerals $\{n_i\}_{i=0}^{\beta-1}$.

Definition 3.13 is considerably more complicated than definition 3.4 since we may have an infinite number of digits. This becomes apparent if we try to check the size of numbers on the form given by (3.3). Since none of the terms in the sum are negative, the smallest number is the one where all the digits are 0, i.e., where $d_i = 0$ for $i = -1, -2, \ldots$. But this can be nothing but the number 0.

The largest possible number occurs when all the digits are as large as possible, i.e. when $d_i = \beta - 1$ for all i. If we call this number x, we find

$$\begin{aligned} x &= (\beta-1)\beta^{-1} + (\beta-1)\beta^{-2} + (\beta-1)\beta^{-3} + \cdots \\ &= (\beta-1)\beta^{-1}(1 + \beta^{-1} + \beta^{-2} + \cdots \\ &= \frac{\beta-1}{\beta}\sum_{i=0}^{\infty}(\beta^{-1})^i. \end{aligned}$$

In other words x is given by a sum of an infinite geometric series with factor $\beta^{-1} = 1/\beta < 1$. This series converges to $1/(1-\beta^{-1})$ so x has the value

$$x = \frac{\beta-1}{\beta}\frac{1}{1-\beta^{-1}} = \frac{\beta-1}{\beta}\frac{\beta}{\beta-1} = 1.$$

Let us record our findings so far.

Lemma 3.14. *Any number on the form* (3.3) *lies in the interval* $[0,1]$.

The fact that the base-β fractional number with all digits equal to $\beta-1$ is the number 1 is a bit disturbing since it means that real numbers cannot be represented uniquely in base β. In the decimal system this corresponds to the fact that 0.99999999999999... (infinitely many 9s) is in fact the number 1. And this is not the only number that has two representations. Any number that ends with an infinite number of digits equal to $\beta-1$ has a simpler representation. Consider for example the decimal number 0.12999999999999.... Using the same technique as above we find that this number is 0.13. However, it turns out that these are the only numbers that have a double representation, see theorem 3.15 below.

Let us now see how we can determine the digits of a fractional number in a numeral system other than the decimal one. As for natural numbers, it is easiest to understand the procedure through an example, so we try to determine the digits of 1/5 in the octal (base 8) system. According to definition 3.13 we seek digits $d_{-1}d_{-2}d_{-3}\dots$ (possibly infinitely many) such that the relation

$$\frac{1}{5} = d_{-1}8^{-1} + d_{-2}8^{-2} + d_{-3}8^{-3} + \cdots \tag{3.4}$$

becomes true. If we multiply both sides by 8 we obtain

$$\frac{8}{5} = d_{-1} + d_{-2}8^{-1} + d_{-3}8^{-2} + \cdots. \tag{3.5}$$

The number 8/5 lies between 1 and 2 and we know from Lemma 3.14 that the sum $d_{-2}8^{-1} + d_{-3}8^{-2} + \cdots$ can be at most 1. Therefore we must have $d_{-1} = 1$. Since d_{-1} has been determined we can subtract this from both sides of (3.5)

$$\frac{3}{5} = d_{-2}8^{-1} + d_{-3}8^{-2} + d_{-4}8^{-3} + \cdots. \tag{3.6}$$

This equation has the same form as (3.4) and can be used to determine d_{-2}. We multiply both sides of (3.6) by 8,

$$\frac{24}{5} = d_{-2} + d_{-3}8^{-1} + d_{-4}8^{-3} + \cdots. \tag{3.7}$$

The fraction 24/5 lies in the interval $(4,5)$ and since the terms on the right that involve negative powers of 8 must be a number in the interval $[0,1]$, we must have $d_{-2} = 4$. We subtract this from both sides of (3.7) and obtain

$$\frac{4}{5} = d_{-3}8^{-1} + d_{-4}8^{-2} + d_{-5}8^{-3} + \cdots. \tag{3.8}$$

Multiplication by 8 now gives

$$\frac{32}{5} = d_{-3} + d_{-4}8^{-1} + d_{-5}8^{-2} + \cdots.$$

from which we conclude that $d_{-3} = 6$. Subtracting 6 and multiplying by 8 we obtain

$$\frac{16}{5} = d_{-4} + d_{-5}8^{-1} + d_{-6}8^{-2} + \cdots.$$

from which we conclude that $d_{-4} = 3$. If we subtract 3 from both sides we find

$$\frac{1}{5} = d_{-5}8^{-1} + d_{-6}8^{-2} + d_{-7}8^{-3} + \cdots.$$

But this relation is essentially the same as (3.4), so if we continue we must generate the same digits again. In other words, the sequence $d_{-5}d_{-6}d_{-7}d_{-8}$ must be the same as $d_{-1}d_{-2}d_{-3}d_{-4} = 1463$. But once d_{-8} has been determined we must again come back to a relation with 1/5 on the left, so the same digits must also repeat in $d_{-9}d_{-10}d_{-11}d_{-12}$ and so on. The result is that

$$\frac{1}{5} = 0.1463146314631463\cdots_8.$$

Based on this procedure we can prove an important theorem.

Theorem 3.15. *Any real number in the interval* $(0,1)$ *can be represented in a unique way as a fractional base-β number provided representations with infinitely many trailing digits equal to $\beta - 1$ are prohibited.*

Proof. We have already seen how the digits of 1/5 in the octal system can be determined, and it is easy to generalise the procedure. However, there are two additional questions that must be settled before the claims in the theorem are completely settled.

We first prove that the representation is unique. If we look back on the conversion procedure in the example we considered, we had no freedom in the choice of any of the digits. The digit d_{-2} was for example determined by equation 3.7, where the left-hand side is 4.8 in the decimal system. Then our only hope of satisfying the equation is to choose $d_{-2} = 4$ since the remaining terms can only sum up to a number in the interval $[0, 1]$.

How can the procedure fail to determine the digits uniquely? In our example, any digit is determined by an equation on the form (3.7), and as long as the left-hand side is not an integer, the corresponding digit is uniquely determined. If the left-hand side should happen to be an integer, as in

$$5 = d_{-2} + d_{-3}8^{-1} + d_{-4}8^{-3} + \cdots,$$

the natural solution is to choose $d_{-2} = 5$ and choose all the remaining digits as 0. However, since we know that 1 may be represented as a fractional number with all digits equal to 7, we could also choose $d_{-2} = 4$ and $d_i = 7$ for all $i < -2$. The natural solution is to choose $d_{-2} = 5$ and prohibit the second solution. This is exactly what we have done in the statement of the theorem, so this secures the uniqueness of the representation.

The second point that needs to be settled is more subtle; do we really compute the correct digits? It may seem strange to think that we may not compute the right digits since the digits are forced upon us by the equations. But if we look carefully, the equations are not quite standard since the sums on the right may contain infinitely many terms. In general it is therefore impossible to achieve equality in the equations, all we can hope for is that we can make the sum on the right in (3.4) come as close to $1/5$ as we wish by including sufficiently many terms.

Set $a = 1/5$. Then equation (3.6) can be written

$$8(a - d_{-1}8^{-1}) = d_{-2}8^{-1} + d_{-3}8^{-2} + d_{-4}8^{-3} + \cdots$$

while (3.8) can be written

$$8^2(a - d_{-1}8^{-1} - d_{-2}8^{-2}) = d_{-3}8^{-1} + d_{-4}8^{-2} + d_{-5}8^{-3} + \cdots.$$

After i steps the equation becomes

$$8^i(a - d_{-1}8^{-1} - d_{-2}8^{-2} - \cdots - d_{-i}8^{-i}) = \\ d_{-i-1}8^{-1} + d_{-i-2}8^{-2} + d_{-i-3}8^{-3} + \cdots.$$

The expression in the bracket on the left we recognise as the error e_i in approximating a by the first i numerals in the octal representation. We can rewrite this slightly and obtain

$$e_i = 8^{-i}(d_{-i-1}8^{-1} + d_{-i-2}8^{-2} + d_{-i-3}8^{-3} + \cdots).$$

From Lemma 3.14 we know that the number in the bracket on the right lies in the interval $[0,1]$ so we have $0 \le e_i \le 8^{-i}$. But this means that by including sufficiently many digits (choosing i sufficiently big), we can get e_i to be as small as we wish. In other words, by including sufficiently many digits, we can get the octal representation of $a = 1/5$ to be as close to a as we wish. Therefore our method for computing numerals does indeed generate the digits of a. ∎

3.3.2 An Algorithm for Converting Fractional Numbers

The basis for the proof of Theorem 3.15 is the procedure for computing the digits of a fractional number in base-β. We only considered the case $\beta = 8$, but it is simple to generalise the algorithm to arbitrary β.

Algorithm 3.16. *Let a be a fractional number whose first k digits in base β are $(0.d_{-1}d_{-2}\cdots d_{-k})_\beta$. These digits may be computed by performing the operations:*

```
for $i = -1, -2, \dots, -k$
    $d_i = \lfloor a * \beta \rfloor$;
    $a = a * \beta - d_i$;
```

Compared with the description on pages 42 to 43 there should be nothing mysterious in this algorithm except for perhaps the notation $\lfloor x \rfloor$. This is a fairly standard way of writing the *floor function* which is equal to the largest integer that is less than or equal to x. We have for example $\lfloor 3.4 \rfloor = 3$ and $\lfloor 5 \rfloor = 5$.

When converting natural numbers to base-β representation there is no need to know or compute the number of digits beforehand, as is evident in algorithm 3.7. For fractional numbers we do need to know how many digits to compute as there may often be infinitely many. A for-loop is therefore a natural construction in algorithm 3.16.

It is convenient to have a standard way of writing down the computations involved in converting a fractional number to base-β, and it turns out that we can use the same format as for converting natural numbers. Let us take as an example the computations in the proof of theorem 3.15 where the fraction $1/5$ was converted to base-8. We start by writing the number to be converted to the left of the vertical line. We then multiply the number by β (which is 8 in this case) and write the integer part of the result, which is the first digit, to the right of the line. The result itself we write in brackets to the right. We then start with the fractional part of the result one line down and continue until the result becomes 0 or we have all the digits we want,

1/5	1	(8/5)
3/5	4	(24/5)
4/5	6	(32/5)
2/5	3	(16/5)
1/5	1	(8/5)

Here we are back at the starting point, so the same digits will just repeat again.

3.3.3 Conversion between binary and hexadecimal

It turns out that hexadecimal representation is handy short-hand for the binary representation of fractional numbers, just like it was for natural numbers. To see why this is, we consider the number $x = 0.11010111_2$. In all detail this is

$$\begin{aligned} x &= 1\times 2^{-1} + 1\times 2^{-2} + 0\times 2^{-3} + 1\times 2^{-4} + 0\times 2^{-5} + 1\times 2^{-6} + 1\times 2^{-7} + 1\times 2^{-8} \\ &= 2^{-4}(1\times 2^3 + 1\times 2^2 + 0\times 2^1 + 1\times 2^0) + 2^{-8}(0\times 2^3 + 1\times 2^2 + 1\times 2^1 + 1\times 2^0) \\ &= 16^{-1}(1\times 2^3 + 1\times 2^2 + 0\times 2^1 + 1\times 2^0) + 16^{-2}(0\times 2^3 + 1\times 2^2 + 1\times 2^1 + 1\times 2^0). \end{aligned}$$

From table 3.1 we see that the two four-digit binary numbers in the brackets correspond to the hexadecimal numbers $1101_2 = d_{16}$ and $111_2 = 7_{16}$. We therefore have

$$x = 16^{-1}13 + 16^{-2}7 = 0.d7_{16}.$$

As for natural numbers, this works in general.

Observation 3.17. *A hexadecimal fractional number can be converted to binary by converting each digit separately. Conversely, a binary fractional number can be converted to hexadecimal by converting each group of four successive binary digits to hexadecimal, starting with the most significant digits.*

A couple of examples will illustrate how this works in general.

Example 3.18. Let us convert the number $x = 0.3a8_{16}$ to binary. From table 3.1 we find

$$3_{16} = 0011_2, \quad a_{16} = 1010_2, \quad 8_{16} = 1000_2,$$

which means that

$$0.3a8_{16} = 0.0011\ 1010\ 1000_2 = 0.0011\ 1010\ 1_2. \quad \blacksquare$$

Example 3.19. To convert the binary number $0.1100\ 1001\ 0110\ 1_2$ to hexadecimal form we note from table 3.1 that

$$1100_2 = c_{16}, \quad 1001_2 = 9_{16}, \quad 0110_2 = 6_{16}, \quad 1000_2 = 8_{16}.$$

Note that the last group of binary digits was not complete so we added three zeros. From this we conclude that

$$0.1100\ 1001\ 0110\ 1_2 = 0.c968_{16}. \quad \blacksquare$$

3.3.4 Properties of Fractional Numbers in Base-β

Real numbers in the interval $(0,1)$ have some interesting properties related to their representation. In the decimal numeral system we know that fractions with a denominator that only contains the factors 2 and 5 can be written as a decimal number with a finite number of digits. In general, the decimal representation of a rational number will contain a finite sequence of digits that are repeated infinitely many times, while for an irrational number there will be no such structure. In this section we shall see that similar properties are valid when fractional numbers are represented in any numeral system.

For rational numbers algorithm 3.16 can be expressed in a different form which makes it easier to deduce properties of the digits. So let us consider what happens when a rational number is converted to base-β representation. A rational number in the interval $(0,1)$ has the form $a = b/c$ where b and c are nonzero natural numbers with $b < c$. If we look at the computations in algorithm 3.16, we note that d_i is the integer part of $(b*\beta)/c$ which can be computed as $(b*\beta)\,//\,c$. The right-hand side of the second statement is $a*\beta - d_1$, i.e., the fractional part of $a*\beta$. But if $a = b/c$, the fractional part of $a*\beta$ is given by the remainder in the division $(b*\beta)/c$, divided by c, so the new value of a is given by

$$a = \frac{(b*\beta)\,\%\,c}{c}.$$

This is a new fraction with the same denominator c as before. But since the denominator does not change, it is sufficient to keep track of the numerator. This can be done by the statement

$$b = (b*\beta)\,\%\,c. \tag{3.9}$$

The result is a new version of algorithm 3.16 for rational numbers.

Algorithm 3.20. *Let $a = b/c$ be a rational number in $(0,1)$ whose first k digits in base β are $(0.d_{-1}d_{-2}\cdots d_{-k})_\beta$. These digits may be computed by performing the operations:*

```
for $i = -1, -2, \ldots, -k$
    $d_i = (b*\beta)\,//\,c$;
    $b = (b*\beta)\,\%\,c$;
```

This version of the conversion algorithm is more convenient for deducing properties of the numerals of a rational number. The clue is to consider more carefully the different values of b that are computed by the algorithm. Since b is the remainder when integers are divided by c, the only possible values of b are $0, 1, 2, \dots, c-1$. Sooner or later, the value of b must therefore become equal to an earlier value. But once b returns to an earlier value, it must cycle through exactly the same values again until it returns to the same value a third time. And then the same values must repeat again, and again, and again, ... Since the numerals d_i are computed from b, they must repeat with the same frequency. Note however that may be some initial digits that do not repeat. This proves part of the following lemma.

Lemma 3.21. *Let a be a fractional number. Then the digits of a written in base β will eventually repeat, i.e.,*

$$a = (0.d_{-1}\cdots d_{-i}d_{-(i+1)}\cdots d_{-(i+m)}d_{-(i+1)}\cdots d_{-(i+m)}\cdots)_\beta$$

for some integer $m \geq 1$ if and only if a is a rational number.

As an example, consider the fraction 1/7 written in different numeral systems. If we run algorithm 3.20 we find

$$1/7 = 0.00100100100100100\cdots_2,$$
$$1/7 = 0.01021201021201021\cdots_3,$$
$$1/7 = 0.1_7.$$

In the binary numeral system, there is no initial sequence of digits; the sequence 001 repeats from the start. In the trinary system, there is no intial sequence either and the repeating sequence is 010212, whereas in the septenary system the initial seqeunce is 1 and the repeating sequence 0 (which we do not write according to the conventions of the decimal system).

An example with an initial sequence is the fraction 87/98 which in base 7 becomes $0.6133333\cdots_7$. Another example is 503/1100 which is $0.457272727272\cdots$ in the decimal system.

The argument preceding lemma 3.21 proves the fact that if a is a rational number, then the digits must eventually repeat. But this statement leaves the possibility open that there may be nonrational (i.e., irrational) numbers that may also have digits that eventually repeat. However, this is not possible and this is the reason for the 'only if'-part of the lemma. In less formal language the complete statement is: *The digits of a will eventually repeat if a is a rational* *number, and only if a is a rational number.* This means that there are two statements to prove: (i) The digits repeat if a is a rational number and (ii) if the digits do repeat then a must be a rational number. The proof of this latter statement is left to excercise 12.

Although all rational numbers have repeating digits, for some numbers the repeating sequence is '0', like 1/7 in base 7, see above. Or equivalently, some fractional numbers can in some numeral systems be represented exactly by a finite number of digits. It is possible to characterise exactly which numbers have this property.

Lemma 3.22. *The representation of a fractional number a in base-β consists of a finite number of digits,*

$$a = (0.d_{-1}d_{-2}\cdots d_{-k})_\beta,$$

if and only if a is a rational number b/c with the property that all the prime factors of c divide β.

Proof. Since the statement is of the 'if and only if' type, there are two claims to be proved. The fact that a fractional number in base-β with a finite number of digits is a rational number is quite straightforward, see exercise 13.

What remains is to prove that if $a = b/c$ and all the prime factors of c divide β, then the representation of a in base-β will have a finite number of digits. We give the proof in a special case and leave it to the reader to write down the proof in general. Let us consider the representation of the number $a = 8/9$ in base-6. The idea of the proof is to rewrite a as a fraction with a power of 6 in the denominator. The simplest way to do this is to observe that $8/9 = 32/36$. We next express 32 in base 6. For this we can use algorithm 3.7, but in this simple situation we see directly that

$$32 = 5 \times 6 + 2 = 52_6.$$

We therefore have

$$\frac{8}{9} = \frac{32}{36} = \frac{5 \times 6 + 2}{6^2} = 5 \times 6^{-1} + 2 \times 6^{-2} = 0.52_6. \quad \blacksquare$$

In the decimal system, fractions with a denominator that only has 2 and 5 as prime factors have finitely many digits, for example $3/8 = 0.375$, $4/25 = 0.16$ and $7/50 = 0.14$. These numbers will *not* have finitely many digits in most other numeral systems. In base-3, the only fractions with finitely many digits are the ones that can be written as fractions with powers of 3 in the denominator,

$$\frac{8}{9} = 0.22_3,$$
$$\frac{7}{27} = 0.021_3,$$
$$\frac{1}{2} = 0.111111111111\cdots_3,$$
$$\frac{3}{10} = 0.02200220022\cdots_3.$$

In base-2, the only fractions that have an exact representation are the ones with denominators that are powers of 2,

$$\frac{1}{2} = 0.5 = 0.1_2,$$
$$\frac{3}{16} = 0.1875 = 0.0011_2,$$
$$\frac{1}{10} = 0.1 = 0.00011001100110011\cdots_2.$$

These are therefore the only fractional numbers that can be represented exactly on most computers unless special software is utilised.

3.4 Arithmetic in Base β

The methods we learn in school for performing arithemetic are closely tied to properties of the decimal numeral system, but the methods can easily be generalised to any numeral system. We are not going to do this in detail, but some examples will illustrate the general ideas. All the methods should be familiar from school, but if you never quite understood the arithmetic methods, you may have to think twice to understand why it all works. Although the methods themselves are the same across the world, it should be remembered that there are many variations in how the methods are expressed on paper. You may therefore find the description given here unfamiliar at first.

3.4.1 Addition

Addition of two one-digit numbers is just like in the decimal system as long as the result has only one digit. For example, we have $4_8 + 3_8 = 4 + 3 = 7 = 7_8$. If the result requires two digits, we must remember that the carry is β in base-β, and not 10. So if the result becomes β or greater, the result will have two digits, where the left-most digit is 1 and the second has the value of the sum, reduced by β. This means that

$$5_8 + 6_8 = 5 + 6 = 11 = 8 + 11 - 8 = 8 + 3 = 13_8.$$

This can be written exactly the same way as you would write a sum in the decimal numeral system, you must just remember that the value of the carry is β.

Let us now try the larger sum $457_8 + 325_8$. This requires successive one-digit additions, just like in the decimal system. One way to write this is

$$\begin{array}{r} {\scriptstyle 1\,1\;\;} \\ 457_8 \\ +325_8 \\ \hline = 1004_8 \end{array}$$

This corresponds to the decimal sum $303 + 213 = 516$.

3.4.2 Subtraction

One-digit subtractions are simple, for example $7_8 - 3_8 = 4_8$. A subtraction like $14_8 - 7_8$ is a bit more difficult, but we can 'borrow' from the '1' in 14 just like in the decimal system. The only difference is that in base-8, the '1' represents 8 and not 10, so we borrow 8. We then see that we must perform the subtraction $12 - 7$ so the answer is 5 (both in decimal and base 8). Subtraction of larger numbers can be done by repeating this. Consider for example $321_8 - 177_8$. This can be written

$$\begin{array}{r} {\scriptstyle 8\,8\;\;\;} \\ \cancel{3}\cancel{2}1_8 \\ -177_8 \\ \hline = 122_8 \end{array}$$

By converting everything to decimal, it is easy to see that this is correct.

3.4.3 Multiplication

Let us just consider one example of a multiplication, namely $312_4 \times 12_4$. As in the decimal system, the basis for performing multiplication of numbers with multiple digits is the multiplication table for one-digit numbers. In base 4 the multiplication table is

	1	2	3
1	1	2	3
2	2	10	12
3	3	12	21

We can then perform the multiplication as we are used to in the decimal system

$$\begin{array}{r}
312_4 \times 12_4 \\
\hline
1230_4 \qquad \\
312_4 \qquad\quad \\
\hline
11010_4 \qquad
\end{array}$$

The number 1230_4 in the second line is the result of the multiplication $312_4 \times 2_4$, i.e., the first factor 312_4 multiplied by the second digit of the right-most factor 12_4. The number on the line below, 312_4, is the first factor multiplied by the first digit of the second factor. This second product is shifted one place to the left since multiplying with the first digit in 12_4 corresponds multiplication by 1×4. The number on the last line is the sum of the two numbers above, with a zero added at the right end of 312_4, i.e., the sum is $1230_4 + 3120_4$. This sum is calculated as indicated in section 3.4.1 above.

Exercises

3.1 Convert the following natural numbers:

- **a)** 40 to base-4
- **b)** 17 to base-5
- **c)** 17 to base-2
- **d)** 123456 to base-7
- **e)** 22875 to base-7
- **f)** 126 to base 16

3.2 Convert the following rational numbers:

- **a)** 1/4 to base-2
- **b)** 3/7 to base-3
- **c)** 1/9 to base-3
- **d)** 1/18 to base-3
- **e)** 7/8 to base-8
- **f)** 7/8 to base-7
- **g)** 7/8 to base-16
- **h)** 5/16 to base-8
- **i)** 5/8 to base-6

3.3 Convert π to base-9.

3.4 Convert to base-8:

- **a)** 1011001_2
- **b)** 110111_2

c) 10101010_2

3.5 Convert to base-2:

a) 44_8
b) 100_8
c) 327_8

3.6 Convert to base-16:

a) 1001101_2
b) 1100_2
c) 10100111100100_2
d) 0.0101100101_2
e) 0.00000101001_2
f) 0.111111111_2

3.7 Convert to base-2:

a) $3c_{16}$
b) 100_{16}
c) $e51_{16}$
d) $0.0aa_{16}$
e) 0.001_{16}
f) $0.f01_{16}$

3.8 **a)** Convert 7 to base-7, 37 to base-37, and 4 to base-4 and formulate a generalisation of what you observe.

b) Determine β such that $13 = 10_\beta$. Also determine β such that $100 = 10_\beta$ For which numbers $a \in \mathbb{N}$ is there a β such that $a = 10_\beta$?

3.9 **a)** Convert 400 to base-20, 4 to base-2, 64 to base-8, 289 to base-17 and formulate a generalisation of what you observe.

b) Determine β such that $25 = 100_\beta$. Also determine β such that $841 = 100_\beta$. For which numbers $a \in \mathbb{N}$ is there a number β such that $a = 100_\beta$?

c) For which numbers $a \in \mathbb{N}$ is there a number β such that $a = 1000_\beta$?

3.10 **a)** For which value of β is $a = b/c = 0.b_\beta$? Does such a β exist for all $a < 1$? And for $a \geq 1$?

b) For which rational number $a = b/c$ does there exist a β such that $a = b/c = 0.01_\beta$?

c) For which rational number $a = b/c$ is there a β such that $a = b/c = 0.0b_\beta$? If β exists, what will it be?

3.11 If $a = b/c$, what is the maximum length of the repeating sequence?

3.12 Show that if the digits of the fractional number a eventually repeat, then a must be a rational number.

3.13 Show that a fractional numbers in base-β with a finite number of digits is a rational number.

3.14 Perform the following additions:

- **a)** $3_7 + 1_7$
- **b)** $5_6 + 4_6$
- **c)** $110_2 + 1011_2$
- **d)** $122_3 + 201_3$
- **e)** $43_5 + 10_5$
- **f)** $3_5 + 1_7$

3.15 Perform the following subtractions:

- **a)** $5_8 - 2_8$
- **b)** $100_2 - 1_2$
- **c)** $527_8 - 333_8$
- **d)** $210_3 - 21_3$
- **e)** $43_5 - 14_5$
- **f)** $3_7 - 11_7$

3.16 Perform the following multiplications:

- **a)** $110_2 \cdot 10_2$
- **b)** $110_2 \cdot 11_2$
- **c)** $110_3 \cdot 11_3$
- **d)** $43_5 \cdot 2_5$
- **e)** $720_8 \cdot 15_8$
- **f)** $210_3 \cdot 12_3$
- **g)** $101_2 \cdot 11_2$