Symbolic Computation 5:

HOMOMORPHISMS AND CHINESE REMAINDER ALGORITHMS

by

Keith O. Geddes
Department of Computer Science
University of Waterloo
Waterloo, Ontario, Canada
N2L 3G1

Research Report CS-81-35
December 1981

PREFACE

This report consists of Chapter 5 of a textbook being written under the title 'Algebraic Algorithms for Symbolic Computation' by Keith O. Geddes. The table of contents for Chapter 5 appears below. A more complete table of contents for the textbook appears on the following pages.

ALGEBRAIC ALGORITHMS FOR SYMBOLIC COMPUTATION

KEITH O. GEDDES

Department of Computer Science
University of Waterloo

CONTENTS

5. HOMOMORPHISMS AND CHINESE REMAINDER ALGORITHMS

An important concept in algebraic ring theory is that of a homomorphism which maps one ring into another. This concept has been found to be an indispensible tool in the development of efficient algorithms for various problems in symbolic computation. The basic approach in these algorithms is to map a given domain onto one or more simpler domains (called homomorphic images), perform the desired computations in the simpler domains, and finally to reconstruct the result in the original domain from the results obtained in the simpler domains. In this chapter the algebraic theory of homomorphisms is introduced and then integer and polynomial Chinese remainder algorithms are developed for performing the above-mentioned 'reconstruction'.

5.1. RING MORPHISMS

In this section we introduce the concept of mapping an algebraic system onto a simpler 'model' of itself and, alternatively, the concept of embedding an algebraic system into a larger algebraic system. A related concept is that of an isomorphism between two algebraic systems which we have already encountered in chapter 2. It is convenient to adopt the terminology of universal algebra when discussing these concepts.

Subalgebras

By an *algebra* (or *algebraic system*) we understand a set S together with a collection of operations defined on S. Specifically for our purposes, by an algebra we shall mean any one of the following types of *rings:*

commutative ring
integral domain
unique factorization domain (UFD)
Euclidean domain
field .

Thus the operations in the algebras we are considering are the *binary* operations of addition and multiplication, the *unary* operation of negation, and the *nullary* operations of 'select 0' and 'select 1'.[6] In addition, if the algebra is a field then there is also the unary operation of inversion which maps each element (except 0) onto its multiplicative inverse. When the collection of operations is implied by context, we often refer to the algebra S when what we mean is the algebra consisting of the set S together with the operations defined on S.

6. A binary operation takes two operands and a unary operation takes one operand. Similarly, a nullary operation takes no operands and is simply a *selection function* – in our case, the operations of selecting the additive identity 0 and the multiplicative identity 1.

Definition 5.1.

Let S be an algebra. A subset S' of the set S is called a *subalgebra* if S' is closed under the operations defined on S. □

We use the following terminology for subalgebras of the specific algebras listed above. If S is a (commutative) ring then a subalgebra of S is called a *subring*. If S is an integral domain, UFD, or a Euclidean domain then a subalgebra of S is called a *subdomain*. If S is a field then a subalgebra of S is called a *subfield*. For any algebra S, it is clear that if a subset S' is closed under all of the operations defined on S then all of the axioms which hold in S are automatically inherited by the subalgebra S'. In particular, a subring of a commutative ring is itself a commutative ring, a subdomain of an integral domain (UFD, Euclidean domain) is itself an integral domain (UFD, Euclidean domain), and a subfield of a field is itself a field.

Example 5.1.

In Figure 2.2 of chapter 2, if two domains S and R are related by the notation $S \rightarrow R$ then S is a subdomain of R. For example, $D[x]$ and $D[[x]]$ are subdomains of $F_D((x))$. Also, $D(x)$ and $D((x))$ are subfields of $F_D((x))$. □

Morphisms

In discussing mappings between two rings R and R' we will adopt the convention of using the same notation to denote the operations in R and in R' Thus + will denote addition in R or in R', depending on context, multiplication in both R and R' will be denoted by juxtaposition without any operator symbol, and 0 and 1 will denote (respectively) the additive and multiplicative identities in R or R', depending on context. This convention is particularly appropriate in the common situation where one of R, R' is a subring of the other.

Definition 5.2.

Let R and R' be two rings. Then a mapping $\phi : R \rightarrow R'$ is called a *ring morphism* if

(i) $\phi(a + b) = \phi(a) + \phi(b)$ for all $a,b \in R$;

(ii) $\phi(ab) = \phi(a)\,\phi(b)$ for all $a,b \in R$;

(iii) $\phi(1) = 1$. □

The general (universal algebra) concept of a *morphism* between two algebras is that of a mapping which preserves *all* of the operations defined on the algebras. In Definition 5.2, note that properties (i) - (iii) ensure that three of the ring operations are preserved but that no mention has been made of the unary operation of negation and the nullary operation 'select 0'. This is because the two additional properties:

$$\phi(0) = 0;$$

$$\phi(-a) = -\phi(a) \quad \text{for all } a \in R$$

are simple consequences of the ring axioms and properties (i) - (iii). Similarly, if R and R' are fields with the additional unary operation of inversion then the ring morphism of Definition 5.2 is in fact a *field morphism* because the additional property:

$$\phi(a^{-1}) = [\phi(a)]^{-1} \quad \text{for all } a \in R-\{0\}$$

is a consequence of the field axioms and properties (i) - (iii). Therefore in the sequel when we refer to a morphism it will be understood that we are referring to a ring morphism as defined in Definition 5.2.

Morphisms are classified according to their properties as functions. If $\phi : R \rightarrow R'$ is a morphism then it is called a *monomorphism* if the function ϕ is injective (i.e. one-to-one), an *epimorphism* if the function ϕ is surjective (i.e. onto), and an *isomorphism* if the function ϕ is bijective (i.e. one-to-one and onto). The classical term *homomorphism* in its most general usage is simply a synonym for the more modern term 'morphism' used in the context of universal algebra. However in common usage the term 'homomorphism' is most often identified with an epimorphism and in particular if $\phi : R \rightarrow R'$ is an epimorphism then R' is called a *homomorphic image* of R.

A monomorphism $\phi : R \rightarrow R'$ is called an *embedding* of R into R' since clearly the mapping $\phi : R \rightarrow \phi(R)$ onto the image set

$$\phi(R) = \{r' \in R' : \ \phi(r) = r' \quad \text{for some } r \in R\}$$

is an isomorphism – i.e. the ring R' contains R (more correctly, an isomorphic copy of R) as a subring. An epimorphism $\phi : R \rightarrow R'$ is called a *projection* of R onto the homomorphic image R'. In this terminology, it is clear that for any morphism $\phi : R \rightarrow R'$ the image set $\phi(R)$ is a homomorphic image of R. An important property of morphisms is that a homomorphic image of a (commutative) ring is itself a (commutative) ring. However, a homomorphic image of an integral domain is not necessarily an integral domain (see Example 5.4).

Example 5.2.

Several instances of isomorphic algebras were encountered in chapter 2. For any commutative ring R, the polynomial domains $R[x,y]$, $R[x][y]$, and $R[y][x]$ are isomorphic; for example, the natural mapping

$$\phi : R[x,y] \rightarrow R[x][y]$$

defined by

$$\phi\left(\sum_{i=0}^{m}\sum_{j=0}^{n} a_{ij}x^i y^j\right) = \sum_{j=0}^{n}\left(\sum_{i=0}^{m} a_{ij}x^i\right)y^j$$

is an isomorphism. Similarly, for any integral domain D with quotient field F_D the fields of rational functions $D(\mathbf{x})$ and $F_D(\mathbf{x})$ are isomorphic with a natural mapping between them. Also, for any field F the fields $F((x))$ and $F\langle x\rangle$ are isomorphic with a natural mapping from the canonical form of a power series rational function in $F((x))$ onto an extended power series in $F\langle x\rangle$. □

Example 5.3.

Let D be an integral domain and let F_D be its quotient field. The mapping

$$\phi : D \rightarrow F_D$$

defined by

$$\phi(a) = [a/1] \quad \text{for all } a \in D$$

is a monomorphism. Thus ϕ is an embedding of D into F_D and we call F_D an *extension* of D (the smallest extension of the integral domain D into a field). □

Example 5.4.

Let $\mathbf{Z}$ be the integers and let $\mathbf{Z}_6$ be the set of integers modulo 6. Let $\phi : \mathbf{Z} \rightarrow \mathbf{Z}_6$ be the mapping defined by

$$\phi(a) = \text{rem}(a, 6) \quad \text{for all } a \in \mathbf{Z}$$

where the remainder function 'rem' is as defined in chapter 2. Then ϕ is an epimorphism and thus ϕ is a projection of $\mathbf{Z}$ onto the homomorphic image $\mathbf{Z}_6$. $\mathbf{Z}_6$ is a commutative ring because $\mathbf{Z}$ is a commutative ring. $\mathbf{Z}_6$ is not an integral domain (see Exercise 2-3) even though $\mathbf{Z}$ is an integral domain. □

Example 5.5.

Let R and R′ be commutative rings with R a subring of R′. Let R[x] be the commutative ring of univariate polynomials over R and let

$$\phi : R[x] \rightarrow R'$$

be the mapping defined by

$$\phi(a(x)) = a(\alpha)$$

for some fixed element $\alpha \in R'$. (i.e. The image of $a(x)$ is obtained by evaluating $a(x)$ at the value $x = \alpha$). Then ϕ is a morphism of rings. □

Modular and Evaluation Homomorphisms

In the sequel the morphisms of interest to us will be projections of a ring R onto (simpler) homomorphic images of R. In keeping with common usage we will use the term 'homomorphism' for such projections. We now consider two particular classes of homomorphisms which have many practical applications in algorithms for symbolic computation.

The first homomorphism of interest is a generalization of the projection of the integers considered in Example 5.4. Formally, a *modular homomorphism*

$$\phi_m : \mathbf{Z}[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_m[x_1, \ldots, x_v]$$

is a homomorphism defined for a fixed integer $m \in \mathbf{Z}$ by the rules:

$$\phi_m(x_i) = x_i, \quad \text{for } 1 \leqslant i \leqslant v;$$

$$\phi_m(a) = \text{rem}(a,m), \quad \text{for all coefficients } a \in \mathbf{Z}.$$

In other words, a modular homomorphism ϕ_m is a projection of $\mathbf{Z}[x_1, \ldots, x_v]$ onto $\mathbf{Z}_m[x_1, \ldots, x_v]$ obtained by simply replacing every coefficient of a polynomial $a(\mathbf{x}) \in \mathbf{Z}[x_1, \ldots, x_v]$ by its 'modulo m' representation. Of course ϕ_m remains well-defined in the case $v = 0$ in which case it is simply a projection of $\mathbf{Z}$ onto $\mathbf{Z}_m$.

Example 5.6.

In $\mathbf{Z}[x,y]$ let $a(x,y)$ and $b(x,y)$ be the polynomials

(1) $a(x,y) = 3x^2y^2 - x^2y + 5x^2 + xy^2 - 3xy;$

(2) $b(x,y) = 2xy + 7x + y^2 - 2.$

The modular homomorphism ϕ_5 maps these two polynomials onto the following polynomials in the domain $\mathbf{Z}_5[x,y]$:

$$\phi_5(a(x,y)) = 3x^2y^2 + 4x^2y + xy^2 + 2xy;$$

$$\phi_5(b(x,y)) = 2xy + 2x + y^2 + 3.$$

Similarly, the modular homomorphism ϕ_7 maps (1) - (2) onto the following polynomials in the domain $\mathbf{Z}_7[x,y]$:

$$\phi_7(a(x,y)) = 3x^2y^2 + 6x^2y + 5x^2 + xy^2 + 4xy;$$

$$\phi_7(b(x,y)) = 2xy + y^2 + 5. \quad \square$$

The second homomorphism of interest is a special case of the ring morphism considered in Example 5.5 applied in the context of a multivariate polynomial domain $D[x_1, \ldots, x_v]$. In the notation of Example 5.5, we identify x with a *particular* indeterminate x_i and we choose

$$R = R' = D[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_v]$$

so that

$$R[x] = D[x_1, \ldots, x_v].$$

Formally, an *evaluation homomorphism*

$$\phi_{x_i - \alpha} : D[x_1, \ldots, x_v] \rightarrow D[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_v]$$

is a homomorphism defined for a particular indeterminate x_i and a fixed element $\alpha \in D$ such that for any polynomial $a(x_1, \ldots, x_v) \in D[x_1, \ldots, x_v]$,

$$\phi_{x_i - \alpha}(a(x_1, \ldots, x_v)) = a(x_1, \ldots, x_{i-1}, \alpha, x_{i+1}, \ldots, x_v).$$

In other words, an evaluation homomorphism $\phi_{x_i - \alpha}$ is a projection of $D[x_1, \ldots, x_v]$ onto $D[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_v]$ obtained by simply substituting the value $\alpha \in D$ for the i-th indeterminate x_i. Thus the notation $\phi_{x_i - \alpha}$ can be read 'substitute α for x_i'. (The particular choice of notation $\phi_{x_i - \alpha}$ for an evaluation homomorphism is such that the subscript $x_i - \alpha$ corresponds to the subscript m in the notation ϕ_m for a modular homomorphism. The reason for this correspondence of notation will become clear in a later section).

Compositions of modular and evaluation homomorphisms will be used frequently in later chapters for projecting the multivariate polynomial domain $\mathbf{Z}[x_1, \ldots, x_v]$ onto simpler homomorphic images of itself. In most such applications a modular homomorphism ϕ_p, where p is a positive prime integer, will be chosen to project $\mathbf{Z}[x_1, \ldots, x_v]$ onto $\mathbf{Z}_p[x_1, \ldots, x_v]$ where the coefficient domain Z_p is now a field. A sequence of evaluation homomorphisms (one for each indeterminate) can then be applied to project the multivariate polynomial domain $\mathbf{Z}_p[x_1, \ldots, x_v]$ onto a homomorphic image of the form $\mathbf{Z}_p[x_1]$ (a Euclidean domain) or, if desired, onto a homomorphic image of the form $\mathbf{Z}_p$ (a field). It will be seen in later chapters that for the problem of GCD computation in $\mathbf{Z}[x_1, \ldots, x_v]$, and also for the problem of polynomial factorization in $\mathbf{Z}[x_1, \ldots, x_v]$, very efficient algorithms can be obtained by projecting to homomorphic images of the form $\mathbf{Z}_p[x_1]$ where the ordinary Euclidean algorithm applies. The following example considers the more elementary problem of polynomial multiplication in which case projections onto fields $\mathbf{Z}_p$ are appropriate.

Example 5.7.

In the domain $\mathbf{Z}[x]$ let

$$a(x) = 7x + 5;$$

$$b(x) = 2x - 3.$$

Suppose we wish to determine the product polynomial

$$c(x) = a(x)\,b(x).$$

Rather than directly multiplying these polynomials in the domain $\mathbf{Z}[x]$ we could choose to project $\mathbf{Z}[x]$ onto homomorphic images $\mathbf{Z}_p$ and perform the (simpler) multiplications in the fields $\mathbf{Z}_p$. For example, the composite homomorphism

$$\phi_{x-0}\,\phi_5 : \mathbf{Z}[x] \rightarrow \mathbf{Z}_5$$

maps $a(x)$ and $b(x)$ as follows:

$$a(x) \xrightarrow{\phi_5} 2x \xrightarrow{\phi_{x-0}} 0\,;$$

$$b(x) \xrightarrow{\phi_5} 2x + 2 \xrightarrow{\phi_{x-0}} 2\,.$$

Thus the product in this particular homomorphic image $\mathbf{Z}_5$ is $0 \times 2 = 0$. Using standard congruence notation for 'mod p' arithmetic we represent this as follows:

$$c(0) \equiv 0 \pmod 5.$$

Similarly, applying the composite homomorphism $\phi_{x-1}\phi_5$ yields:

$$a(x) \xrightarrow{\phi_5} 2x \xrightarrow{\phi_{x-1}} 2;$$

$$b(x) \xrightarrow{\phi_5} 2x + 2 \xrightarrow{\phi_{x-1}} 4.$$

This time the product in $\mathbf{Z}_5$ is $2 \times 4 = 3$. Thus,

$$c(1) \equiv 3 \pmod 5.$$

Similarly, we find

$$c(2) \equiv 4 \pmod 5$$

by applying the composite homomorphism $\phi_{x-2}\phi_5$. If in addition we apply the triple of composite homomorphisms:

$$\phi_{x-0}\phi_7 ; \ \phi_{x-1}\phi_7 ; \ \phi_{x-2}\phi_7$$

each of which projects $\mathbf{Z}[x]$ onto $\mathbf{Z}_7$, we get

$$c(0) \equiv 6 \pmod 7;$$
$$c(1) \equiv 2 \pmod 7;$$
$$c(2) \equiv 5 \pmod 7.$$

The above process is only useful if we can 'invert' the homomorphisms to reconstruct the polynomial $c(x) \in \mathbf{Z}[x]$, given information about the images of $c(x)$ in fields $\mathbf{Z}_p$. The inverse process involves the concepts of interpolation and Chinese remaindering which will be discussed later in this chapter. Briefly, since we know that $\deg[c(x)] = \deg[a(x)] + \deg[b(x)] = 2$, the polynomial $c(x)$ is completely specified by its values at 3 points. Using the above information, we obtain by interpolation:

$$c(x) \equiv 4x^2 + 4x \pmod 5;$$

$$c(x) \equiv 3x + 6 \pmod 7.$$

Thus we know the images of $c(x)$ in $\mathbf{Z}_5[x]$ and in $\mathbf{Z}_7[x]$. Finally, we can determine

$$c(x) = c_2x^2 + c_1x + c_0 \in \mathbf{Z}[x]$$

by a process known as Chinese remaindering. For example, since we know that

$$c_2 \equiv 4 \pmod 5 \quad \text{and} \quad c_2 \equiv 0 \pmod 7$$

we can determine that

$$c_2 \equiv 14 \pmod{35}$$

(where $35 = 5 \times 7$). We eventually get:

$$c(x) = 14x^2 - 11x - 15 \in \mathbf{Z}[x]. \quad \square$$

5.2. CHARACTERIZATION OF MORPHISMS

Ideals

A ring morphism $\phi : R \rightarrow R'$ can be conveniently characterized in terms of its action on particular subsets of R known as ideals.

Definition 5.3.

Let R be a commutative ring. A nonempty subset I of R is called an *ideal* if

(I) $a - b \in I$ for all $a, b \in I$;

(II) $ar \in I$ for all $a \in I$ and for all $r \in R$. □

Two very special ideals in any commutative ring R are the subsets {0} and R since properties (i) and (ii) of Definition 5.3 are clearly satisfied by these two subsets. We call {0} the *zero ideal* and R the *universal ideal*. By a *proper ideal* we mean any ideal I such that $I \neq \{0\}$ and $I \neq R$. Note that the subset {0} is not a subring of R according to Definition 5.1 since it is not closed under the nullary operation 'select 1' defined on R (i.e. {0} does not contain the multiplicative identity of R). This is a characteristic property of ideals which we formulate as the following theorem.

Theorem 5.1.

Every proper ideal I in a commutative ring R is closed under all of the ring operations defined on R except that I is not closed under the nullary operation 'select 1' (i.e. $1 \notin I$).

Proof:

It is easy to verify that property (i) of Definition 5.3 guarantees that I is closed under the operations + (binary), − (unary), and 'select 0' (nullary). (Indeed property (i) is used precisely because it is sufficient to guarantee closure with respect to these three 'group' operations). It is also trivial to see that property (ii) guarantees that I is closed under multiplication. As for the nullary operation 'select 1', if $1 \in I$ then by property (ii) $r \in I$ for all $r \in R$ − i.e. $I = R$ so that I is not a proper ideal. □

The crucial property of an ideal I, apart from the closure properties of Theorem 5.1, is the 'extended closure' property (ii) of Definition 5.3 which guarantees that I is closed under multiplication *by any element of the ring* R.

Example 5.8.

In the integral domain **Z** of integers, the subset

$$<m> = \{mr : r = 0, \pm 1, \pm 2, \cdots\}$$

for some fixed integer $m \in \mathbf{Z}$ is an ideal called the *ideal generated by m*. For example, the ideal <4> is the set

$$<4> = \{0, \pm 4, \pm 8, \pm 12, \cdots\}.$$ □

Example 5.9.

In the polynomial domain $\mathbf{Q}[x]$, the subset

$$<p(x)> = \{p(x)\, a(x) : a(x) \in \mathbf{Q}[x]\}$$

for some fixed polynomial $p(x) \in \mathbf{Q}[x]$ is an ideal called the *ideal generated by* p(x). For example, the ideal $<x - \alpha>$ for some fixed $\alpha \in \mathbf{Q}$ is the set of all polynomials over **Q** which have $x - \alpha$ as a factor (i.e. polynomials a(x) such that $a(\alpha) = 0$). □

Example 5.10.

In the bivariate polynomial domain $\mathbf{Z}[x,y]$, the subset

$$<p_1(x,y),\ p_2(x,y)> = \{p_1(x,y)\, a_1(x,y) + p_2(x,y)\, a_2(x,y) :\ a_1(x,y), a_2(x,y) \in \mathbf{Z}[x,y]\}$$

for some fixed polynomials $p_1(x,y),\ p_2(x,y) \in \mathbf{Z}[x,y]$ is an ideal called the *ideal generated by*

$p_1(x,y)$ *and* $p_2(x,y)$. For example, the ideal $\langle x,y \rangle$ is the set of all bivariate polynomials over **Z** with constant term zero. Also note that $\langle y - \alpha \rangle$ for some fixed $\alpha \in \mathbf{Z}$ is an ideal in $\mathbf{Z}[x,y]$ consisting of all bivariate polynomials over **Z** which have $y - \alpha$ as a factor (i.e. polynomials $a(x,y)$ such that $a(x,\alpha) = 0$). Similarly $\langle m \rangle$ for some fixed integer $m \in \mathbf{Z}$ is an ideal in $\mathbf{Z}[x,y]$ consisting of all bivariate polynomials whose integer coefficients are multiples of m. □

The fact that an ideal I in a commutative ring R is closed under addition and is closed under multiplication by any element of R, implies that if I contains the n elements $a_1, \ldots, a_n$ then it must contain the set of all linear combinations of these elements, defined by:

$$\langle a_1, \ldots, a_n \rangle = \{a_1 r_1 + \cdots + a_n r_n : r_i \in \mathrm{R}\}.$$

On the other hand, it is easy to verify that for any given elements $a_1, \ldots, a_n \in$ R, the set $\langle a_1, \ldots, a_n \rangle$ of all linear combinations of these elements is an ideal in R. The ideal $\langle a_1, \ldots, a_n \rangle$ is called the ideal with *basis* $a_1, \ldots, a_n$.

Definition 5.4.

An ideal I in a commutative ring R is called an *ideal with finite basis* if I can be expressed as the set $\langle a_1, \ldots, a_n \rangle$ of all linear combinations of a finite number n of elements $a_1, \ldots, a_n \in$ R. □

Definition 5.5.

An ideal I in a commutative ring R is called a *principal ideal* if I can be expressed as the set $\langle a \rangle$ of all multiples of a single element $a \in$ R. □

Domains with Special Ideals

Definition 5.6.

An integral domain D is called a *Noetherian integral domain* if every ideal in D is an ideal with finite basis. □

Definition 5.7.

An integral domain D is called a *principal ideal domain* if every ideal in D is a principal ideal. □

It can be proved that every Euclidean domain is a principal ideal domain and therefore the domains **Z** and **Q**[x] considered in Examples 5.8 and 5.9 are principal ideal domains. The polynomial domain $\mathbf{Z}[x,y]$ considered in Example 5.10 is an example of an integral domain that is not a principal ideal domain since it is not possible to generate the ideal $\langle x,y \rangle$, for example, by a single element. However it can be proved that if D is a Noetherian integral domain then so is the domain D[x], which implies by induction that $\mathbf{Z}[x,y]$ and indeed any multivariate polynomial domain over **Z** or over a field is a Noetherian integral domain.

In the hierarchy of domains given in Table 2.3 of chapter 2, the principal ideal domain lies between the unique factorization domain (UFD) and the Euclidean domain (i.e. every Euclidean domain is a principal ideal domain and every principal ideal domain is a UFD). However the multivariate polynomial domains considered in this book are Noetherian integral domains but are not principal ideal domains. The abstract concept of a Noetherian integral domain, unlike a principal ideal domain, is not simply a UFD which satisfies additional axioms. (For example, the integral domain

$$\mathrm{S} = \{a + b\sqrt{-5} : a,b \in \mathbf{Z}\}$$

considered in Exercises 2-8 and 2-9 is a Noetherian integral domain but is not a UFD).

In the sequel we will require the concepts of the sum and product of two ideals and also the concept of an integral power of an ideal. These concepts are defined in the following definition in the context of an arbitrary Noetherian integral domain. Before proceeding to the definition let us note the following generalization of our notation for specifying ideals in a Noetherian integral domain D. If I and J are two ideals in D then by the notation $\langle I, J\rangle$ we understand the ideal $\langle a_1, \ldots, a_n, b_1, \ldots, b_m\rangle$ where $a_1, \ldots, a_n \in D$ forms a basis for I and where $b_1, \ldots, b_m \in D$ forms a basis for J (i.e. $I = \langle a_1, \ldots, a_n\rangle$ and $J = \langle b_1, \ldots, b_m\rangle$). The notation $\langle I, b\rangle$ or $\langle b, I\rangle$ where $b \in D$ and I is an ideal in D is similarly defined – i.e. $\langle I, b\rangle = \langle I, \langle b\rangle\rangle$.

Definition 5.8.

Let I and J be two ideals in a Noetherian integral domain D and suppose $I = \langle a_1, \ldots, a_n\rangle$, $J = \langle b_1, \ldots, b_m\rangle$ for elements $a_i \in D$ $(1 \leqslant i \leqslant n)$, $b_j \in D$ $(1 \leqslant j \leqslant m)$.

(i) The *sum* of the ideals I and J in D is the ideal $\langle I, J\rangle = \langle a_1, \ldots, a_n, b_1, \ldots, b_m\rangle$. Note that the ideal $\langle I, J\rangle$ consists of all possible sums $a + b$ where $a \in I$ and $b \in J$.

(ii) The *product* $I \cdot J$ of the ideals I and J in D is the ideal generated by all elements $a_i b_j$ such that a_i is a basis element for I and b_j is a basis element for J. Thus the product can be expressed as

$$I \cdot J = \langle a_1 b_1, a_1 b_2, \ldots, a_1 b_m, a_2 b_1, a_2 b_2, \ldots, a_2 b_m, \ldots, a_n b_1, a_n b_2, \ldots, a_n b_m\rangle.$$

(iii) The i-th *power* of the ideal I in D (for i a positive integer) is defined recursively in terms of products of ideals as follows:

$$I^1 = I;$$

$$I^i = I \cdot I^{i-1} \quad \text{for } i \geqslant 2. \qquad \square$$

The application of Definition 5.8 to the case of principal ideals should be noted in particular. For the product of two principal ideals $\langle a\rangle$ and $\langle b\rangle$ in D it follows from Definition 5.8 that

$$\langle a\rangle \cdot \langle b\rangle = \langle ab\rangle.$$

Similarly for the i-th power of the ideal $\langle a\rangle$ in D we have

$$\langle a\rangle^i = \langle a^i\rangle \quad \text{for } i \geqslant 1.$$

The sum of the ideals $\langle a\rangle$ and $\langle b\rangle$ in D is simply the ideal $\langle a, b\rangle$ which may not be a principal ideal. However if D is a principal ideal domain then the sum $\langle a, b\rangle$ must be a principal ideal. It can be proved that in any principal ideal domain,

$$\langle a, b\rangle = \langle \mathrm{GCD}(a, b)\rangle.$$

(Note that since D is a principal ideal domain it is also a UFD and therefore the GCD exists by Theorem 2.1).

The Characterization Theorem

Definition 5.9.

Let R and R′ be commutative rings and let $\phi: R \to R'$ be a morphism. The *kernel* K of the morphism ϕ is the set defined by:

$$K = \phi^{-1}(0) = \{a: a \in R \text{ and } \phi(a) = 0\}. \qquad \square$$

Theorem 5.2.

Let R and R′ be commutative rings. The kernel K of a morphism $\phi : R \to R'$ is an ideal in R.

Proof:

The set K is not empty since $\phi(0) = 0$. If $a,b \in K$ then

$$\phi(a - b) = \phi(a) - \phi(b) = 0 - 0 = 0$$

so that $a - b \in K$, proving property (i) of Definition 5.3. Similarly property (ii) holds because if $a \in K$ and $r \in R$ then

$$\phi(ar) = \phi(a)\phi(r) = 0 \cdot \phi(r) = 0$$

so that $ar \in K$. □

There is a direct connection between the homomorphic images of a commutative ring R and the set of ideals in R. Recall that every morphism $\phi : R \to R'$ determines a homomorphic image $\phi(R)$ of the ring R. We see from Theorem 5.2 that to each morphism $\phi : R \to R'$ there corresponds an ideal in R which is the kernel K of ϕ. Conversely, we shall see in the next section that to each ideal I in R there corresponds a homomorphic image R′ of R such that I is the kernel of the corresponding morphism $\phi: R \to R'$. We first prove that a homomorphic image of R is completely determined (up to isomorphism) by the ideal of elements mapped onto zero.

Theorem 5.3. Characterization Theorem.

Let R be a commutative ring and let K be an ideal in R. If $\phi_1: R \to R'$ and $\phi_2: R \to R''$ are two morphisms both having kernel K then the correspondence between the two homomorphic images $\phi_1(R)$ and $\phi_2(R)$ defined by

$$\phi_1(a) \leftrightarrow \phi_2(a) \text{ for all } a \in R$$

is an isomorphism.

Proof:

Suppose ϕ_1 and ϕ_2 have kernel K. The correspondence mentioned above can be formally specified as follows. For any element $\alpha \in \phi_1(R)$ the set of pre-images of α is the set

$$\phi_1^{-1}(\alpha) = \{a \in R:\ \phi_1(a) = \alpha\}.$$

We define the mapping

$$\psi : \phi_1(R) \to \phi_2(R)$$

by

$$\psi(\alpha) = \phi_2(\phi_1^{-1}(\alpha)) \text{ for all } \alpha \in \phi_1(R) \quad (3)$$

where we claim that the image under ϕ_2 of the set $\phi_1^{-1}(\alpha)$ is a *single* element in $\phi_2(R)$. To see this, note that if $a,b \in R$ are two elements in the set $\phi_1^{-1}(\alpha)$ then $a - b \in K$ (the kernel of ϕ_1) since

$$\phi_1(a - b) = \phi_1(a) - \phi_1(b) = \alpha - \alpha = 0.$$

Hence,

$$\phi_2(a - b) = \phi_2(a) - \phi_2(b) = 0$$

(because K is also the kernel of ϕ_2) yielding

$$\phi_2(a) = \phi_2(b).$$

Thus (3) defines a valid mapping of $\phi_1(R)$ into $\phi_2(R)$ and clearly ψ specifies the correspondence

mentioned above. We may calculate (3) by letting $a \in \phi_1^{-1}(\alpha)$ be *any* particular pre-image of α and setting $\psi(\alpha) = \phi_2(a)$.

We now claim that ψ is an isomorphism. Properties (i) - (iii) of Definition 5.2 are satisfied by ψ because they are satisfied by the morphisms ϕ_1 and ϕ_2. To see this, for any $\alpha, \beta \in \phi_1(R)$ let $a \in \phi_1^{-1}(\alpha)$ and $b \in \phi_1^{-1}(\beta)$ be particular pre-images of α and β, respectively. Then a particular pre-image of $\alpha + \beta \in \phi_1(R)$ is the element $a + b \in R$ since

$$\phi_1(a + b) = \phi_1(a) + \phi_1(b) = \alpha + \beta;$$

similarly, $ab \in R$ is a particular pre-image of $\alpha\beta \in \phi_1(R)$. Thus,

$$\psi(\alpha + \beta) = \phi_2(\phi_1^{-1}(\alpha + \beta)) = \phi_2(a + b) = \phi_2(a) + \phi_2(b) = \psi(\alpha) + \psi(\beta)$$

and

$$\psi(\alpha\beta) = \phi_2(\phi_1^{-1}(\alpha\beta)) = \phi_2(ab) = \phi_2(a)\,\phi_2(b) = \psi(\alpha)\,\psi(\beta)$$

verifying properties (i) and (ii). To verify property (iii), note that $1 \in R$ is a particular pre-image of $1 \in \phi_1(R)$ because $\phi_1(1) = 1$ (i.e. ϕ_1 is a morphism) and therefore

$$\psi(1) = \phi_2(\phi_1^{-1}(1)) = \phi_2(1) = 1$$

(because ϕ_2 is a morphism). We have thus proved that ψ is a morphism. It is easy to see that the mapping ψ is surjective since the mappings

$$\phi_1 : R \to \phi_1(R) \text{ and } \phi_2 : R \to \phi_2(R)$$

are surjective. To see that ψ is injective, let $\alpha,\ \beta \in \phi_1(R)$ have particular pre-images $a,b \in R$ (i.e. $\alpha = \phi_1(a)$ and $\beta = \phi_1(b)$) and suppose that $\psi(\alpha) = \psi(\beta)$. Then we have

$$\phi_2(\phi_1^{-1}(\alpha)) = \phi_2(\phi_1^{-1}(\beta))$$

$$\Rightarrow \phi_2(a) = \phi_2(b)$$

$$\Rightarrow a - b \in K \text{ (the kernel of } \phi_2)$$

$$\Rightarrow \phi_1(a) = \phi_1(b) \text{ (because K is the kernel of } \phi_1)$$

$$\Rightarrow \alpha = \beta.$$

Hence the mapping ψ is injective and ψ defines an isomorphism between $\phi_1(R)$ and $\phi_2(R)$. □

Corollary to Theorem 5.3.

Let $\phi : R \to R'$ be a morphism between commutative rings R and R'. If K denotes the kernel of ϕ then:

(i) $K = \{0\}$ if and only if ϕ is injective (i.e. $\phi(R) = R$ in the sense of isomorphism);

(ii) $K = R$ if and only if $\phi(R) = \{0\}$.

Proof:

(i) If ϕ is injective then $K = \{0\}$ because $\phi(0) = 0$. In the other direction, suppose $K = \{0\}$. Then since the identity mapping $\Phi : R \to R$ is also a morphism with kernel $\{0\}$, we have from Theorem 5.3 that the mapping $\phi : R \to \phi(R)$ is an isomorphism; i.e. ϕ is injective.

(ii) By definition of the kernel K, if $\phi(R) = \{0\}$ then $K = R$ and if $K = R$ then $\phi(R) = \{0\}$. □

By Theorem 5.3, we can specify a homomorphic image of a commutative ring R by simply specifying the ideal of elements which is mapped onto zero. The above corollary specifies the two 'degenerate' cases corresponding to the two choices of ideals which are not proper ideals. By a

proper homomorphic image of a commutative ring R we mean a homomorphic image specified by a morphism ϕ whose kernel is a proper ideal in R.

5.3. HOMOMORPHIC IMAGES

Quotient Rings

If R is a commutative ring and if I is any ideal in R, we now show how to construct a homomorphic image $\phi(R)$ such that I is the kernel of the morphism ϕ. Note that if $\phi : R \to R'$ is to be a morphism with kernel I then we must have

$$\phi(a) = \phi(b) \text{ if and only if } a - b \in I.$$

We therefore define the following *congruence relation* on R:

(4) $\quad a \equiv b$ if and only if $a - b \in I$.

It is readily verified that the congruence relation $\equiv$ is an equivalence relation on R and it therefore divides R into equivalence classes, called *residue classes*. For any element $a \in R$, it is easy to prove that every element in the set

$$a + I = \{a + c : c \in I\}$$

belongs to the same residue class with respect to the congruence relation $\equiv$, that $a \in a + I$, and moreover that if b is in the same residue class as a (i.e. if $b \equiv a$) then $b \in a + I$. Thus the residue class containing a is precisely the set $a + I$.

The set of all residue classes with respect to the congruence relation $\equiv$ defined by (4) is called a *quotient set*, denoted by

$$R/I = \{a + I : a \in R\}$$

(read 'R modulo the ideal I'). Note that if a and b are in the same residue class (i.e. if $a \equiv b$) then $a + I$ and $b + I$ are two representatives for the same element in the quotient set R/I. We define the operations of addition and multiplication on the quotient set R/I, in terms of the operations defined on R, as follows:

(5) $\quad (a + I) + (b + I) = (a + b) + I$;

(6) $\quad (a + I)(b + I) = (ab) + I$.

Using the fact that I is an ideal, it can be verified that the operations of addition and multiplication on residue classes in R/I are well-defined by (5) - (6) in the sense that the definitions are independent of the particular representatives used for the residue classes. (Note that the terminology being used here is very similar to the terminology used in chapter 2 for defining the quotient field of an integral domain). The following theorem proves that the quotient set R/I with the operations (5) - (6) is a commutative ring, and R/I is called the *quotient ring* of R modulo the ideal I. Moveover, the theorem specifies a 'natural' homomorphism $\phi : R \to R/I$ such that I is the kernel of ϕ and the quotient ring R/I is the desired homomorphic image of R.

Theorem 5.4.

Let R be a commutative ring and let I be an ideal in R. The quotient set R/I is a commutative ring under the operations (5) - (6) and the mapping $\phi : R \to R/I$ defined by

$$\phi(a) = a + I \text{ for all } a \in R$$

is an epimorphism with kernel I.

Proof:

First note that the residue classes $0 + I$ and $1 + I$ act as the zero and identity (respectively) in R/I since from (5) - (6) we have:

$$(a + I) + (0 + I) = a + I \quad \text{for any } a + I \in R/I;$$

$$(a + I)(1 + I) = a + I \quad \text{for any } a + I \in R/I.$$

Now consider the mapping $\phi : R \to R/I$ defined by

$$\phi(a) = a + I \quad \text{for all } a \in R.$$

It follows immediately from (5) - (6) that for any $a, b \in R$,

$$\phi(a + b) = (a + b) + I = (a + I) + (b + I) = \phi(a) + \phi(b)$$

and

$$\phi(ab) = (ab) + I = (a + I)(b + I) = \phi(a)\,\phi(b).$$

Also,

$$\phi(1) = 1 + I$$

by definition of ϕ. Thus ϕ is a morphism according to Definition 5.2. But ϕ is surjective by the definition of R/I, so ϕ is an epimorphism. The fact that R/I is a homomorphic image of R implies that R/I is a commutative ring. Finally, we can prove that the kernel of ϕ is precisely I as follows:

$$a \in I \implies \phi(a) = a + I = 0 + I$$

and

$$\phi(a) = 0 + I \implies a + I = 0 + I \implies a - 0 \in I \implies a \in I. \quad \square$$

Example 5.11.

In the integral domain $\mathbf{Z}$ of integers, we noted in Example 5.8 that $\langle m \rangle$ is an ideal, for some fixed $m \in \mathbf{Z}$. Thus the quotient ring $\mathbf{Z}/\langle m \rangle$ is a homomorphic image of $\mathbf{Z}$ and $\langle m \rangle$ is the kernel of the natural homomorphism $\phi : \mathbf{Z} \to \mathbf{Z}/\langle m \rangle$. Assuming that m is positive, the elements of $\mathbf{Z}/\langle m \rangle$ are given by:

$$\mathbf{Z}/\langle m \rangle = \{0 + \langle m \rangle, 1 + \langle m \rangle, \ldots, m - 1 + \langle m \rangle\}.$$

We usually denote $\mathbf{Z}/\langle m \rangle$ by $\mathbf{Z}_m$ (the ring of integers modulo m) and we may denote its elements simply by

$$\{0, 1, \ldots, m - 1\}.$$

The natural homomorphism is precisely the modular homomorphism $\phi_m : \mathbf{Z} \to \mathbf{Z}_m$ defined in section 5.1. $\square$

Example 5.12.

In the polynomial domain $\mathbf{Q}[x]$, we noted in Example 5.9 that $\langle p(x) \rangle$ is an ideal for a fixed polynomial $p(x) \in \mathbf{Q}[x]$. Thus the quotient ring $\mathbf{Q}[x]/\langle p(x) \rangle$ is a homomorphic image of $\mathbf{Q}[x]$ and $\langle p(x) \rangle$ is the kernel of the natural homomorphism $\phi : \mathbf{Q}[x] \to \mathbf{Q}[x]/\langle p(x) \rangle$. Two polynomials $a(x), b(x) \in \mathbf{Q}[x]$ are in the same residue class if they have the same remainder after division by $p(x)$. In particular if $p(x) = x - \alpha$ for some constant $\alpha \in \mathbf{Q}$ then

$$\mathbf{Q}[x]/\langle x - \alpha \rangle = \{r + \langle x - \alpha \rangle : r \in \mathbf{Q}\}.$$

In this case we may identify $\mathbf{Q}[x]/\langle x - \alpha \rangle$ with $\mathbf{Q}$ and the natural homomorphism is precisely

the evaluation homomorphism $\phi_{x-\alpha}$: $\mathbf{Q}[x] \rightarrow \mathbf{Q}$ defined in section 5.1. (See Exercise 5-14). □

Example 5.13.

In the bivariate polynomial domain $\mathbf{Z}[x,y]$, we noted in Example 5.10 that $<m>$ is an ideal for some fixed integer $m \in \mathbf{Z}$. The quotient ring $\mathbf{Z}[x,y]/<m>$ can be identified with the ring $\mathbf{Z}_m[x,y]$ and the natural homomorphism is precisely the modular homomorphism $\phi_m : \mathbf{Z}[x,y] \rightarrow \mathbf{Z}_m[x,y]$ defined in section 5.1. We also noted in Example 5.10 that $<y - \alpha>$ is an ideal in $\mathbf{Z}[x,y]$, for some fixed $\alpha \in \mathbf{Z}$. The quotient ring $\mathbf{Z}[x,y]/<y - \alpha>$ can be identified with the ring $\mathbf{Z}[x]$ (see Example 5.12) and the natural homomorphism is the evaluation homomorphism $\phi_{y-\alpha}$: $\mathbf{Z}[x,y] \rightarrow \mathbf{Z}[x]$ defined in section 5.1. (See Exercise 5-15). □

Ideal Notation for Homomorphisms

The choice of notation used for the modular and evaluation homomorphisms defined in section 5.1 and used in the above examples can now be justified. In general if R is a commutative ring then any ideal I in R determines a homomorphic image R/I and we use the notation ϕ_I to denote the corresponding natural homomorphism from R to R/I. Thus if $R = \mathbf{Z}[x_1, \ldots, x_v]$ and if $I = <m>$ for some fixed integer $m \in \mathbf{Z}$ then $\phi_{<m>}$ (or simply ϕ_m) denotes the modular homomorphism which projects $\mathbf{Z}[x_1, \ldots, x_v]$ onto $\mathbf{Z}[x_1, \ldots, x_v]/<m> = \mathbf{Z}_m[x_1, \ldots, x_v]$. Similarly for the evaluation homomorphism we have $R = D[x_1, \ldots, x_v]$ for some coefficient domain D (usually D will be a field $\mathbf{Z}_p$ in practical applications), and if $I = <x_i - \alpha>$ for some fixed $\alpha \in D$ then $\phi_{<x_i-\alpha>}$ (or simply $\phi_{x_i-\alpha}$) denotes the evaluation homomorphism which projects $D[x_1, \ldots, x_v]$ onto $D[x_1, \ldots, x_v]/<x_i - \alpha> = D[x_i, \ldots, x_{i-1}, x_{i+1}, \ldots, x_v]$.

As we have noted previously, modular and evaluation homomorphisms will be used in practice to project the multivariate polynomial domain $\mathbf{Z}[x_1, \ldots, x_v]$ onto a Euclidean domain $\mathbf{Z}_p[x_1]$ or else onto a field $\mathbf{Z}_p$. For example the projection of a polynomial domain $D[x_1, \ldots, x_v]$ onto its coefficient domain D can be accomplished by a composite homomorphism of the form

$$\phi_{x_1-\alpha_1}\, \phi_{x_2-\alpha_2} \cdots \phi_{x_v-\alpha_v}$$

where $\alpha_i \in D$, $1 \leqslant i \leqslant v$. It is convenient to express such a composite homomorphism as a single homomorphism ϕ_I but in order to do so we must specify the kernel I of the composite homomorphism. The following theorem proves that under special conditions (which are satisfied in the cases of interest here) the kernel of a composite homomorphism is simply the sum of the individual kernels, where the 'sum' of two ideals was defined in Definition 5.8.

Theorem 5.5.

Let $D[x_1, \ldots, x_v]$ be a polynomial domain over a UFD D. Let $\phi_{x_i-\alpha_i}$ be an evaluation homomorphism defined on $D[x_1, \ldots, x_v]$ with kernel $<x_i - \alpha_i>$ and let ϕ_I be another homomorphism defined on $D[x_1, \ldots, x_v]$ with kernel I. Suppose that the homomorphism ϕ_I is *independent of* the homomorphism $\phi_{x_i-\alpha_i}$ in the sense that the composite mappings $\phi_{x_i-\alpha_i}\, \phi_I$ and $\phi_I\, \phi_{x_i-\alpha_i}$ are valid homomorphisms defined on $D[x_1, \ldots, x_v]$ and moreover the composition of these two homomorphisms is commutative (i.e. $\phi_{x_i-\alpha_i}\, \phi_I = \phi_I\, \phi_{x_i-\alpha_i}$). Then the kernel of the composite homomorphism is the sum $<x_i - \alpha_i, I>$ of the two kernels. Notationally,

$$\phi_{x_i - \alpha_i}\, \phi_I = \phi_{<x_i-\alpha_i,\, I>}.$$

Proof:

We must prove that for any polynomial $a \in D[x_1, \ldots, x_v]$, $\phi_{x_i - \alpha_i}\, \phi_1(a) = 0$ if and only if $a \in \langle x_i - \alpha_i, I \rangle$.

'if':

Suppose $a \in \langle x_i - \alpha_i, I \rangle$. Then $a = p + r$ for some polynomials $p \in \langle x_i - \alpha_i \rangle$ and $r \in I$. Hence

$$\begin{aligned} \phi_{x_i - \alpha_i}\, \phi_1(a) &= \phi_{x_i - \alpha_i}\, \phi_1(p) + \phi_{x_i - \alpha_i}\, \phi_1(r) \\ &= \phi_{x_i - \alpha_i}\, \phi_1(p) \text{ because } r \in I \\ &= \phi_1\, \phi_{x_i - \alpha_i}(p) \text{ by commutativity} \\ &= 0 \text{ because } p \in \langle x_i - \alpha_i \rangle. \end{aligned}$$

'only if':

Suppose $\phi_{x_i - \alpha_i}\, \phi_1(a) = 0$ for a polynomial $a \in D[x_1, \ldots, x_v]$. Consider the polynomial domain $D[x_1, \ldots, x_v]$ as the univariate domain $C[x_i]$ over the coefficient domain $C = D[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_v]$. Then since $C[x_i]$ is a UFD the pseudo-division property holds and applying it to the polynomials a and $(x_i - \alpha_i)$, we can write

$$(7) \qquad a = (x_i - \alpha_i)\, q + r$$

for some polynomials $q, r \in C[x_i]$ with either $r = 0$ or $\partial_i[r] < \partial_i[x_i - \alpha_i] = 1$. (Note that in applying the pseudo-division property to obtain (7) we have used the fact that the leading coefficient of the 'divisor' $x_i - \alpha_i$ is 1). Hence a can be expressed as the sum (7) where the first term of the sum is clearly a member of the ideal $\langle x_i - \alpha_i \rangle$ and it remains only to prove that $r \in I$. We will then have the desired result that $a \in \langle x_i - \alpha_i, I \rangle$.

To prove that $r \in I$, apply the composite homomorphism $\phi_{x_i - \alpha_i}\, \phi_1$ to equation (7). Then since by supposition $\phi_{x_i - \alpha_i}\, \phi_1(a) = 0$ we get

$$\begin{aligned} 0 &= \phi_{x_i - \alpha_i}\, \phi_1((x_i - \alpha_i) q) + \phi_{x_i - \alpha_i}\, \phi_1(r) \\ &= \phi_1\, \phi_{x_i - \alpha_i}(r) \text{ by commutativity} \\ &= \phi_1(r) \text{ because either } r = 0 \text{ or } \partial_i[r] = 0 \end{aligned}$$

(where in the last step we have used the fact that the evaluation homomorphism $\phi_{x_i - \alpha_i}$ clearly acts as the identity mapping on any polynomial r which is independent of x_i). But $\phi_1(r) = 0$ implies that $r \in I$. □

From Theorem 5.5 we see that if $\phi_{x_i - \alpha_i}$ and $\phi_{x_j - \alpha_j}$ $(j \neq i)$ are two distinct evaluation homomorphisms defined on a polynomial domain $D[x_1, \ldots, x_v]$ (where D is a UFD) then

$$\phi_{x_i - \alpha_i}\, \phi_{x_j - \alpha_j} = \phi_{\langle x_i - \alpha_i,\, x_j - \alpha_j \rangle}.$$

By repeated application of Theorem 5.5 we have the more general result that for any n distinct evaluation homomorphisms $\phi_{x_1 - \alpha_1}, \ldots, \phi_{x_n - \alpha_n}$ defined on $D[x_1, \ldots, x_v]$, where $1 \leqslant n \leqslant v$,

$$\phi_{x_1 - \alpha_1}\, \phi_{x_2 - \alpha_2} \cdots \phi_{x_n - \alpha_n} = \phi_{\langle x_1 - \alpha_1, \ldots, x_n - \alpha_n \rangle}.$$

Thus the notation $\phi_{\langle x_1 - \alpha_1, \ldots, x_n - \alpha_n \rangle}$ can be read 'substitute α_i for x_i, $1 \leqslant i \leqslant n$' and we call this a *multivariate evaluation homorphism*. (Note that the order in which the substitutions are performed is irrelevant).

It also follows from Theorem 5.5 that if $\phi_p : \mathbf{Z}[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_p[x_1, \ldots, x_v]$ is a modular homomorphism (with p a prime integer) and if

$$\phi_{x_i - \alpha_i} : \mathbf{Z}_p[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_p[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_v]$$

is an evaluation homomorphism (with $\alpha_i \in \mathbf{Z}_p$) then

$$\phi_{x_i - \alpha_i} \phi_p = \phi_{<x_i - \alpha_i, p>}.$$

Again by repeated application of Theorem 5.5 we can generalize this result to show that

$$\phi_I \phi_p = \phi_{<I,p>}$$

if I is the kernel of a multivariate evaluation homomorphism. In practical applications the most commonly used homomorphisms will be of the form

$$\phi_{<I,p>} : \mathbf{Z}[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_p[x_1]$$

where $I = <x_2 - \alpha_2, \ldots, x_v - \alpha_v>$ with $\alpha_i \in \mathbf{Z}_p$ $(2 \leq i \leq v)$. For implementation purposes a composite homomorphism $\phi_{<I,p>}$ where p is a prime integer and I is the kernel of a multivariate evaluation homomorphism will be viewed as the composition of precisely two mappings, namely a modular homomorphism

$$\phi_p : \mathbf{Z}[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_p[x_1, \ldots, x_v]$$

followed by a multivariate evaluation homomorphism

$$\phi_I : \mathbf{Z}_p[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_p[x_1, \ldots, x_v]/I.$$

The notation $\phi_{<I,p>}$ will be freely used for this pair of mappings but for computational efficiency it will be important that the order of application of the mappings is as specified above, namely $\phi_{<I,p>} = \phi_I \phi_p$.

Congruence Arithmetic

It is useful to formally specify a *congruence notation* that is used when performing arithmetic on residue classes in a homomorphic image R/I of a ring R. Recall that if I is an ideal in a commutative ring R then the residue classes (i.e. equivalence classes) which form the quotient ring R/I are determined by the congruence relation $\equiv$ defined on R by

$$a \equiv b \quad \text{if and only if} \quad a - b \in I.$$

We read this relation as 'a is congruent to b modulo I' and we write

$$a \equiv b \pmod{I}.$$

In the particular case where I is a principal ideal $<q>$ for some fixed element $q \in R$, we write $(\text{mod } q)$ rather than $(\text{mod } <q>)$. (This notation was already seen briefly in Example 5.7 for the particular case of 'modulo p' arithmetic in the quotient ring $\mathbf{Z}/<p> = \mathbf{Z}_p$).

We will have occasion to solve certain equations involving the congruence relation $\equiv$, so let us note some useful properties of $\equiv$ in addition to the standard properties of an equivalence relation. For any commutative ring R and I an ideal in R we have the following relationships. For any $a,b,c,d \in R$, if $a \equiv b \pmod{I}$ and $c \equiv d \pmod{I}$ then

(8) $a + c \equiv b + d \pmod{I}$;

(9) $a - c \equiv b - d \pmod{I}$;

(10) $ac \equiv bd \pmod{I}$.

For (8) and (9) it is easy to see that

$$(a \pm c) - (b \pm d) = (a - b) \pm (c - d) \in I.$$

The proof of (10) is only slightly less obvious, namely

$$ac - bd = c(a - b) + b(c - d) \in I.$$

We will need another property which will allow us to solve a congruence equation of the form

$$ax \equiv b \pmod{I}$$

for x if a and b are given. Clearly if there is an element, say a^{-1}, such that $aa^{-1} \equiv 1 \pmod{I}$ then by (10) it follows that $aa^{-1}b \equiv b \pmod{I}$ so that choosing $x = a^{-1}b$ yields a solution to the given congruence equation. Since an element in an arbitrary commutative ring does not necessarily have a multiplicative inverse, the property which will allow us to solve congruence equations in the above sense will be less general than properties (8) - (10).

In order to obtain the desired property we will restrict attention to the case where the ring R is a Euclidean domain D. As we noted in section 5.2, every ideal I in a Euclidean domain D is a principal ideal so $I = \langle q \rangle$ for some fixed element $q \in D$. The following theorem states a condition under which an element $a \in D$ has an inverse modulo $\langle q \rangle$. The proof of the theorem is constructive – i.e. it gives an algorithm for computing inverses modulo $\langle q \rangle$.

Theorem 5.6.

Let $\langle q \rangle$ be an ideal in a Euclidean domain D and let $a \in D$ be relatively prime to q (i.e. $GCD(a,q) = 1$). Then there exists an element $a^{-1} \in D$ such that

$$aa^{-1} \equiv 1 \pmod{q}.$$

This is equivalent to saying that in the homomorphic image $D/\langle q \rangle$ the element $\phi_q(a)$ has a multiplicative inverse.

Proof:

Since D is a Euclidean domain we can apply the extended Euclidean algorithm (Algorithm 2.2) to $a,q \in D$ yielding elements $s,t \in D$ such that

$$sa + tq = 1,$$

where we have used the fact that $GCD(a,q) = 1$. Then $sa - 1 \in \langle q \rangle$, or $sa \equiv 1 \pmod{q}$. Thus $a^{-1} = s$ is the desired inverse.

To show the equivalence of the last statement in the theorem, first suppose that $aa^{-1} \equiv 1 \pmod{q}$. Then $aa^{-1} - 1 \in \langle q \rangle$, so $\phi_q(aa^{-1} - 1) = 0$ which yields $\phi_q(a)\phi_q(a^{-1}) = 1$ – i.e. $\phi_q(a^{-1})$ is the multiplicative inverse of $\phi_q(a)$ in $D/\langle q \rangle$. In the other direction, suppose $\phi_q(a)$ has a multiplicative inverse $b \in D/\langle q \rangle$. Then there is an element $b \in D$ such that $\phi_q(b) = b$. We have $\phi_q(a)\phi_q(b) = 1$ which implies that $\phi_q(ab - 1) = 0$, or $ab - 1 \in \langle q \rangle$, or $ab \equiv 1 \pmod{q}$. □

Finally we are able to state the property of congruence relations that we have been seeking. For any Euclidean domain D and $\langle q \rangle$ the ideal generated by a fixed element $q \in D$ the following property holds:

(11) For any $a,b \in D$ with a relatively prime to q there is an element $a^{-1} \in D$ which is the inverse (mod q) of a and any element $x \in D$ such that

$$x \equiv a^{-1}b \pmod{q}$$

is a solution of the congruence equation

$$ax \equiv b \pmod{q}.$$

5.4. THE INTEGER CHINESE REMAINDER ALGORITHM

We now turn to the development of algorithms for inverting homomorphisms. The basic tenet of these 'inversion' algorithms is that under appropriate conditions an element a in a ring R can be reconstructed if its images $\phi_{I_i}(a)$, $i = 1, 2, \cdots$ are known in an 'appropriate number' of homomorphic images R/I_i of R.

The Chinese Remainder Problem

Recall that for any fixed integer $m \in \mathbf{Z}$ the modular homomorphism $\phi_m : \mathbf{Z} \rightarrow \mathbf{Z}_m$ which projects the ring $\mathbf{Z}$ of integers onto the finite ring $\mathbf{Z}_m$ of 'integers modulo m' is specified by

(12) $\phi_m(a) = \text{rem}(a,m)$ for all $a \in \mathbf{Z}$.

Using congruence notation, if $a \in \mathbf{Z}$ and if $\phi_m(a) = \tilde{a} \in \mathbf{Z}_m$ then we write

$$a \equiv \tilde{a} \pmod{m}.$$

The classical mathematical problem known as the *Chinese remainder problem* can be stated as follows:

Given *moduli* $m_0, m_1, \ldots, m_n \in \mathbf{Z}$ and given corresponding *residues* $u_i \in \mathbf{Z}_{m_i}$, $0 \leqslant i \leqslant n$, find an integer $u \in \mathbf{Z}$ such that

$$u \equiv u_i \pmod{m_i}, \quad 0 \leqslant i \leqslant n.$$

(This problem, in a less general form, was considered by the ancient Chinese and by the ancient Greeks about 2000 years ago). Note that an algorithm for solving the Chinese remainder problem will be an algorithm for 'inverting' the modular homomorphism, since if we know the images (residues) $u_i = \phi_{m_i}(u)$ of an integer u, for several modular homomorphisms ϕ_{m_i}, then such an algorithm will reconstruct the integer u. (More correctly, the latter statement will be true once we have determined conditions such that there exists a *unique* integer u which solves the problem). The following theorem specifies conditions under which there exists a unique solution to the Chinese remainder problem.

Theorem 5.7. *Chinese Remainder Theorem.*

Let the moduli $m_0, m_1, \ldots, m_n \in \mathbf{Z}$ be integers which are pairwise relatively prime – i.e.

$$\text{GCD}(m_i, m_j) = 1 \text{ for } i \neq j,$$

and let $u_i \in \mathbf{Z}_{m_i}$, $i = 0, 1, \ldots, n$ be $n + 1$ specified residues. For any fixed integer $a \in \mathbf{Z}$ there exists a unique integer $u \in \mathbf{Z}$ which satisfies the following conditions:

(13) $a \leqslant u < a + m$, where $m = \prod_{i=0}^{n} m_i$;

(14) $u \equiv u_i \pmod{m_i}$, $0 \leqslant i \leqslant n$.

Proof:

Uniqueness:

Let $u,v \in \mathbf{Z}$ be two integers satisfying conditions (13) and (14). Then using the fact that $\equiv$ is an equivalence relation, it follows from condition (14) that

$$u \equiv v \pmod{m_i}, \quad \text{for } i = 0, 1, \ldots, n$$

$$\Rightarrow \quad u - v \in \langle m_i \rangle, \quad \text{for } i = 0, 1, \ldots, n$$

$$\Rightarrow \quad u - v \in \langle m \rangle \text{ where } m = \prod_{i=0}^{n} m_i$$

where in the last step we have used the fact that since the moduli $m_0, m_1, \ldots, m_n$ are pairwise relatively prime, an integer which is a multiple of each m_i must also be a multiple of the product m. But from condition (13) it follows that

$$| u - v | < m$$

and hence $u - v = 0$ since 0 is the only element of the ideal $<m>$ which has absolute value less than m. Thus $u = v$.

Existence:

Let u run through the m distinct integer values in the range specified by condition (13) and consider the corresponding $(n + 1)$-tuples $(\phi_{m_0}(u), \phi_{m_1}(u), \ldots, \phi_{m_n}(u))$, where ϕ_{m_i} is the modular homomorphism defined by (12). By the uniqueness proof above, no two of these $(n + 1)$-tuples can be identical and hence the $(n + 1)$-tuples also take on m distinct values. But since the finite ring $\mathbf{Z}_{m_i}$ contains precisely m_i elements there are exactly $m = \prod_{i=0}^{n} m_i$ distinct $(n + 1)$-tuples $(v_0, v_1, \ldots, v_n)$ such that $v_i \in \mathbf{Z}_{m_i}$. Hence each possible $(n + 1)$-tuple occurs exactly once and therefore there must be one value of u in the given range such that

$$(\phi_{m_0}(u), \phi_{m_1}(u), \ldots, \phi_{m_n}(u)) = (u_0, u_1, \ldots, u_n). \quad \square$$

It is important to note the sense in which the solution to the Chinese remainder problem is unique. If we are given $n + 1$ residues $u_i \in \mathbf{Z}_{m_i}$ $(0 \leqslant i \leqslant n)$ corresponding to $n + 1$ moduli m_i $(0 \leqslant i \leqslant n)$ (assumed to be pairwise relatively prime) then the Chinese remainder problem has an infinite set of integer solutions, but by property (13) of Theorem 5.7 (choosing $a = 0$) we see that the solution is unique if we restrict it to the range $0 \leqslant u < m$. Thus we say that the solution is *unique modulo m*. In other words, given $u_i \in \mathbf{Z}_{m_i}$ $(0 \leqslant i \leqslant n)$ the system of congruences (14) does not have a unique solution in the ring $\mathbf{Z}$ but it does have a unique solution in the ring $\mathbf{Z}_m$, where $m = \prod_{i=0}^{n} m_i$.

Different choices of values for the arbitrary integer a in Theorem 5.7 correspond to different representations for the ring $\mathbf{Z}_m$. The choice $a = 0$ corresponds to the familiar *positive representation* of $\mathbf{Z}_m$ as

$$\mathbf{Z}_m = \{0, 1, \ldots, m-1\}$$

(where we are assuming that m is positive). In practical applications all of the moduli $m_0, m_1, \ldots, m_n$ and m will be *odd positive* integers and another useful representation will be the *symmetric representation* of $\mathbf{Z}_m$ as

$$\mathbf{Z}_m = \{-\frac{m-1}{2}, \ldots, -1, 0, 1, \ldots, \frac{m-1}{2}\}.$$

The choice of value for the integer a in Theorem 5.7 which corresponds to the symmetric representation of $\mathbf{Z}_m$ is clearly

$$a = -\frac{m-1}{2}.$$

The proof given above for Theorem 5.7 is not a constructive proof since it would be highly impractical to determine the solution u by simply trying each element of the ring $\mathbf{Z}_m$ when m is a large integer. We will now proceed to develop an efficient algorithm for solving the Chinese remainder problem.

Garner's Algorithm

The algorithm which is generally used to solve the Chinese remainder problem is named after H. L. Garner who developed a version of the algorithm in the late 1950's. Given positive moduli $m_i \in Z$ $(0 \leqslant i \leqslant n)$ which are pairwise relatively prime and given corresponding residues $u_i \in \mathbf{Z}_{m_i}$ $(0 \leqslant i \leqslant n)$, we wish to compute the unique $u \in \mathbf{Z}_m$ (where $m = \prod_{i=0}^{n} m_i$) which satisfies the system of congruences (14). The key to Garner's algorithm is to express the solution $u \in \mathbf{Z}_m$ in the *mixed radix representation*

$$\textbf{(15)} \quad u = v_0 + v_1(m_0) + v_2(m_0 m_1) + \cdots + v_n\left(\prod_{i=0}^{n-1} m_i\right)$$

where $v_k \in \mathbf{Z}_{m_k}$ for $k = 0, 1, \ldots, n$.

The mixed radix representation (15) is not meaningful in the full generality stated above since the addition and multiplication operations appearing in (15) are to be performed in the ring $\mathbf{Z}_m$ but each *mixed radix coefficient* v_k lies in a different ring $\mathbf{Z}_{m_k}$. In order to make (15) meaningful, we will require that the rings $\mathbf{Z}_{m_k}$ $(0 \leqslant k \leqslant n)$ and $\mathbf{Z}_m$ be represented in one of the following two *consistent representations:*

(i) Each ring $\mathbf{Z}_{m_k}$ $(0 \leqslant k \leqslant n)$ and $\mathbf{Z}_m$ is represented in its positive representation; or

(ii) Each ring $\mathbf{Z}_{m_k}$ $(0 \leqslant k \leqslant n)$ and $\mathbf{Z}_m$ is represented in its symmetric representation (where we assume that each m_k is odd).

Then the natural identification of elements in a ring $\mathbf{Z}_{m_k}$ with elements in the larger ring $\mathbf{Z}_m$ gives the desired interpretation of (15). It can be proved that any $u \in \mathbf{Z}_m$ can be represented in the form (15) and if one of the consistent representations (i) or (ii) is used then the coefficients v_k $(0 \leqslant k \leqslant n)$ are uniquely determined. It should be noted that in the case when the positive consistent representation (i) is used, (15) is a straightforward generalization of the familiar fact the any integer u in the range $0 \leqslant u < \beta^{n+1}$ (i.e. $u \in \mathbf{Z}_{\beta^{n+1}}$), for a positive integer $\beta > 1$, can be uniquely represented in the *radix β representation:*

$$u = v_0 + v_1\beta + v_2\beta^2 + \cdots + v_n\beta^n$$

where $0 \leqslant v_k < \beta$ (i.e. $v_k \in \mathbf{Z}_\beta$).

Example 5.14.

Let $m_0 = 3$, $m_1 = 5$, and $m = m_0 m_1 = 15$. Using the positive consistent representation, the integer $u = 11 \in \mathbf{Z}_{15}$ has the unique mixed radix representation

$$11 = v_0 + v_1(3)$$

with $v_0 = 2 \in \mathbf{Z}_3$ and $v_1 = 3 \in \mathbf{Z}_5$. Using the symmetric consistent representation, the integer $\tilde{u} = -4 \in \mathbf{Z}_{15}$ has the unique mixed radix representation

$$-4 = \tilde{v}_0 + \tilde{v}_1(3)$$

with $\tilde{v}_0 = -1 \in \mathbf{Z}_3$ and $\tilde{v}_1 = -1 \in \mathbf{Z}_5$. Note that $u = 11$ and $\tilde{u} = -4$ are simply two different representations for the same element in $\mathbf{Z}_{15}$ but that the corresponding coefficients v_1 and $\tilde{v}_1$ are *not* simply two different representations for the same element in $\mathbf{Z}_5$. □

Writing the solution u of the system of congruences (14) in the mixed radix representation (15), it is easy to determine formulas for the coefficients v_k $(0 \leqslant k \leqslant n)$ appearing in (15). It is obvious from (15) that

$$u \equiv v_0 \pmod{m_0}$$

and therefore the case $i = 0$ of the system of congruences (14) will be satisfied if v_0 is chosen such that

(16) $v_0 \equiv u_0 \pmod{m_0}$.

In general for $k \geqslant 1$, if coefficients $v_0, v_1, \ldots, v_{k-1}$ have been determined then noting from (15) that

$$u \equiv v_0 + v_1(m_0) + \cdots + v_k \left(\prod_{i=0}^{k-1} m_i\right) \pmod{m_k},$$

we can satisfy the case $i = k$ of the system of congruences (14) by choosing v_k such that

$$v_0 + v_1(m_0) + \cdots + v_k \left(\prod_{i=0}^{k-1} m_i\right) \equiv u_k \pmod{m_k}.$$

Using properties (8) - (11) to solve this congruence equation for v_k we get for $k \geqslant 1$:

$$\textbf{(17)} \quad v_k \equiv \left[u_k - \left[v_0 + v_1(m_0) + \cdots + v_{k-1}\left(\prod_{i=0}^{k-2} m_i\right)\right]\right] \left[\prod_{i=0}^{k-1} m_i\right]^{-1} \pmod{m_k}$$

where the inverse appearing here is valid because $\prod_{i=0}^{k-1} m_i$ is relatively prime to m_k. Finally we note that once a consistent representation has been chosen, there is a *unique* integer $v_0 \in \mathbf{Z}_{m_0}$ satisfying (16) (namely $v_0 = u_0 \in \mathbf{Z}_{m_0}$) and similarly for $k = 1, 2, \ldots, n$ there is a *unique* integer $v_k \in \mathbf{Z}_{m_k}$ satisfying (17).

Implementation Details for Garner's Algorithm

Garner's algorithm is presented formally as Algorithm 5.1. Some details about the implementation of this algorithm need further discussion. It is important to note that in the usual applications of Garner's algorithm the moduli m_i $(0 \leqslant i \leqslant n)$ are single-precision integers (typically, large single-precision integers) and therefore the residues u_i $(0 \leqslant i \leqslant n)$ are also single-precision integers. The integer u being computed will be a multiprecision integer and indeed the list of residues $(u_0, u_1, \ldots, u_n)$ can be viewed simply as a different representation for the multiprecision integer u (see chapter 4). Algorithm 5.1 is organized so that in this typical situation operations on multiprecision integers are completely avoided until the last step. In particular we use the notation ϕ_{m_k} in Algorithm 5.1 in a manner that is consistent with its mathematical meaning as a modular homomorphism but we give it the following more precise algorithmic specification:

$\phi_{m_k}(expression)$ means 'evaluate *expression* in the ring $\mathbf{Z}_{m_k}$'.

More specifically, it means that when *expression* is decomposed into a sequence of binary operations, the intermediate result of *each* binary operation is to be reduced modulo m_k before proceeding with the evaluation of *expression*. In this way we are guaranteed that every variable (except of course u) appearing in Algorithm 5.1 is a single-precision variable and moreover that every operation appearing in step 1 and step 2 is an operation on single-precision integers. (Note however that if a and b are single-precision integers then the operation $\phi_{m_k}(a \times b)$, for example, is usually performed by an ordinary integer multiplication $a \times b$ yielding a double-precision integer, say c, followed by an integer division operation to compute rem(c, m_k)).

For $k = 1, 2, \ldots, n$ the integer v_k satisfying (17) is computed in step 2 of Algorithm 5.1 by evaluating the right hand side of (17) in the ring $\mathbf{Z}_{m_k}$. The inverses appearing in (17):

$$\gamma_k = \left(\prod_{i=0}^{k-1} m_i\right)^{-1} \pmod{m_k}, \quad \text{for } k = 1, 2, \ldots, n$$

are all computed in step 1. Note that a method for implementing the procedure

reciprocal(a, q)

Algorithm 5.1. Garner's Chinese Remainder Algorithm.

begin

comment Given positive moduli $m_i \in \mathbf{Z}$ $(0 \leqslant i \leqslant n)$ which are pairwise relatively prime and given corresponding residues $u_i \in \mathbf{Z}_{m_i}$ $(0 \leqslant i \leqslant n)$, compute the unique integer $u \in \mathbf{Z}_m$ (where $m = \prod_{i=0}^{n} m_i$) such that

$u \equiv u_i \pmod{m_i}$, $i = 0, 1, , \ldots, n$.

Consistent representations for $\mathbf{Z}_{m_i}$ $(0 \leqslant i \leqslant n)$ and $\mathbf{Z}_m$ are assumed;

comment Step 1: Compute the required inverses using a procedure reciprocal(a,q) which computes $a^{-1} \pmod{q}$. Note that this step of the algorithm should be separately precomputed if the algorithm is to be used repeatedly with same set $\{m_i\}$ of moduli;

for $k \leftarrow 1$ **until** n **do**
begin
product $\leftarrow \phi_{m_k}(m_0)$;
for $i \leftarrow 1$ **until** $k-1$ **do**
product $\leftarrow \phi_{m_k}(\text{product} \times m_i)$;
$\gamma_k \leftarrow$ reciprocal(product,m_k)
end;

comment Step 2: Compute the mixed radix coefficients $\{v_k\}$;

$v_0 \leftarrow u_0$;
for $k \leftarrow 1$ **until** n **do**
begin
temp $\leftarrow v_{k-1}$;

for $j \leftarrow k-2$ **step** -1 **until** 0 **do**
temp $\leftarrow \phi_{m_k}(\text{temp} \times m_j + v_j)$;

$v_k \leftarrow \phi_{m_k}((u_k - \text{temp}) \times \gamma_k)$
end;

comment Step 3: Convert from mixed radix representation to standard representation;

$u \leftarrow v_n$;

for $k \leftarrow n-1$ **step** -1 **until** 0 **do**
$u \leftarrow u \times m_k + v_k$

end.

to compute $a^{-1} \pmod{q}$ for relatively prime a and q, is given in the proof of Theorem 5.6; namely, apply the extended Euclidean algorithm (Algorithm 2.2) to $a,q \in \mathbf{Z}$ yielding integers s and t such that

$$sa + tq = 1$$

and then $\phi_q(s) = \text{rem}(s, q)$ is the desired inverse in the ring $\mathbf{Z}_q$. The computation of the inverses $\{\gamma_k\}$ was purposely separated from the rest of the computation in Algorithm 5.1 because $\{\gamma_k\}$ depend only on the moduli $\{m_i\}$. For typical applications of Garner's algorithm in a system using the modular representation for multiprecision integers, the moduli $\{m_i\}$ would be fixed so that step 1 would be removed from Algorithm 5.1 and the inverses $\{\gamma_k\}$ would be given to the algorithm as precomputed constants. It is also worth noting that there are situations when both step 1 and step 3 would be removed from Algorithm 5.1. For example, in the above-mentioned setting if it is desired to compare two multiprecision integers a and b represented in their modular representations then it is sufficient to compute their (single-precision) mixed radix coefficients and compare them: (cf. [Knu69]).

Finally, step 3 needs some justification. We have stated that if consistent representations are used for $\mathbf{Z}_{m_k}$ $(0 \leqslant k \leqslant n)$ and $\mathbf{Z}_m$ then the mixed radix representation (15) for $u \in \mathbf{Z}_m$ is unique. However we have not shown that if the operations in (15) are performed in the ring $\mathbf{Z}$ rather than in the ring $\mathbf{Z}_m$, we will still obtain the unique $u \in \mathbf{Z}_m$ as desired – i.e. in step 3 of Algorithm 5.1 there is no need to write the **for**-loop statement as

$$u \leftarrow \phi_m(u \times m_k + v_k).$$

To justify this, note from (15) that if $|\, v_k \,| \leqslant \dfrac{m_k - 1}{2}$ for $k = 0, 1, \ldots, n$ (i.e. if the symmetric consistent representation is used) then

$$\begin{aligned} |\, u \,| &\leqslant \frac{m_0 - 1}{2} + \frac{m_1 - 1}{2}(m_0) + \cdots + \frac{m_n - 1}{2}\left(\prod_{i=0}^{n-1} m_i\right) \\ &\leqslant \frac{1}{2}\left[\left(\prod_{i=0}^{n} m_i\right) - 1\right] \end{aligned}$$

proving that u lies in the correct range. Similarly if $0 \leqslant v_k \leqslant m_k - 1$ for $k = 0, 1, \ldots, n$ (i.e. if the positive consistent representation is used) then clearly $u \geqslant 0$ and, proceeding as above,

$$u \leqslant \left(\prod_{i=0}^{n} m_i\right) - 1$$

proving again that u lies in the correct range. Finally, step 3 performs the evaluation of (15) using the method of nested multiplication:

$$u = v_0 + m_0(v_1 + m_1(v_2 + \cdots + m_{n-2}(v_{n-1} + m_{n-1}(v_n)) \cdots)).$$

Example 5.15.

Suppose that the single-precision integers on a particular computer are restricted to the range $-100 < a < 100$ (i.e. two-digit integers). Consider as moduli the three largest single-precision integers which are odd and pairwise relatively prime:

$$m_0 = 99; \quad m_1 = 97; \quad m_2 = 95.$$

Then $m = m_0 m_1 m_2 = 912285$. Using the symmetric consistent representation, the range of integers in $\mathbf{Z}_{912285}$ is

$$-456142 \leqslant u \leqslant 456142.$$

Now consider the problem of determining u given that:

$u \equiv 49 \pmod{99}$;
$u \equiv -21 \pmod{97}$;
$u \equiv -30 \pmod{95}$.

Applying Algorithm 5.1, we compute in step 1 the following inverses:

$$\gamma_1 = m_0^{-1} \pmod{m_1} = 99^{-1} \pmod{97} = 2^{-1} \pmod{97} = -48;$$

$$\gamma_2 = (m_0 m_1)^{-1} \pmod{m_2} = 8^{-1} \pmod{95} = 12.$$

Carrying out the computation of step 2, we get the following mixed radix coefficients for u:

$$v_0 = 49; \quad v_1 = -35; \quad v_2 = -28.$$

At this point we have the following mixed radix representation for u:

$$u = 49 - 35(99) - 28(99)(97).$$

Finally, carrying out the conversion of step 3 using 'multiprecision' arithmetic we find

$$u = -272300. \quad \square$$

5.5. THE POLYNOMIAL INTERPOLATION ALGORITHM

We now consider the corresponding inversion process for evaluation homomorphisms. Recall that we are primarily interested in homomorphisms $\phi_{<I,p>}$ which project the multivariate polynomial domain $\mathbf{Z}[x_1, \ldots, x_v]$ onto the Euclidean domain $\mathbf{Z}_p[x_1]$ (or perhaps onto the field $\mathbf{Z}_p$). In the notation $\phi_{<I,p>}$, p denotes a prime integer, I denotes the kernel of a multivariate evaluation homomorphism, and $\phi_{<I,p>}$ denotes the composite homomorphism $\phi_I \phi_p$ with domains of definition indicated by:

$$\textbf{(18)} \quad \phi_p : \mathbf{Z}[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_p[x_1, \ldots, x_v]$$

and

$$\textbf{(19)} \quad \phi_I : \mathbf{Z}_p[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_p[x_1]$$

(or the homomorphic image in (19) could as well be $\mathbf{Z}_p$). The inversion process for homomorphisms of the form (18) is the Chinese remainder algorithm of the preceding section. (Note that Garner's Chinese remainder algorithm can be applied coefficient-by-coefficient in the polynomial case, with the polynomials expressed in expanded canonical form). The inversion process for homomorphisms of the form (19) is the problem of polynomial interpolation.

The Polynomial Interpolation Problem

The inversion of multivariate evaluation homomorphisms of the form (19) will be accomplished one indeterminate at a time, viewing ϕ_I in the natural way as a composition of univariate evaluation homomorphisms. Therefore it is sufficient to consider the inversion of univariate evaluation homomorphisms of the form

$$\phi_{x-\alpha_i} : \mathrm{D}[x] \rightarrow \mathrm{D}$$

where D is (in general) a multivariate polynomial domain over a field $\mathbf{Z}_p$ and where $\alpha_i \in \mathbf{Z}_p$. It will be important computationally that α_i lies in the field $\mathbf{Z}_p$.

The development of an algorithm for polynomial interpolation will directly parallel the development of Garner's algorithm for the integer Chinese remainder problem. Indeed it should become clear that the two processes are identical if one takes an appropriately abstract (ring-theoretic) point of view. In particular, by paraphrasing the statement of the integer Chinese remainder problem we get the following statement of the *polynomial interpolation problem:*

Let D be a domain of polynomials (in zero or more indeterminates other than x) over a coefficient field $\mathbf{Z}_p$. Given *moduli* $x - \alpha_0, x - \alpha_1, \ldots, x - \alpha_n$ where $\alpha_i \in \mathbf{Z}_p$, $0 \leqslant i \leqslant n$, and given corresponding *residues* $u_i \in D$, $0 \leqslant i \leqslant n$, find a polynomial $u(x) \in D[x]$ such that

(20) $u(x) \equiv u_i \pmod{x - \alpha_i}$, $0 \leqslant i \leqslant n$.

Note that in this case the congruences (20) are usually stated in the following equivalent form:

(21) $u(\alpha_i) = u_i$, $0 \leqslant i \leqslant n$

and the elements $\alpha_i \in \mathbf{Z}_p$ $(0 \leqslant i \leqslant n)$ are usually called *evaluation points* or *interpolation points*. As in the case of the integer Chinese remainder problem, in order to guarantee that a solution exists we must impose the additional condition that the moduli $\{x - \alpha_i\}$ be pairwise relatively prime. But clearly

$$\mathrm{GCD}(x - \alpha_i, x - \alpha_j) = 1 \text{ if and only if } \alpha_i \neq \alpha_j$$

so the additional condition reduces to the rather obvious condition that the moduli $\{x - \alpha_i\}$ must be *distinct* (i.e. the evaluation points $\{\alpha_i\}$ must be distinct). Also as in the integer Chinese remainder problem, the solution to the polynomial interpolation problem is only unique modulo $\prod_{i=0}^{n} (x - \alpha_i)$, which is to say that the solution is unique if we restrict it to be of degree less than $n+1$.

The following theorem proves the above existence and uniqueness results in a more general setting where the domain D is an arbitrary integral domain and the evaluation points $\{\alpha_i\}$ are arbitrary distinct points in D. However this theorem allows the solution $u(x)$ to lie in $F_D[x]$ rather than in $D[x]$, where F_D denotes the quotient field of the integral domain D. We will then proceed to develop an efficient algorithm for solving the polynomial interpolation problem and it will be obvious that in the particular setting presented above, the solution $u(x)$ will lie in $D[x]$ because the only divisions required will be divisions in the coefficient field $\mathbf{Z}_p$.

Theorem 5.8.

Let D be an arbitrary integral domain, let $\alpha_i \in D$, $i = 0, 1, \ldots, n$ be $n+1$ distinct elements in D, and let $u_i \in D$, $i = 0, 1, \ldots, n$ be $n+1$ specified values in D. There exists a unique polynomial $u(x) \in F_D[x]$ (where F_D is the quotient field of D) which satisfies the following conditions:

(i) $\deg[u(x)] \leqslant n$;

(ii) $u(\alpha_i) = u_i$, $0 \leqslant i \leqslant n$.

Proof:

By condition (i) we may write $u(x)$ in the form

$$u(x) = a_0 + a_1 x + \cdots + a_n x^n$$

where the coefficients $a_i \in F_D$ $(0 \leqslant i \leqslant n)$ are to be determined. Condition (ii) then becomes the following linear system of order $(n+1)$:

$$\mathbf{V}\mathbf{a} = \mathbf{u}$$

where V is the *Vandermonde matrix* with (i,j)-th entry α_i^j $(i,j = 0, 1, \ldots, n)$, $\mathbf{u}$ is the vector with i-th entry u_i $(i = 0, 1, \ldots, n)$, and $\mathbf{a}$ is the vector of unknowns with i-th entry a_i $(i = 0, 1, \ldots, n)$. From elementary linear algebra, this linear system can be solved in the field F_D and the solution is unique if $\det(V) \neq 0$. Employing the classical formula for the Vandermonde determinant:

$$\det(V) = \prod_{0 \leqslant i < k \leqslant n} (\alpha_k - \alpha_i),$$

we see that $\det(V) \neq 0$ because the elements $\alpha_0, \alpha_1, \ldots, \alpha_n \in D$ are distinct. □

The Newton Interpolation Algorithm

The proof of Theorem 5.8 is a constructive proof since we can solve linear equations over the quotient field of an integral domain (see chapter 9). However the solution to the interpolation problem can be computed by an algorithm requiring much less work than solving a system of linear equations. (cf Exercise 5-22). The algorithm we will develop for polynomial interpolation dates back to Newton in the 17th century. As with Garner's algorithm, the key to the development is to express the solution $u(x) \in F_D[x]$ in the following mixed radix representation (sometimes called the *Newton form* or the *divided-difference form):*

$$\textbf{(22)} \quad u(x) = v_0 + v_1(x - \alpha_0) + v_2(x - \alpha_0)(x - \alpha_1) + \cdots + v_n \prod_{i=0}^{n-1} (x - \alpha_i)$$

where the *Newton coefficients* $v_k \in F_D\ (0 \leqslant k \leqslant n)$ are to be determined. The justification for this mixed radix representation is the fact from elementary linear algebra that *any* set of polynomials $m_k(x) \in F_D[x]$, $k = 0, 1, \ldots, n$ with $\deg[m_k(x)] = k$ forms a valid *basis* for polynomials of degree n in x over the field F_D; in this case we are choosing $m_0(x) = 1$, $m_k(x) = \prod_{i=0}^{k-1} (x - \alpha_i)$ for $k = 1, 2, \ldots, n$.

The Newton interpolation algorithm can be developed for the general setting of Theorem 5.8 in which case the Newton coefficients $\{v_k\}$ in (22) will be quotients of elements in D (called *divided-differences*). However we will develop the algorithm for the case of practical interest to us, namely the setting indicated in the preamble to (20). In this case no quotients of elements (polynomials) in D will arise since the only divisions which arise will be divisions (i.e. multiplications by inverses) in the coefficient field $\mathbf{Z}_p$ of the polynomial domain D.

Writing the solution $u(x)$ in the Newton form (22) we apply the conditions (21) to obtain formulas for the Newton coefficients v_k $(0 \leqslant k \leqslant n)$. It is obvious from (22) that

$$u(\alpha_0) = v_0$$

and therefore the case $i = 0$ of the conditions (21) will be satisfied if v_0 is chosen to be

$$\textbf{(23)} \quad v_0 = u_0$$

In general for $k \geqslant 1$, if the Newton coefficients $v_0, v_1, \ldots, v_{k-1}$ have been determined then noting from (22) that

$$u(\alpha_k) = v_0 + v_1(\alpha_k - \alpha_0) + \cdots + v_k \prod_{i=0}^{k-1} (\alpha_k - \alpha_i),$$

the case $i = k$ of the conditions (21) will be satisfied if v_k is chosen such that

$$v_0 + v_1(\alpha_k - \alpha_0) + \cdots + v_k \prod_{i=0}^{k-1} (\alpha_k - \alpha_i) = u_k.$$

Now since $\alpha_i \in \mathbf{Z}_p\ (0 \leqslant i \leqslant n)$ we can compute in the field $\mathbf{Z}_p$ the inverse of the nonzero element $\prod_{i=0}^{k-1} (\alpha_k - \alpha_i) \in \mathbf{Z}_p$, using once again the extended Euclidean algorithm since any nonzero integer in $\mathbf{Z}_p$ is relatively prime (in $\mathbf{Z}$) to the prime integer p. Solving for v_k, we get for $k \geqslant 1$:

$$\textbf{(24)} \quad v_k = \left[u_k - [v_0 + v_1(\alpha_k - \alpha_0) + \cdots + v_{k-1} \prod_{i=0}^{k-2} (\alpha_k - \alpha_i)] \right] \left[\prod_{i=0}^{k-1} (\alpha_k - \alpha_i) \right]^{-1}.$$

It is important to note that $u_k\ (0 \leqslant k \leqslant n)$ and $v_k\ (0 \leqslant k \leqslant n)$ will be, in general, multivariate polynomials in a domain D with coefficients lying in a field $\mathbf{Z}_p$ and all coefficient arithmetic arising in equation (24) will be performed in the field $\mathbf{Z}_p$.

The Newton interpolation algorithm is presented formally as Algorithm 5.2. Comparing with Algorithm 5.1 it can be seen that the two algorithms are statement-by-statement identical except for computational details. As with Garner's algorithm, the Newton interpolation algorithm is divided into three steps. Step 1 again could be removed and precomputed if the evaluation points $\{\alpha_i\}$ are fixed, although in the multivariate case the computational cost of step 1 will be insignificant compared with step 2 (because step 1 involves only operations on integers in $\mathbf{Z}_p$ while step 2 involves operations on polynomials in $\mathbf{Z}_p[y]$). The notation ϕ_p has an algorithmic specification as before:

ϕ_p(*expression*) means 'evaluate *expression* with all operations on integers being performed modulo p'.

Note that in Algorithm 5.2 *all* coefficient arithmetic is to be performed modulo p (i.e. in the field $\mathbf{Z}_p$).

Example 5.16.

Let us determine the polynomial $u(x,y) \in \mathbf{Z}_{97}[x,y]$ of maximum degree 2 in x and maximum degree 1 in y specified by the following values in the field $\mathbf{Z}_{97}$:

$$u(0,0) = -21; \quad u(0,1) = -30;$$

$$u(1,0) = 20; \quad u(1,1) = 17;$$

$$u(2,0) = -36; \quad u(2,1) = -31.$$

Let us first reconstruct the image of $u(x,y)$ in $\mathbf{Z}_{97}[x,y]/\langle x - 0\rangle$ (i.e. the case $x = 0$). In the notation of Algorithm 5.2 we have $D = \mathbf{Z}_{97}$, $\alpha_0 = 0, \alpha_1 = 1, u_0 = -21, u_1 = -30$, and we are computing a polynomial $u(0,y) \in \mathbf{Z}_{97}[y]$ (i.e. the indeterminate x in Algorithm 5.2 is y for now). Step 1 is in this case trivial:

$$\gamma_1 = (\alpha_1 - \alpha_0)^{-1} (\text{mod } 97) = 1^{-1} (\text{mod } 97) = 1.$$

Step 2 computes the following Newton coefficients for $u(0,y)$:

$$v_0 = -21; \quad v_1 = -9;$$

and therefore in step 3 we find

$$u(0,y) = -21 - 9(y - 0) = -9y - 21.$$

Similarly, reconstructing the images of $u(x,y)$ in $\mathbf{Z}_{97}[x,y]/\langle x - 1\rangle$ and $\mathbf{Z}_{97}[x,y]/\langle x - 2\rangle$ we find

$$u(1,y) = -3y + 20;$$

$$u(2,y) = 5y - 36.$$

Now for the multivariate step, we apply Algorithm 5.2 with $D = \mathbf{Z}_{97}[y]$, $\alpha_0 = 0$, $\alpha_1 = 1$, $\alpha_2 = 2$, $u_0 = u(0,y), u_1 = u(1,y), u_2 = u(2,y)$, and we compute the polynomial $u(x,y) \in D[x] = \mathbf{Z}_{97}[y][x]$. Step 1 in this case computes the following inverses:

$$\gamma_1 = (\alpha_1 - \alpha_0)^{-1} (\text{mod } 97) = 1^{-1} (\text{mod } 97) = 1;$$

$$\gamma_2 = [(\alpha_2 - \alpha_0)(\alpha_2 - \alpha_1)]^{-1} (\text{mod } 97) = 2^{-1} (\text{mod } 97) = -48.$$

Step 2 computes the following Newton coefficients:

$$v_0 = -9y - 21;$$

$$v_1 = 6y + 41;$$

$$v_2 = y.$$

Finally in step 3 we find

$$u(x, y) = (-9y - 21) + (6y + 41)(x - 0) + y(x - 0)(x - 1)$$

Algorithm 5.2. Newton Interpolation Algorithm.

```
begin
    comment Let D = Z_p[y] denote a domain of polynomials
        in v ⩾ 0 indeterminates y = (y_1, . . . , y_v) over a
        finite field Z_p (D = Z_p in case v = 0). Given distinct
        evaluation points α_i ∈ Z_p (0 ⩽ i ⩽ n) and given
        corresponding values u_i ∈ D (0 ⩽ i ⩽ n), compute the
        unique polynomial u(x) ∈ D[x] such that deg[u(x)] ⩽ n and
            u(α_i) = u_i, i = 0, 1, . . . , n;

    comment Step 1: Compute the required inverses using a
        procedure reciprocal(a,q) which computes a^-1 (mod q);

    for k ← 1 until n do
        begin
            product ← φ_p(α_k − α_0);
            for i ← 1 until k−1 do
                product ← φ_p(product × (α_k − α_i));

            γ_k ← reciprocal(product,p)
        end;

    comment Step 2: Compute the Newton coefficients {v_k};

    v_0 ← u_0;
    for k ← 1 until n do
        begin
            temp ← v_{k−1};
            for j ← k−2 step −1 until 0 do
                temp ← φ_p(temp × (α_k − α_j) + v_j);

            v_k ← φ_p((u_k − temp) × γ_k)
        end;

    comment Step 3: Convert from Newton form to standard form;

        u ← v_n;
        for k ← n−1 step −1 until 0 do
            u ← φ_p(u × (x − α_k) + v_k)

end.
```

$= x^2y + 5xy + 41x - 9y - 21$

which is the desired polynomial in the domain $\mathbf{Z}_{97}[x,y]$. □

5.6. FURTHER DISCUSSION OF THE TWO ALGORITHMS

Integer and Polynomial Representations

It is important in some circumstances to recognize that in each of Algorithms 5.1 and 5.2, three different representations arise for the same object. In the polynomial case (Algorithm 5.2), the polynomial $u(x) \in D[x]$ is initially represented uniquely by its $n + 1$ *values (residues)* $\{u_0, u_1, \ldots, u_n\}$ corresponding to the $n + 1$ distinct evaluation points $\{\alpha_0, \alpha_1, \ldots, \alpha_n\}$. At the end of step 2, the polynomial $u(x)$ is represented uniquely in *Newton form* by its $n + 1$ Newton coefficients $\{v_0, v_1, \ldots, v_n\}$ with respect to the basis polynomials

$$1, (x - \alpha_0), (x - \alpha_0)(x - \alpha_1), \ldots, \prod_{i=0}^{n-1} (x - \alpha_i).$$

In step 3 the Newton form of $u(x)$ is converted to *standard polynomial form*, which can be characterized as uniquely representing $u(x)$ by its $n + 1$ coefficients $\{a_0, a_1, \ldots, a_n\}$ with respect to the standard basis polynomials $1, x, x^2, \ldots, x^n$. Similarly in the integer Chinese remainder case (Algorithm 5.1), the initial representation for the integer u is by its $n + 1$ *residues* $\{u_0, u_1, \ldots, u_n\}$ with respect to the $n + 1$ moduli $\{m_0, m_1, \ldots, m_n\}$. The second representation is the *mixed radix representation* $\{v_0, v_1, \ldots, v_n\}$ with respect to the mixed radices

$$1, m_0, m_0 m_1, \ldots, \prod_{i=0}^{n-1} m_i.$$

The final step converts the mixed radix representation to the more familiar *radix β representation* where the base β depends on the representation being used for multiprecision integers (see chapter 3).

Residue representations of integers and of polynomials arise (by the application of homomorphisms) because some operations are easier to perform in this representation. For example, multiplication of integers or of polynomials is a simpler operation when residue representations are used than when standard representations are used. The conversion processes of Algorithms 5.1 and 5.2 are required not only because the human computer user will generally want to see his answers presented in the more standard representations of objects but also because some required operations cannot be performed directly on the residue representations. For example, the result of the comparison 'Is $u < v$?' where u and v are integers cannot be determined directly from knowledge of the residue representations of u and v, but as previously noted the result of such a comparison can be directly determined by comparing the mixed radix coefficients of u and v. As another example, if a polynomial $u(x)$ is to be evaluated for arbitrary values of x then the residue representation of $u(x)$ is not appropriate, but $u(x)$ can be evaluated in the Newton form as well as in standard polynomial form. These two examples indicate circumstances where conversions from residue representations are required but where step 3 of the algorithms may be considered to be unnecessary extra computations. However in the context of applying Algorithms 5.1 and 5.2 to the inversion of composite modular/evaluation homomorphisms on the polynomial domain $\mathbf{Z}[x_1, \ldots, x_v]$ (which is the context of primary interest in this book), step 3 of the algorithms will always be applied. The reason for this in the polynomial case (i.e. Algorithm 5.2) will be explained shortly. In the integer case (i.e. Algorithm 5.1) the reason is simply that subsequent operations on the integer coefficients (whether output operations or arithmetic operations) will require the standard integer representation. For output this is a user requirement, while for arithmetic operations there is no practical advantage in requiring a system to support arithmetic operations on integers in more than one representation. (Of course it is conceivable to have a system in which the 'standard' integer representation is not a radix β representation but all of the present-day systems of interest to us use a radix β representation for integers).

Another issue which arises in the practical application of modular and evaluation homomorphisms and their corresponding inversion algorithms is to determine the number of moduli (evaluation points) needed to uniquely represent an unknown integer (polynomial). In the polynomial case, the information needed is an upper bound D for the degree of the result since then D+1 moduli (evaluation points) are sufficient to uniquely represent the polynomial result. Similarly in the integer case, if an upper bound M for the magnitude of the integer result is known

then by choosing enough moduli $\{m_i\}$ such that

$$m = \prod_{i=0}^{n} m_i > 2M,$$

we are guaranteed that the ring $\mathbf{Z}_m$ is large enough to represent the integer result. In other words, the result determined by Algorithm 5.1 lies in the ring $\mathbf{Z}_m$ and this result will be the same (when expressed in the symmetric representation) as the desired result in $\mathbf{Z}$, since

$$\mathbf{Z}_m = \left\{ -\frac{m-1}{2}, \ldots, -1, 0, 1, \ldots, \frac{m-1}{2} \right\}$$

with $\frac{m-1}{2} \geqslant M$. Such polynomial degree bounds and integer magnitude bounds can usually be calculated with little effort for practical problems, although it is quite common for easily-calculated bounds to be very pessimistic (i.e. much larger than necessary). An alternate computational approach is available in situations where it is easy to verify the correctness of the result. This alternate approach is based on the observation that the mixed radix coefficients (or Newton coefficients) $\{v_k\}$ will be zero for $k > \mathrm{K}$ if the moduli $m_0, m_1, \ldots, m_{\mathrm{K}}$ (or $x - \alpha_0, x - \alpha_1, \ldots, x - \alpha_{\mathrm{K}}$) are sufficient to uniquely represent the result. The computation therefore can be halted once $v_{\mathrm{K}+1} = 0$ for some K (on the assumption that $v_k = 0$ for all $k > \mathrm{K}$) as long as the result is later verified to be correct.

A Generalization of Garner's Algorithm

There is a slight generalization of Garner's Chinese remainder algorithm which is useful in the applications of interest to us. Recall that we wish to invert composite homomorphisms $\phi_{<I_{ij}, p_i>} = \phi_{1_{ij}} \phi_{p_i}$ where

$$\textbf{(25)} \quad \phi_{p_i} : \mathbf{Z}[x_1, \ldots, x_v] \rightarrow \mathbf{Z}_{p_i}[x_1, \ldots, x_v], \; i = 0, 1, \ldots, n$$

is a sequence of modular homomorphisms for some chosen prime moduli $p_0, p_1, \ldots, p_n$, and for *each* i there is a corresponding sequence of some N multivariate evaluation homomorphisms

$$\textbf{(26)} \quad \phi_{1_{ij}} : \mathbf{Z}_{p_i}[x_i, \ldots, x_v] \rightarrow \mathbf{Z}_{p_i}[x_1], \; j = 1, 2, \ldots, \mathrm{N}$$

with kernels of the form $I_{ij} = <x_2 - \alpha_{2ij}, \ldots, x_v - \alpha_{vij}>$. In this notation, for a fixed i the evaluation points $\alpha_{2ij}, \ldots, \alpha_{vij}$ all lie in the field $\mathbf{Z}_{p_i}$ and the number N of different kernels I_{ij} is determined by the degree of the solution in each indeterminate. Now suppose that Algorithm 5.2 is applied (as in Example 5.16) to invert the evaluation homomorphisms (26) and suppose that the $n + 1$ polynomials which arise are $u_i(\mathbf{x}) \in \mathbf{Z}_{p_i}[x_1, \ldots, x_v]$, for $i = 0, 1, \ldots, n$. If the polynomials $u_i(\mathbf{x})$ are all expressed in expanded canonical form then Algorithm 5.1 can be applied coefficient by coefficient to reconstruct the coefficients of the desired solution $\mathrm{u}(\mathbf{x}) \in \mathbf{Z}[x_1, \ldots, x_v]$ (i.e. to invert the modular homomorphisms (25)).

The desired generalization of Garner's algorithm is obtained by simply noting that Algorithm 5.1 can be applied directly to the polynomials $u_i(\mathbf{x})$ to reconstruct $\mathrm{u}(\mathbf{x})$, rather than being applied many times separately for each coefficient of the polynomial $\mathrm{u}(\mathbf{x})$. To see this, suppose $\mathrm{u}(\mathbf{x})$ is the polynomial

$$\mathrm{u}(\mathbf{x}) = \sum_{\mathbf{e}} u_{\mathbf{e}} \mathbf{x}^{\mathbf{e}} \in \mathbf{Z}[\mathbf{x}]$$

with images

$$u_i(\mathbf{x}) = \sum_{\mathbf{e}} u_{\mathbf{e},i} \mathbf{x}^{\mathbf{e}} \in \mathbf{Z}_{p_i}[\mathbf{x}], \; i = 0, 1, \ldots, n$$

where $u_{\mathbf{e},i} = \phi_{p_i}(u_{\mathbf{e}})$. If Algorithm 5.1 is applied separately for each coefficient $u_{\mathbf{e}}$, it calculates (in step 2) each integer $u_{\mathbf{e}}$ in its mixed radix representation

$$u_{\mathbf{e}} = \sum_{k=0}^{n} v_{\mathbf{e},k} \left(\prod_{j=0}^{k-1} p_j \right)$$

where $v_{\mathbf{e},k} \in \mathbf{Z}_{p_k}$, $0 \leqslant k \leqslant n$. But since the same mixed radices appear in the mixed radix representations for each different coefficient $u_{\mathbf{e}}$ we may express the polynomial $\mathbf{u(x)}$ as follows:

$$\begin{aligned} \mathbf{u(x)} &= \sum_{\mathbf{e}} \left(\sum_{k=0}^{n} v_{\mathbf{e},k} \left(\prod_{j=0}^{k-1} p_j \right) \right) \mathbf{x}^{\mathbf{e}} \\ &= \sum_{k=0}^{n} \left(\sum_{\mathbf{e}} v_{\mathbf{e},k} \mathbf{x}^{\mathbf{e}} \right) \left(\prod_{j=0}^{k-1} p_j \right) \end{aligned}$$

The latter expression for the polynomial $\mathbf{u(x)}$ is called a *polynomial mixed radix representation* with respect to the mixed radices $1, p_0, p_0p_1, \ldots, \prod_{j=0}^{n-1} p_j$, and its general form is

$$\mathbf{u(x)} = v_0(\mathbf{x}) + v_1(\mathbf{x})(p_0) + v_2(\mathbf{x})(p_0p_1) + \cdots + v_n(\mathbf{x}) \left(\prod_{j=0}^{n-1} p_j \right)$$

where $v_k(\mathbf{x}) \in \mathbf{Z}_{p_k}[\mathbf{x}]$ for $k = 0, 1, \ldots, n$. It can be seen that step 2 of Algorithm 5.1 will directly generate the polynomial $\mathbf{u(x)}$ in its polynomial mixed radix representation if we simply change the specification of Algorithm 5.1 to allow the residues to be polynomials $u_i(\mathbf{x}) \in \mathbf{Z}_{p_i}[\mathbf{x}]$ $(0 \leqslant i \leqslant n)$. Note that step 3 of Algorithm 5.1 also remains valid to convert the polynomial to its standard representation as a polynomial with integer coefficients. The validity of this *generalized Garner's algorithm* follows immediately from the fact that the operations of multiplying a polynomial by a constant and of adding two polynomials are by definition coefficient-by-coefficient operations. This generalization can be viewed simply as a method for computing 'in parallel' the separate Chinese remainder processes for each coefficient of the polynomial solution $\mathbf{u(x)}$.

The generalized Garner's algorithm is only valid if all of the polynomial residues $u_i(\mathbf{x})$, $0 \leqslant i \leqslant n$ are expressed in expanded canonical form for only then can we be assured that the operations in the algorithm are the correct coefficient-by-coefficient operations. Since the given polynomial residues $u_i(\mathbf{x}) \in \mathbf{Z}_{p_i}[\mathbf{x}]$ will usually result from a previous interpolation step, it is worth noting in particular why the polynomials cannot be left in Newton form. The reason is that the basis polynomials for the Newton form of one polynomial residue $u_i(\mathbf{x}) \in \mathbf{Z}_{p_i}[\mathbf{x}]$ involve evaluation points lying in the field $\mathbf{Z}_{p_i}$ while the basis polynomials for the Newton form of a different polynomial residue $u_j(\mathbf{x}) \in \mathbf{Z}_{p_j}[\mathbf{x}]$ involve evaluation points lying in the different field $\mathbf{Z}_{p_j}$. There is in general no consistent interpretation of these various polynomial residues as images of the solution $\mathbf{u(x)}$ unless each polynomial residue $u_i(\mathbf{x}) \in \mathbf{Z}_{p_i}[\mathbf{x}]$ is first converted to expanded canonical form in its own domain $\mathbf{Z}_{p_i}[\mathbf{x}]$. The basis polynomials for the expanded canonical form are independent of the evaluation points. This explains why step 3 of Algorithm 5.2 is an essential step of the Newton interpolation algorithm in the context of inverting composite modular/evaluation homomorphisms.

Example 5.17.

Let us complete the details of the process of inverting the homomorphisms used in Example 5.7 at the end of section 5.1. The problem was to determine the product polynomial

$$c(x) = a(x)\,b(x) = (7x + 5)(2x - 3) \in \mathbf{Z}[x].$$

To determine the number of evaluation homomorphisms to use, note that $\deg[c(x)] = \deg[a(x)] + \deg[b(x)] = 2$ so that three evaluation points will be sufficient. For a bound on the magnitudes of the integers in the product $c(x)$, it is easy to see that the product of two linear polynomials $a(x)$ and $b(x)$ will have coefficients bounded in magnitude by

$$M = 2a_{\max}b_{\max}$$

where a_{max} and b_{max} are the magnitudes of the largest coefficients in a(x) and b(x) respectively. Thus $M = 42$ in this example so it will be sufficient to use moduli such that

$$m = \prod_{i=0}^{n} m_i > 84.$$

In example 5.7 it was seen that the three composite homomorphisms

$$\phi_{x-\alpha_i}\phi_5 : \mathbf{Z}[x] \rightarrow \mathbf{Z}_5, \text{ where } \alpha_0 = 0,\ \alpha_1 = 1,\ \alpha_2 = 2$$

yield the following images in the field $\mathbf{Z}_5$ (when converted to the symmetric representation):

$$c(0) = 0,\ c(1) = -2,\ c(2) = -1.$$

Applying Algorithm 5.2 to this interpolation problem yields

$$c(x) = -x^2 - x \in \mathbf{Z}_5[x].$$

Next the three composite homomorphisms

$$\phi_{x-\alpha_i}\phi_7 : \mathbf{Z}[x] \rightarrow \mathbf{Z}_7, \text{ where } \alpha_0 = 0,\ \alpha_1 = 1,\ \alpha_2 = 2$$

led to the following images in the field $\mathbf{Z}_7$:

$$c(0)=-1;\ c(1)=2;\ c(2)=-2.$$

Applying Algorithm 5.2 to this interpolation problem yields

$$c(x) = 3x - 1 \in \mathbf{Z}_7[x].$$

Now since the moduli $p_0 = 5$ and $p_1 = 7$ do not satisfy $p_0 p_1 > 84$, let us choose also $p_2 = 3$. Then $p_0 p_1 p_2 = 105 > 84$ so these moduli will be sufficient. The three composite homomorphisms

$$\phi_{x-\alpha_i}\phi_3 : \mathbf{Z}[x] \rightarrow \mathbf{Z}_3, \text{ where } \alpha_0 = 0,\ \alpha_1 = 1,\ \alpha_2 = -1$$

yield the following images in the field $\mathbf{Z}_3$:

$$c(0)=0;\ c(1)=0;\ c(-1)=1.$$

Applying Algorithm 5.2 to this interpolation problem yields

$$c(x) = -x^2 + x \in \mathbf{Z}_3[x].$$

Now let us apply the generalized Garner's algorithm to invert the three modular homomorphisms:

$$\phi_{p_i} : \mathbf{Z}[x] \rightarrow \mathbf{Z}_{p_i}[x], \text{ where } p_0 = 5,\ p_1 = 7,\ p_2 = 3.$$

The given residues are

$$u_0(x) = -x^2 - x;\ u_1(x) = 3x - 1;\ u_2(x) = -x^2 + x.$$

The inverses computed in step 1 are:

$$\gamma_1 = p_0^{-1} (\text{mod } p_1) = 5^{-1} (\text{mod } 7) = 3;$$

$$\gamma_2 = (p_0 p_1)^{-1} (\text{mod } p_2) = (-1)^{-1} (\text{mod } 3) = -1.$$

In step 2 the following polynomial mixed radix coefficients are computed:

$$v_0(x) = -x^2 - x;\ v_1(x) = 3x^2 - 2x - 3;\ v_2(x) = 0.$$

Finally in step 3 we find

$$u(x) = (-x^2 - x) + (3x^2 - 2x - 3)(5) + (0)(5)(7)$$

$$= 14x^2 - 11x - 15 \in \mathbf{Z}_{105}[x].$$

Note that the last polynomial mixed radix coefficient $v_2(x)$ is zero which implies that the two

moduli $p_0 = 5$ and $p_1 = 7$ would have been sufficient for this problem. In other words, the bound $M = 42$ on the magnitudes of the integer coefficients in the result is a large overestimate. In any case, we are guaranteed that $u(x)$ is the desired result – i.e.

$$c(x) = 14x^2 - 11x - 15 \in \mathbf{Z}[x]. \quad \square$$

A Homomorphism Diagram

Finally in this chapter, Figure 5.1 presents a ***homomorphism diagram*** which is a convenient way to visualize the computational 'route' to the solution of a problem when homomorphism methods are used. The particular homomorphism diagram expressed in Figure 5.1 is for the case of applying composite modular/evaluation homomorphisms as in (25) - (26) to project the multivariate polynomial domain $\mathbf{Z}[x_1, \ldots, x_v]$ onto Euclidean domains $\mathbf{Z}_{p_i}[x_1]$. Of course, the same diagram is valid if $\mathbf{Z}_{p_i}[x_1]$ is replaced by $\mathbf{Z}_{p_i}$ which would express the case of Example 5.17. Note that for the particular problem considered in Example 5.17 the homomorphism method in fact requires much more work than a 'direct method' (i.e. ordinary polynomial multiplication), which can be expressed in the diagram of Figure 5.1 by drawing an arrow from the 'Given problem' box directly to the 'Desired solution' box. However for problems such as multivariate GCD computation and multivariate polynomial factorization, the 'long route' of homomorphism methods can yield substantial decreases in total computational cost in many cases as we shall see in later chapters.

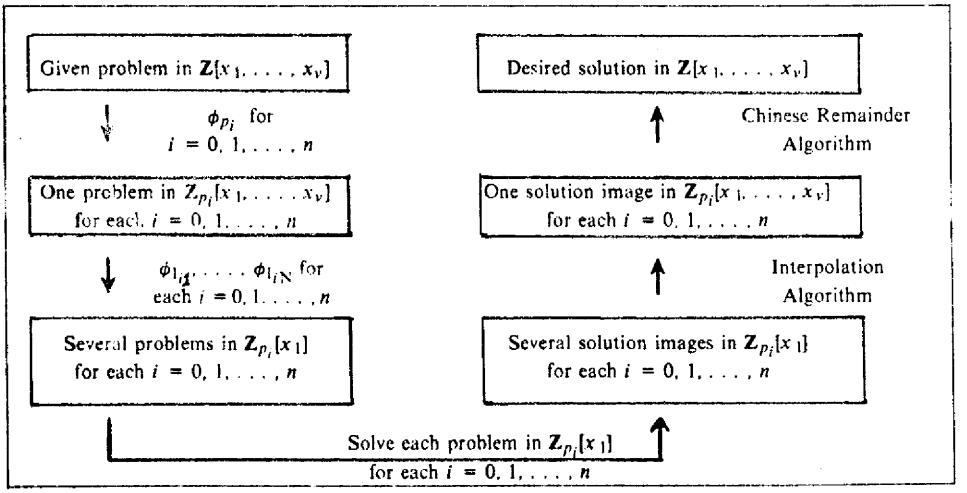

Figure 5.1: Homomorphism diagram for Chinese remainder and interpolation algorithms.

BIBLIOGRAPHY FOR CHAPTER 5

[B&M65] G. Birkhoff and S. MacLane, *A Survey of Modern Algebra*, 3rd ed., Macmillan, New York, 1965.

[Bro71] W.S. Brown, On Euclid's algorithm and the computation of polynomial greatest common divisors, *J. Assoc. Comput. Mach.*, 18 (4), Oct. 1971, pp. 478-504.

[Gar59] H.L. Garner, The residue number system, *IRE Transactions*, EC-8, 1959, pp. 140-147.

[Hor71] E. Horowitz, Modular arithmetic and finite field theory: A tutorial, *Proc. of the Second Symposium on Symbolic and Algebraic Manipulation*, S.R. Petrick (ed.), ACM, New York, 1971, pp. 188-194.

[Knu69] D.E. Knuth, *The Art of Computer Programming, Vol. 2: Seminumerical Algorithms*, Addison-Wesley, Reading, Mass., 1969.

[Lip71] J.D. Lipson, Chinese remainder and interpolation algorithms, *Proc. of the Second Symposium on Symbolic and Algebraic Manipulation*, S.R. Petrick (ed), ACM, New York, 1971, pp. 372-391.

[Lip81] J.D. Lipson, *Elements of Algebra and Algebraic Computing*. To be published, 1981.

[M&B67] S. MacLane and G. Birkhoff, *Algebra*, Macmillan, New York, 1967.

[Wae70] B.L. van der Waerden, *Algebra*, Vol. 1 and Vol. 2, trans. by J.R. Schulenberger, Ungar, New York, 1970.

EXERCISES

5-1. **(a)** Let R and R′ be two rings and let $\phi : R \to R'$ be a ring morphism as defined by Definition 5.2. Properties (i) - (iii) of Definition 5.2 guarantee that ϕ preserves three of the ring operations. Using properties (i) - (iii) and the fact that R and R′ are rings, prove that the other two ring operations are also preserved – i.e. prove that

$\phi(0) = 0$;

$\phi(-a) = -\phi(a)$ for all $a \in R$.

(b) Suppose that the rings R and R′ in part (a) are fields. Prove that the operation of inversion is also preserved by any ring morphism $\phi : R \to R'$ – i.e. prove that

$$\phi(a^{-1}) = [\phi(a)]^{-1} \quad \text{for all } a \in R - \{0\}.$$

(c) Suppose that the ring R is commutative. Prove that if R′ is a homomorphic image of R then R′ is also commutative.

5-2. **(a)** In the integral domain **Z**, give a complete description of each of the following ideals: $\langle 3 \rangle$, $\langle -3 \rangle$, $\langle 4, 6 \rangle$, $\langle 4, 7 \rangle$.

(b) In the polynomial domain $\mathbf{Q}[x]$ let $\alpha \in \mathbf{Q}$ be a fixed constant. The subset $I = \{a(x) : a(\alpha) = 0\}$ is an ideal in $\mathbf{Q}[x]$ (see Example 5.9). Consider the subset $J = \{a(x) : a(\alpha) = 1\}$. Prove or disprove that J is an ideal in $\mathbf{Q}[x]$.

5-3. In any integral domain D, prove that $\langle a \rangle = \langle b \rangle$ if and only if a and b are associates in D.

5-4. In the bivariate polynomial domain $\mathbf{Z}[x, y]$ consider the ideals $I = \langle x, y \rangle$ and $J = \langle x \rangle$. The subset relationships between I, J and $\mathbf{Z}[x, y]$ can be specified as follows:

$$J \subset I \subset \mathbf{Z}[x, y].$$

The ideal I can be described as the set of all bivariate polynomials over **Z** with no constant term and the ideal J can be described as the set of all polynomials in I which have no constant term when expressed as univariate polynomials in x – i.e. when expressed as elements of the domain $\mathbf{Z}[y][x]$.

(a) Express in the usual notation for ideals the following three ideals: the sum $\langle I, J \rangle$, the product $I \cdot J$, and the power I^2.

(b) Specify the subset relationships between $\langle I, J \rangle$, $I \cdot J$, and I^2.

(c) Given a description (in the sense of the descriptions of I and J given above) of each of the ideals $\langle I, J \rangle$, $I \cdot J$, and I^2.

5-5. Let D be a Euclidean domain and let $a, b \in D$ be any two elements. Use Theorem 2.2 (i.e. the extended Euclidean algorithm) to prove that the ideal $\langle a, b \rangle$ generated by these two elements is a principal ideal. More specifically, prove that

$$\langle a, b \rangle = \langle g \rangle$$

where $g = \text{GCD}(a,b)$. (*Remark:* A proof that every Euclidean domain is a principal ideal domain can be based on this result.)

5-6. **(a)** Let D be a principal ideal domain and let $a, b \in D$ be any two elements. Then the ideal $\langle a, b \rangle$ generated by these two elements must be a principal ideal, say

$$\langle a, b \rangle = \langle g \rangle$$

for some element $g \in D$. Prove that g is a greatest common divisor of a and b.

(b) Use the result of part (a) to prove that the extended Euclidean property of Theorem 2.2 holds in any principal ideal domain.

5-7. Show that the domain $\mathbf{Z}[x]$ is not a principal ideal domain by exhibiting an ideal in $\mathbf{Z}[x]$ which is not a principal ideal.

5-8. **(a)** Determine all of the ideals in $\mathbf{Z}_5$. Thus determine all of the homomorphic images of $\mathbf{Z}_5$.

(b) Prove that a field has no proper homomorphic images.

5-9. **(a)** Determine all of the ideals in $\mathbf{Z}_6$. Thus determine all of the homomorphic images of $\mathbf{Z}_6$.

(b) Determine all of the ideals in $\mathbf{Z}_m$, for every integer m. Thus determine all of the homomorphic images of $\mathbf{Z}_m$.

5-10. **(a)** Prove that the only proper homomorphic images of the ring $\mathbf{Z}$ are rings of the form $\mathbf{Z}_m$. (*Hint:* $\mathbf{Z}$ is a principal ideal domain.)

(b) Prove that the quotient ring $\mathbf{Z}_p = \mathbf{Z}/\langle p \rangle$ of the ring $\mathbf{Z}$ is an integral domain if and only if p is a prime integer. (*Hint:* A fundamental step in the proof is to deduce that if p is a prime integer then

$$ab \in \langle p \rangle \Rightarrow a \in \langle p \rangle \text{ or } b \in \langle p \rangle.)$$

(c) Prove that if p is a prime integer then the integral domain $\mathbf{Z}_p$ of part (b) is in fact a field. (*Hint:* Use Theorem 5.6.)

5-11. Generalize Exercise 5-10 (a), (b), and (c) to the case where the Euclidean domain $\mathbf{Z}$ is replaced by the Euclidean domain $F[x]$ of univariate polynomials over a field F.

5-12. **(a)** In the integral domain $\mathbf{Q}[[x]]$ of power series over $\mathbf{Q}$, describe the ideal $\langle x^e \rangle$ where e is a fixed positive integer.

(b) Consider the natural homomorphism

$$\phi_{\langle x^e \rangle} : \mathbf{Q}[[x]] \rightarrow \mathbf{Q}[[x]]/\langle x^e \rangle.$$

Describe the elements in the homomorphic image $\mathbf{Q}[[x]]/\langle x^e \rangle$. Describe a practical representation for the elements in this homomorphic image. (cf. Chapter 3.)

5-13. For extended power series in $\mathbf{Q}\langle x \rangle$, what is the ideal $\langle x^e \rangle$ where e is a fixed integer? Does $\mathbf{Q}\langle x \rangle$ have a homomorphic image comparable to the case of ordinary power series considered in the preceding problem?

5-14. **(a)** Let $p(x) \in \mathbf{Q}[x]$ be a fixed polynomial and consider the quotient ring $\mathbf{Q}[x]/\langle p(x) \rangle$. Prove that two polynomials $a(x), b(x) \in \mathbf{Q}[x]$ lie in the same residue class in this quotient ring if and only if

$$\text{rem}(a(x), p(x)) = \text{rem}(b(x), p(x)).$$

Thus deduce a practical representation for the elements in the homomorphic image $\mathbf{Q}[x]/\langle p(x) \rangle$.

(b) Let the polynomial $p(x)$ in part (a) be the linear polynomial

$$p(x) = x - \alpha \quad \text{for some fixed } \alpha \in \mathbf{Q}.$$

Prove that the evaluation homomorphism

$$\phi_{x-\alpha} : \mathbf{Q}[x] \rightarrow \mathbf{Q}$$

as defined in section 5.1 can be defined equivalently by

$$\phi_{x-\alpha}(a(x)) = \text{rem}(a(x), x - \alpha) \quad \text{for all } a(x) \in \mathbf{Q}[x].$$

Thus deduce that the evaluation homomorphism $\phi_{x-\alpha}$ is indeed the natural homomorphism with kernel $\langle x - \alpha \rangle$ which projects $\mathbf{Q}[x]$ onto the homomorphic image $\mathbf{Q}[x]/\langle x - \alpha \rangle$.

5-15. Generalize the preceding problem to the case of a multivariate polynomial domain $D[x_1, \ldots, x_v]$ over an arbitrary UFD D, as follows.

(a) Let $p(x_i) \in D[x_i]$ be a *monic* univariate polynomial over D in the particular indeterminate x_i and consider the quotient ring $D[x_1, \ldots, x_v]/\langle p(x_i) \rangle$. Prove that two polynomials $a(x_1, \ldots, x_v)$, $b(x_1, \ldots, x_v) \in D[x_1, \ldots, x_v]$ lie in the same residue class in this quotient ring if and only if

$$\text{prem}(a(x_1, \ldots, x_v), p(x_i)) = \text{prem}(b(x_1, \ldots, x_v), p(x_i)),$$

where the prem operation is performed in the (univariate) polynomial domain $D[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_v][x_i]$. (Note that since $p(x_i)$ is assumed to be monic, the operation of pseudo-division is in fact just ordinary polynomial division.) Thus deduce a practical representation for the elements in the homomorphic image $D[x_1, \ldots, x_v]/\langle p(x_i) \rangle$.

(b) Let the polynomial $p(x_i)$ in part (a) be the linear polynomial

$$p(x_i) = x_i - \alpha \quad \text{for some fixed } \alpha \in D.$$

Prove that the evaluation homomorphism

$$\phi_{x_i-\alpha} : D[x_1, \ldots, x_v] \rightarrow D[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_v]$$

as defined in section 5.1 can be defined equivalently by

$$\phi_{x_i-\alpha}(a(x_1, \ldots, x_v)) = \text{prem}(a(x_1, \ldots, x_v), x_i - \alpha)$$

$$\text{for all } a(x_1, \ldots, x_v) \in D[x_1, \ldots, x_v].$$

Thus deduce that the evaluation homomorphism $\phi_{x_i-\alpha}$ is indeed the natural homomorphism with kernel $\langle x_i - \alpha \rangle$ which projects $D[x_1, \ldots, x_v]$ onto the homomorphic image $D[x_1, \ldots, x_v]/\langle x_i - \alpha \rangle$.

5-16. **(a)** Describe a practical representation for the elements in the quotient ring $\mathbf{Q}[x]/\langle x^2 + 1 \rangle$. (cf. Exercise 5-14(a).) Prove that this quotient ring is a field. (cf. Exercise 5-11.)

(b) In part (a) suppose that the coefficient field $\mathbf{Q}$ is changed to $\mathbf{R}$ (the real numbers). What is the field $\mathbf{R}[x]/\langle x^2 + 1 \rangle$?

(c) Is the quotient ring $\mathbf{Z}[x]/\langle x^2 + 1 \rangle$ a field? Is it an integral domain? What is the relationship between this quotient ring and the domain G of Gaussian integers defined in Exercise 2-7?

5-17. **(a)** In the congruence notation defined in section 5.3, what is the meaning of

$$a \equiv b \pmod 0?$$

(b) In the integral domain **Z**, compute the inverse of 173 modulo 945 – i.e. solve the following congruence equation for $x \in \mathbf{Z}$:

$$173\,x \equiv 1 \pmod{945}.$$

Use the method given in the proof of Theorem 5.6.

(c) In the polynomial domain $\mathbf{Q}[x]$, solve the following congruence equation for $u(x)$:

$$(x+1)(x+2)\,u(x) \equiv x \pmod{x(x-1)(x-2)}.$$

Be sure to reduce the solution modulo $x(x-1)(x-2)$.

5-18. **(a)** Suppose that the single-precision integers on a particular computer are restricted to the range $-100 < a < 100$ (i.e. two-digit integers). Determine the ten largest such single-precision integers which are odd and pairwise relatively prime. (Note that Example 5.15 uses the three largest such integers.)

(b) What range of integers can be represented in a modular representation using as moduli the ten integers determined in part (a)? In particular, how many decimal digits long must an integer be in order that it not be representable?

5-19. **(a)** Using the positive consistent representation, express the integer $u = 102156$ in mixed radix representation with respect to the moduli $m_0 = 99$, $m_1 = 97$, and $m_2 = 95$.

(b) Repeat part (a) using the symmetric consistent representation.

5-20. Apply Algorithm 5.1 by hand to solve the following Chinese remainder problems for the integer u:

(a) $u \equiv 1 \pmod 5$;
$u \equiv -3 \pmod 7$;
$u \equiv -2 \pmod 9$.

(b) $u \equiv 1 \pmod 5$;
$u \equiv -2 \pmod 7$;
$u \equiv -4 \pmod 9$.

5-21. **(a)** Step 2 of Algorithm 5.1 is based on formulas (16) - (17) with the computation of the required inverses performed in step 1. Show that an alternate method to compute the same mixed radix coefficient v_k $(0 \leqslant k \leqslant n)$ can be based on formula (16) and the following rearrangement of formula (17) for $k \geqslant 1$:

$$v_k \equiv (\cdots((u_k - v_0)m_0^{-1} - v_1)m_1^{-1} - \cdots - v_{k-1})m_{k-1}^{-1} \pmod{m_k}.$$

Note that the inverses appearing in this formula are inverses modulo m_k.

(b) If step 2 of Algorithm 5.1 were based on the alternate formula of part (a), what set of inverses would have to be computed in step 1? In particular, how many inverses are now required?

(c) Compare the computational efficiency of Algorithm 5.1 with the alternate algorithm proposed above. Consider the case where the set $\{m_i\}$ of moduli is fixed (i.e. the case where the computation of inverses in step 1 would be removed from the algorithm and pre-computed) and also consider the case where no pre-computation is possible.

5-22. (a) The proof of Theorem 5.8 outlines a method for solving the polynomial interpolation problem by solving a system of linear equations. For the specific problem described in the preamble of Algorithm 5.2 (in particular, $D = \mathbf{Z}_p[y]$ and $\alpha_i \in \mathbf{Z}_p$, $0 \leqslant i \leqslant n$) the linear system of Theorem 5.8 can be solved by an algorithm based on the familiar Gaussian elimination method and the only divisions required are divisions in the field $\mathbf{Z}_p$. In this case the solution $u(x)$ will lie in the domain $D[x]$ and not in the larger domain $F_D[x]$ of Theorem 5.8. Give an algorithmic description of such a method for solving the polynomial interpolation problem.

(b) Compare the computational cost of the algorithm of part (a) with the cost of Algorithm 5.2.

5-23. Use Algorithm 5.2 (by hand) to determine the polynomial $u(x, y, z) \in \mathbf{Z}_5[x, y, z]$ with maximum degrees 2 in x, 1 in y, and 1 in z specified by the following values in the field $\mathbf{Z}_5$

$$\begin{aligned} u(0,0,0) &= 1; & u(0,0,1) &= 2; \\ u(0,1,0) &= -1; & u(0,1,1) &= 0; \\ u(1,0,0) &= 0; & u(1,0,1) &= 2; \\ u(1,1,0) &= 2; & u(1,1,1) &= -2; \\ u(2,0,0) &= 1; & u(2,0,1) &= 2; \\ u(2,1,0) &= 0; & u(2,1,1) &= 0. \end{aligned}$$

Express the result in expanded canonical form.

5-24. Consider the problem of inverting composite modular/evaluation homomorphisms of the form

$$\phi_{x-\alpha_i}\phi_{p_i} : \mathbf{Z}[x] \rightarrow \mathbf{Z}_{p_i}.$$

Suppose that the polynomial $u(x) \in \mathbf{Z}[x]$ to be determined is known to be of degree 3 with coefficients not exceeding 17 in magnitude, and suppose that the following images of $u(x)$ have been determined.

p_i	α_i	$\phi_{x-\alpha_i}\phi_{p_i}(u(x))$
5	0	1
5	1	1
5	2	2
5	−1	0
7	0	0
7	1	−2
7	2	0
7	−1	2

(a) Verify that the image of $u(x)$ in $\mathbf{Z}_5[x]$ is

$$u_0(x) = 1 - 2x(x-1) + 2x(x-1)(x-2) \in \mathbf{Z}_5[x]$$

and that the image of $u(x)$ in $\mathbf{Z}_7[x]$ is

$$u_1(x) = -2x + 2x(x-1) + 3x(x-1)(x-2) \in \mathbf{Z}_7[x].$$

Note that these interpolating polynomials have been left in Newton (mixed radix) form.

(b) To complete the inversion process, we must solve the following Chinese remainder problem:

$$u(x) \equiv u_0(x) \pmod 5;$$

$$u(x) \equiv u_1(x) \pmod 7.$$

Suppose that this Chinese remainder problem is solved by leaving the polynomials $u_0(x)$ and $u_1(x)$ of part (a) in Newton form, yielding the result in the Newton form

$$u(x) = c_0 + c_1 x + c_2 x(x-1) + c_3 x(x-1)(x-2).$$

Calculate the values of c_0, c_1, c_2, and c_3 that would result.

(c) Determine the polynomial $u(x) \in \mathbf{Z}[x]$ by solving the Chinese remainder problem of part (b) after expressing the polynomials $u_0(x)$ and $u_1(x)$ of part (a) in expanded canonical form in their respective domains. Is the result the same as the result in part (b)? Is there any relationship between the two results?

(d) In the problem considered above, the set of evaluation points is the same for each modulus $p_0 = 5$ and $p_1 = 7$. In many practical problems the set of evaluation points will be different for different moduli. Does this affect the possibility of avoiding the conversion from Newton form to expanded canonical form as contemplated in part (b)?