2.2 Computer Operations with Integers

Before computers were invented, mathematicians did computations either by hand or by using mechanical devices. Either way, they were only able to work with integers of rather limited size. Many number theoretic problems, such as factoring and primality testing, require computations with integers of as many as 100 or even 200 digits. In this section, we will study some of the basic algorithms for doing computer arithmetic. In the following section, we will study the number of basic computer operations required to carry out these algorithms.

We have mentioned that computers internally represent numbers using bits, or binary digits. Computers have a built-in limit on the size of integers that can be used in machine arithmetic. This upper limit is called the *word size*, which we denote by w. The word size is usually a power of 2, such as 2^{32} for Pentium machines or 2^{35}, although sometimes the word size is a power of 10.

To do arithmetic with integers larger than the word size, it is necessary to devote more than one word to each integer. To store an integer $n > w$, we express n in base w notation, and for each digit of this expansion we use one computer word. For instance, if the word size is 2^{35}, using ten computer words we can store integers as large as $2^{350} - 1$, since integers less than 2^{350} have no more than ten digits in their base 2^{35} expansions. Also note that to find the base 2^{35} expansion of an integer, we need only group together blocks of 35 bits.

The first step in discussing computer arithmetic with large integers is to describe how the basic arithmetic operations are methodically performed.

We will describe the classical methods for performing the basic arithmetic operations with integers in base r notation, where $r > 1$ is an integer. These methods are examples of *algorithms*.

Definition. An *algorithm* is a finite set of precise instructions for performing a computation or for solving a problem.

We will describe algorithms for performing addition, subtraction, and multiplication of two n-digit integers $a = (a_{n-1}a_{n-2} \ldots a_1a_0)_r$ and $b = (b_{n-1}b_{n-2} \ldots b_1b_0)_r$, where initial digits of zero are added if necessary to make both expansions the same length. The algorithms described are used for both binary arithmetic with integers less than the word size of a computer, and *multiple precision* arithmetic with integers larger than the word size w, using w as the base.

Addition When we add a and b, we obtain the sum

$$a + b = \sum_{j=0}^{n-1} a_j r^j + \sum_{j=0}^{n-1} b_j r^j = \sum_{j=0}^{n-1} (a_j + b_j) r^j.$$

To find the base r expansion of $a + b$, first note that by the division algorithm, there are integers C_0 and s_0 such that

$$a_0 + b_0 = C_0 r + s_0, \quad 0 \leq s_0 < r.$$

Because a_0 and b_0 are positive integers not exceeding r, we know that $0 \leq a_0 + b_0 \leq 2r - 2$, so that $C_0 = 0$ or 1; here, C_0 is the *carry* to the next place. Next, we find that there are integers C_1 and s_1 such that

$$a_1 + b_1 + C_0 = C_1 r + s_1, \quad 0 \leq s_1 < r.$$

Since $0 \leq a_1 + b_1 + C_0 \leq 2r - 1$, we know that $C_1 = 0$ or 1. Proceeding inductively, we find integers C_i and s_i for $1 \leq i \leq n - 1$ by

$$a_i + b_i + C_{i-1} = C_i r + s_i, \quad 0 \leq s_i < r,$$

with $C_i = 0$ or 1. Finally, we let $s_n = C_{n-1}$, since the sum of two integers with n digits has $n + 1$ digits when there is a carry in the nth place. We conclude that the base r expansion for the sum is $a + b = (s_n s_{n-1} \ldots s_1 s_0)_r$.

When performing base r addition by hand, we can use the same familiar technique as is used in decimal addition.

Example 2.5. To add $(1101)_2$ and $(1001)_2$, we write

$$\begin{array}{cccccc} & \textit{1} & & \textit{1} & & \\ & 1 & 1 & 0 & 1 & \\ + & 1 & 0 & 0 & 1 & \\ \hline 1 & 0 & 1 & 1 & 0 & \end{array}$$

where we have indicated carries by 1s in italics written above the appropriate column. We found the binary digits of the sum by noting that $1 + 1 = 1 \cdot 2 + 0, 0 + 0 + 1 = 0 \cdot 2 + 1, 1 + 0 + 0 = 0 \cdot 2 + 1$, and $1 + 1 + 0 = 1 \cdot 2 + 0$. ◀

Subtraction Assume that $a > b$. Consider

$$a - b = \sum_{j=0}^{n-1} a_j r^j - \sum_{j=0}^{n-1} b_j r^j = \sum_{j=0}^{n-1} (a_j - b_j) r^j.$$

Note that by the division algorithm, there are integers B_0 and d_0 such that

$$a_0 - b_0 = B_0 r + d_0, \quad 0 \leq d_0 < r,$$

Where the Word "Algorithm" Comes from

"Algorithm" is a corruption of the original term "algorism," which originally comes from the name of the author of the ninth-century book *Kitab al-jabr w'al-muqabala (Rules of Restoration and Reduction), Abu Ja'far Mohammed ibn Mûsâ al-Khwârizmî* (see his biography included on the next page). The word "algorism" originally referred only to the rules of performing arithmetic using Hindu-Arabic numerals, but evolved into "algorithm" by the eighteenth century. With growing interest in computing machines, the concept of an algorithm became more general, to include all definite procedures for solving problems, not just the procedures for performing arithmetic with integers expressed in Arabic notation.

and because a_0 and b_0 are positive integers less than r, we have

$$-(r-1) \leq a_0 - b_0 \leq r-1.$$

When $a_0 - b_0 \geq 0$, we have $B_0 = 0$. Otherwise, when $a_0 - b_0 < 0$, we have $B_0 = -1$; B_0 is the *borrow* from the next place of the base r expansion of a. We use the division algorithm again to find integers B_1 and d_1 such that

$$a_1 - b_1 + B_0 = B_1 r + d_1, \quad 0 \leq d_1 < r.$$

From this equation, we see that the borrow $B_1 = 0$ as long as $a_1 - b_1 + B_0 \geq 0$, and that $B_1 = -1$ otherwise, because $-r \leq a_1 - b_1 + B_0 \leq r-1$. We proceed inductively to find integers B_i and d_i, such that

$$a_i - b_i + B_{i-1} = B_i r + d_i, \quad 0 \leq d_i < r$$

with $B_i = 0$ or -1, for $1 \leq i \leq n-1$. We see that $B_{n-1} = 0$, because $a > b$. We can conclude that

$$a - b = (d_{n-1} d_{n-2} \ldots d_1 d_0)_r.$$

When performing base r subtraction by hand, we use the familiar technique used in decimal subtraction.

Example 2.6. To subtract $(10110)_2$ from $(11011)_2$, we have

$$\begin{array}{cccccc} & & \textit{-1} & & & \\ 1 & 1 & 0 & 1 & 1 & \\ -1 & 0 & 1 & 1 & 0 & \\ \hline & & & 1 & 0 & 1 \end{array}$$

where the -1 in italics above a column indicates a borrow. We found the binary digits of the difference by noting that $1-0 = 0 \cdot 2 + 1$, $1 - 1 + 0 = 0 \cdot 2 + 0$, $0 - 1 + 0 = -1 \cdot 2 + 1$, $1 - 0 - 1 = 0 \cdot 2 + 0$, and $1 - 1 + 0 = 0 \cdot 2 + 0$. ◀

ABU JA'FAR MOHAMMED IBN MÛSÂ AL-KHWÂRIZMÎ (c. 780–c. 850), an astronomer and mathematician, was a member of the House of Wisdom, an academy of scientists in Baghdad. The name al-Khwârizmî means "from the town of Kowarzizm," now known as Khiva in modern Uzbekistan. Al-Khwârizmî was the author of books on mathematics, astronomy, and geography. People in the West first learned about algebra from his works; the word "algebra" comes from *al-jabr*, part of the title of his book *Kitab al-jabr w'al muqabala*, which was translated into Latin and widely used as a text. Another book describes procedures for arithmetic operations using Hindu-Arabic numerals.

Multiplication Before discussing multiplication, we describe *shifting*. To multiply $(a_{n-1} \ldots a_1 a_0)_r$ by r^m, we need only shift the expansion left m places, appending the expansion with m zero digits.

Example 2.7. To multiply $(101101)_2$ by 2^5, we shift the digits to the left five places and append the expansion with five zeros, obtaining $(10110100000)_2$. ◀

We first discuss the multiplication of an n-place integer by a one-digit integer. To multiply $(a_{n-1} \ldots a_1 a_0)_r$ by $(b)_r$, we first note that

$$a_0 b = q_0 r + p_0, \quad 0 \leq p_0 < r,$$

and $0 \leq q_0 \leq r - 2$, because $0 \leq a_0 b \leq (r-1)^2$. Next, we have

$$a_1 b + q_0 = q_1 r + p_1, \quad 0 \leq p_1 < r,$$

and $0 \leq q_1 \leq r - 1$. In general, we have

$$a_i b + q_{i-1} = q_i r + p_i, \quad 0 \leq p_i < r,$$

and $0 \leq q_i \leq r - 1$. Furthermore, we have $p_n = q_{n-1}$. This yields $(a_{n-1} \ldots a_1 a_0)_r (b)_r = (p_n p_{n-1} \cdots p_1 p_0)_r$.

To perform a multiplication of two n-place integers, we write

$$ab = a\left(\sum_{j=0}^{n-1} b_j r^j\right) = \sum_{j=0}^{n-1} (a b_j) r^j.$$

For each j, we first multiply a by the digit b_j, then shift j places to the left, and finally add all of the n integers we have obtained to find the product.

When multiplying two integers with base r expansions, we use the familiar method of multiplying decimal integers by hand.

Example 2.8. To multiply $(1101)_2$ and $(1110)_2$, we write

$$\begin{array}{r} 1\ 1\ 0\ 1 \\ \times\ 1\ 1\ 1\ 0 \\ \hline 0\ 0\ 0\ 0 \\ 1\ 1\ 0\ 1 \\ 1\ 1\ 0\ 1 \\ 1\ 1\ 0\ 1 \\ \hline 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0 \end{array}$$

Note that we first multiplied $(1101)_2$ by each digit of $(1110)_2$, shifting each time by the appropriate number of places, and then we added the appropriate integers to find our product. ◀

Division We wish to find the quotient q in the division algorithm

$$a = bq + R, \quad 0 \leq R < b.$$

If the base r expansion of q is $q = (q_{n-1}q_{n-2} \cdots q_1q_0)_r$, then we have

$$a = b\left(\sum_{j=0}^{n-1} q_j r^j\right) + R, \quad 0 \leq R < b.$$

To determine the first digit q_{n-1} of q, notice that

$$a - bq_{n-1}r^{n-1} = b\left(\sum_{j=0}^{n-2} q_j r^j\right) + R.$$

The right-hand side of this equation is not only positive, but also less than br^{n-1}, because $\sum_{j=0}^{n-2} q_j r^j \leq \sum_{j=0}^{n-2}(r-1)r^j = \sum_{j=1}^{n-1} r^j - \sum_{j=0}^{n-2} r^j = r^{n-1} - 1$. Therefore, we know that

$$0 \leq a - bq_{n-1}r^{n-1} < br^{n-1}.$$

This tells us that

$$q_{n-1} = \left\lfloor \frac{a}{br^{n-1}} \right\rfloor.$$

We can obtain q_{n-1} by successively subtracting br^{n-1} from a until we obtain a negative result; q_{n-1} is then one less than the number of subtractions.

To find the other digits of q, we define the sequence of *partial remainders* R_i by

$$R_0 = a$$

and

$$R_i = R_{i-1} - bq_{n-i}r^{n-i}$$

for $i = 1, 2, \ldots, n$. By mathematical induction, we show that

$$R_i = \left(\sum_{j=0}^{n-i-1} q_j r^j\right)b + R. \tag{2.1}$$

For $i = 0$, this is clearly correct, because $R_0 = a = qb + R$. Now, assume that

$$R_k = \left(\sum_{j=0}^{n-k-1} q_j r^j\right)b + R.$$

Then

$$\begin{aligned} R_{k+1} &= R_k - bq_{n-k-1}r^{n-k-1} \\ &= \left(\sum_{j=0}^{n-k-1} q_j r^j\right) b + R - bq_{n-k-1}r^{n-k-1} \\ &= \left(\sum_{j=0}^{n-(k+1)-1} q_j r^j\right) b + R, \end{aligned}$$

establishing (2.1).

By (2.1) we see that $0 \leq R_i < r^{n-i}b$, for $i = 1, 2, \ldots, n$, because $\sum_{j=0}^{n-i-1} q_j r^j \leq r_{n-i} - 1$. Consequently, because $R_i = R_{i-1} - bq_{n-i}r^{n-i}$ and $0 \leq R_i < r^{n-1}b$, we see that the digit q_{n-i} is given by $[R_{i-1}/(br^{n-i})]$ and can be obtained by successively subtracting br^{n-i} from R_{i-1} until a negative result is obtained, and then q_{n-i} is one less than the number of subtractions. This is how we find the digits of q.

Example 2.9. To divide $(11101)_2$ by $(111)_2$, we let $q = (q_2q_1q_0)_2$. We subtract $2^2(111)_2 = (11100)_2$ once from $(11101)_2$ to obtain $(1)_2$, and once more to obtain a negative result, so that $q_2 = 1$. Now, $R_1 = (11101)_2 - (11100)_2 = (1)_2$. We find that $q_1 = 0$, because $R_1 - 2(111)_2$ is less than zero, and likewise $q_0 = 0$. Hence, the quotient of the division is $(100)_2$ and the remainder is $(1)_2$. ◀

2.2 Exercises

1. Add $(101111011)_2$ and $(110011011)_2$.
2. Add $(10001000111101)_2$ and $(11111101011111)_2$.
3. Subtract $(11010111)_2$ from $(1111000011)_2$.
4. Subtract $(101110101)_2$ from $(1101101100)_2$.
5. Multiply $(11101)_2$ and $(11001)_2$.
6. Multiply $(1110111)_2$ and $(1001101)_2$.
7. Find the quotient and remainder when $(110011111)_2$ is divided by $(1101)_2$.
8. Find the quotient and remainder when $(110100111)_2$ is divided by $(11101)_2$.
9. Add $(1234321)_5$ and $(2030104)_5$.
10. Subtract $(434421)_5$ from $(4434201)_5$.
11. Multiply $(1234)_5$ and $(3002)_5$.
12. Find the quotient and remainder when $(14321)_5$ is divided by $(334)_5$.
13. Add $(ABAB)_{16}$ and $(BABA)_{16}$.
14. Subtract $(CAFE)_{16}$ from $(FEED)_{16}$.
15. Multiply $(FACE)_{16}$ and $(BAD)_{16}$.
16. Find the quotient and remainder when $(BEADED)_{16}$ is divided by $(ABBA)_{16}$.