

Mathematics 220 — Fall 2000

Base representation of integers

We write numbers in **decimal** notation. For example, we write the number ‘one thousand’ as ‘1000’, which means that it is equal to 10^3 . More generally, an expression like ‘4567’ refers to a number which is expressed in terms of powers of 10:

$$4 \cdot 10^3 + 5 \cdot 10^2 + 6 \cdot 10^1 + 7 \cdot 10^0 .$$

Decimal notation is based on the number 10, but, as much linguistic evidence shows, this dependence is an historical and biological accident deriving from the number of fingers we have on our hands. We can work with more general bases. Indeed, Babylonian astronomers did arithmetic in base 60 long before the Hindu, Arabs, and Europeans did arithmetic in base 10. The main fact about bases for arithmetic is this:

- If b is any positive integer, then any other positive integer n can be written as a sum of terms

$$n = d_{m-1}b^{m-1} + \cdots + d_1b^1 + d_0b^0$$

where each d_i is a positive integer in the range $0 \leq d_i < b$.

With base b , the numbers in the range $[0, b - 1]$ replace the usual decimal digits in the range $[0, 9]$, and this equation is called the representation of n in base b . We are going to prove this claim by explaining explicitly how to find such a representation, but we first need to recall what long division of integers accomplishes.

- If n and m are any two integers with $m > 0$, then when we divide n by m we obtain a **quotient** q and a **remainder** r such that

$$n = qm + r, \quad 0 \leq r < m .$$

These properties characterize q and r uniquely.

We shall see later on how to find q and r explicitly, pretty much following what you learned in grammar school, but for the moment it is the uniqueness that we want be assured of. If we have

$$n = q_1m + r_1$$

$$n = q_2m + r_2$$

with $0 \leq r_1, r_2 \leq m$, then by subtracting we get

$$m(q_2 - q_1) = r_1 - r_2 .$$

This means that the difference $r_1 - r_2$ is both a multiple of m and in the range $[-(m - 2), (m - 2)]$. Impossible unless $r_1 - r_2 = 0$, or $r_1 = r_2$.

In other words, *we cannot have two different ways of doing long division that are both correct, unless they give the same result.*

The basic facts about representation of a number n in base b is are these: (1) the 0-th order ‘digit’ d_0 of n in base b is the remainder upon dividing n by b ; (2) the number $d_{m-1}b^{m-2} + \cdots + d_1b^0$ is the quotient you get when you divide n by b .

This is because

$$d_{m-1}b^{m-1} + \cdots + d_1b^1 + d_0b^0 = (d_{m-1}b^{m-2} + \cdots + d_1b^0)b + d_0, \quad 0 \leq d_0 < b .$$

In other words, the 0-th order digit of n in base b is the remainder upon dividing n by b ; and the digits d_{m-1}, \dots, d_1 make up the base b representation of the quotient of n by b . So the way to find the base b representation of n goes like this: (1) Divide n by b , getting quotient q and remainder r . Then $d_0 = r$. (2) If $q < b$ we can stop, because it is just a single digit. Otherwise, replace n by the quotient q , and loop through the first step again.

This process of repeating the same process over and over again but with smaller data each time is called **recursion**. Using recursion most directly, we have these rules for writing n in base b : (1) If $n < b$, write down n itself. (2) Otherwise, divide n by b . The base b representation of n is the base b representation of the quotient followed by the remainder.

Example. Let's find the base 13 representation of the number which is expressed as 1457 in decimal notation.

We divide 1457 by 13. Quotient 112, remainder 1.

Divide 112 by 13. Quotient 8, remainder 8.

Now we have $n = 8$. But this is less than 13, so we can stop.

The successive remainders we got are 1, 8, and 8, so we should have

$$1457 = 8 \cdot 13^2 + 8 \cdot 13 + 1$$

which we can check easily. We can also express this in somewhat awkward notation as

$$[1457]_{13} = [8][8][1].$$

In other words, in base 13 we will take the digits (which are really **representations** of numbers in the range $[0, 12]$, not the numbers themselves) to be $[0], [1], \dots, [12]$. We have to do this whenever the base is larger than 10, since otherwise expressions would be ambiguous. For example, unless we put separators in, we could not tell whether 121 in base 13 meant $12 \cdot 13 + 1$ or $1 \cdot 13^2 + 2 \cdot 13 + 1$.

Base 16 is rather special. Here the 'digits' 11 to 15 are usually represented by letters 'a', ..., 'f'. Base 16 expressions are used frequently in computer work, where they are prefaced by '0x' to tell you what's coming. Thus

$$0xa6 = 10 \cdot 16 + 6.$$

Numbers are represented internally by computers in base 2, but expressing a number of any reasonable size in base 2 involves reading or writing a huge number of confusing 0's and 1's. Using base 16 is a nice compromise between machine and human requirements, since translating back and forth in between base 16 and base 2 is very easy, and because 16 is close enough to 10 to give us some intuitive feel for size of base 16 numbers.