

# POLYNOMIALS

James T. Smith  
San Francisco State University

For any real coefficients  $a_0, a_1, \dots, a_n$  with  $n \geq 0$  and  $a_n \neq 0$ , the function  $p$  defined by setting

$$p(x) = a_0 + a_1x + \dots + a_nx^n$$

for all real  $x$  is called a *real polynomial of degree  $n$* . Often, we write  $n = \text{degree } p$  and call  $a_n$  its *leading* coefficient. The constant function with value zero is regarded as a real polynomial of degree  $-1$ . For each  $n$  the set of all polynomials with degree  $\leq n$  is closed under addition, subtraction, and multiplication by scalars. The algebra of polynomials of degree  $\leq 0$  —the constant functions— is the same as that of real numbers, and you need not distinguish between those concepts. Polynomials of degrees 1 and 2 are called *linear* and *quadratic*.

## Polynomial multiplication

Suppose  $f$  and  $g$  are nonzero polynomials of degrees  $m$  and  $n$ :

$$\begin{aligned} f(x) &= a_0 + a_1x + \dots + a_mx^m \quad \& \quad a_m \neq 0, \\ g(x) &= b_0 + b_1x + \dots + b_nx^n \quad \& \quad b_n \neq 0. \end{aligned}$$

Their product  $fg$  is a nonzero polynomial of degree  $m + n$ :

$$f(x)g(x) = a_0b_0 + \dots + a_mb_nx^{m+n} \quad \& \quad a_mb_n \neq 0.$$

From this you can deduce the *cancellation law*: if  $f$ ,  $g$ , and  $h$  are polynomials,  $f \neq 0$ , and  $fg = fh$ , then  $g = h$ . For then you have  $0 = fg - fh = f(g - h)$ , hence  $g - h = 0$ . Note the analogy between this wording of the cancellation law and the corresponding fact about multiplication of numbers.

One of the most useful results about integer multiplication is the unique factorization theorem: for every integer  $f > 1$  there exist primes  $p_1, \dots, p_m$  and integers  $e_1, \dots, e_m > 0$  such that  $f = p_1^{e_1} \dots p_m^{e_m}$ ; the set consisting of all pairs  $\langle p_k, e_k \rangle$  is unique. (An integer is called prime if it's  $> 1$  and not a product of smaller positive integers.) A similar result holds for polynomial multiplication. A nonconstant polynomial is called *prime* if it's not a product of polynomials of smaller degree. Thus, the prime polynomials

include all linear polynomials. Here's the *unique factorization theorem for polynomials*: for every nonconstant polynomial  $f$  there exist a scalar  $a$ , prime polynomials  $p_1, \dots, p_m$  with leading coefficients 1, and integers  $e_1, \dots, e_m > 0$  such that

$$f(x) = ap_1(x)^{e_1} \cdots p_m(x)^{e_m};$$

the set consisting of  $a$  and all pairs  $\langle p_k, e_k \rangle$  is unique. The proof of the unique factorization theorem for polynomials is too tedious to give here, but it involves nothing more than the material already presented and the method of proof of the analogous theorem for integers. The cancellation law is used heavily. For details, see the reference by van der Waerden.

### Polynomial division

In elementary algebra you learned *polynomial division*: given polynomials  $f$  and  $g$  with  $g \neq 0$ , you can find *quotient* and *remainder* polynomials  $q$  and  $r$  such that

$$f = gq + r \text{ \& degree } r < \text{degree } g.$$

You usually write that equation as

$$\frac{f}{g} = q + \frac{r}{g},$$

but the first version is easier to typeset and involves only polynomials. In fact,  $q$  and  $r$  are *unique*. You can prove that as follows. Suppose you also had

$$f = gq_1 + r_1 \text{ \& degree } r_1 < \text{degree } g$$

for some polynomials  $q_1$  and  $r_1$ . Then

$$\begin{aligned} 0 &= f - f = (gq + r) - (gq_1 + r_1) = g(q - q_1) + (r - r_1) \\ r_1 - r &= g(q - q_1). \end{aligned}$$

If  $q - q_1 \neq 0$ , then by the previous paragraph,  $r_1 - r \neq 0$  and

$$\text{degree}(r_1 - r) = \text{degree } g + \text{degree}(q - q_1) \geq \text{degree } g,$$

contradiction! Thus  $q - q_1 = r_1 - r = 0$ .

This view of polynomial division underlies some of the most important facts about polynomials. One quick consequence is the *remainder theorem*: for any polynomial  $f$  and any real  $t$ , there's a unique polynomial  $q$  such that for all  $x$ ,

$$f(x) = (x - t)q(x) + f(t).$$

To prove that, determine polynomials  $q$  and  $r$  by dividing  $f(x)$  by the linear polynomial  $x - t$ :

$$f(x) = (x - t)q(x) + r(x) \quad \text{degree } r(x) < \text{degree}(x - t) = 1.$$

Thus  $r(x)$  is constant; setting  $x = t$ , you see that its value is  $f(t)$ .

### Polynomial evaluation

The naive way to evaluate

$$f(t) = a_0 + a_1t + \dots + a_nt^n$$

—computing  $t^2, \dots, t^n$  then the terms  $a_1t, \dots, a_nt^n$  then adding—requires  $n$  additions and  $2n - 1$  multiplications. Here’s a more efficient method, *Horner’s algorithm*:

$$\begin{aligned} b_{n-1} &\leftarrow a_n; \\ \text{for } k = n - 1 \text{ down to } 1 \text{ do} \\ & \quad b_{k-1} \leftarrow a_k + tb_k; \\ f(t) &\leftarrow a_0 + tb_0. \end{aligned}$$

For  $n = 3$ , this amounts to the computation

$$f(t) = a_0 + t(a_1 + t(a_2 + ta_3))$$

Horner’s method requires only  $n$  additions and  $n$  multiplications!

Horner’s algorithm is also known as *synthetic division*, for the following reason: with its intermediate results  $b_0, \dots, b_{n-1}$ , compute

$$\begin{aligned} f(t) + (x - t)(b_0 + b_1x + \dots + b_{n-1}x^{n-1}) \\ &= f(t) - tb_0 - tb_1x - \dots - tb_{n-1}x^{n-1} \\ & \quad + b_0x + \dots + b_{n-2}x^{n-1} + b_{n-1}x^n \\ &= [f(t) - tb_0] + [b_0 - tb_1]x + \dots + [b_{n-2} - tb_{n-1}]x^{n-1} \\ &= a_0 + a_1x + \dots + a_{n-1}x^{n-1} + a_nx^n \\ &= f(x). \end{aligned}$$

Thus  $b_0, \dots, b_{n-1}$  are the coefficients of the quotient polynomial  $q$  in the division  $f(x) = (x - t)q(x) + f(t)$ , and  $f(t)$  is the remainder.

Moreover,  $q(t) = f'(t)$ . *Careful!* In general,  $q(x) \neq f'(x)$ , but these polynomials do agree when  $x = t$ :

$$\begin{aligned} f'(t) &= \frac{d}{dt} [(x-t)q(x) + f(t)]_{x=t} \\ &= [q(x) + (x-t)q'(x)]_{x=t} = q'(t). \end{aligned}$$

Thus you can evaluate a polynomial  $f$  and its derivative for an argument  $t$  by two applications of Horner's algorithm: first apply it to  $f$  to get  $f(t)$  and the coefficients of  $q$ , then to  $q$  to get  $q(t) = f'(t)$ .

Polynomial evaluation is such an important feature of numerical software that much effort has been spent analyzing its efficiency. It's known that any algorithm that computes

$$f(t) = a_0 + a_1t + \dots + a_nt^n$$

for all possible values of  $a_0, \dots, a_n$  and  $t$  must use at least  $n$  additions and  $n$  multiplications, and that the only one that achieves this optimality is Horner's. (The appendix to these notes presents a partial proof.) On the other hand, if you need to compute  $f(t)$  for many values of  $t$  with a fixed array of coefficients  $a_0, \dots, a_n$ , it may be more efficient to *preprocess* the coefficients. Here's a sample preprocessing algorithm for  $n = 4$ :

$$\begin{aligned} c_0 &\leftarrow (a_3 - a_4)/(2a_4) \\ c_1 &\leftarrow (a_1/a_4) - (c_0 a_2/a_4) + c_0^2(c_0 + 1) \\ c_2 &\leftarrow (a_2/a_4) - c_0(c_0 + 1) - c_1 \\ c_3 &\leftarrow (a_0/a_4) - c_1 c_2 \\ d &\leftarrow (x + c_0)x \\ f(t) &\leftarrow a_4[(d + c_1)(d + x + c_2) + c_3]. \end{aligned}$$

After you compute  $c_0, \dots, c_3$  independently of  $t$ , this scheme requires 5 additions and 3 multiplications to compute  $f(t)$ . It's more efficient than Horner's if multiplication takes substantially longer than addition. Research to determine optimal preprocessing schemes is still under way. It's also known that repeating Horner's algorithm for computing  $f(t)$  and  $f'(t)$  together is *not* the most efficient way to compute them. For details of the results cited in this paragraph, see the references by Knuth and Kronsjö.

## Real polynomial roots

By the remainder theorem, if  $t$  is a root of a polynomial  $f$ , then  $f(x) = (x-t)q(x) + f(t) = (x-t)q(t)$  for some polynomial  $q$ , hence  $x-t$  divides  $f(x)$ . This result is often called the *factor theorem*.

You can apply the factor theorem repeatedly to prove the following result: *if a nonzero polynomial  $f$  has  $n$  distinct roots, then its degree is  $\geq n$* . Suppose the roots are  $t_1, \dots, t_n$ . Then

$$f(x) = (x - t_1)g_1(x)$$

for some polynomial  $g_1$ . Suppose you've shown that

$$f(x) = (x - t_1) \cdots (x - t_k)g_k(x)$$

for some  $k < n$  and some polynomial  $g_k$ . Then, setting  $x = t_{k+1}$  you get  $g_k(t_{k+1}) = 0$ , hence

$$g_k(x) = (x - t_{k+1})g_{k+1}(x)$$

for some polynomial  $g_{k+1}$ , and

$$f(x) = (x - t_1) \cdots (x - t_{k+1})g_{k+1}(x).$$

By recursion, you can conclude that

$$f(x) = (x - t_1) \cdots (x - t_n)g_n(x)$$

for some polynomial  $g_n$ , hence

$$\begin{aligned} \text{degree } f &= \text{degree}[(x - t_1) \cdots (x - t_n)] + \text{degree } g_n \\ &= n + \text{degree } g_n \geq n. \end{aligned}$$

You can improve this result by considering the *orders* of the roots of  $f$ . A real number  $t$  is called a *root of order  $e$*  if the polynomial  $(x - t)^e$  divides  $f(x)$  but  $(x - t)^{e+1}$  does not. Thus every root has order  $\geq 1$ . Here's the stronger result: *if a nonzero polynomial  $f$  has distinct roots  $t_1, \dots, t_n$  of orders  $e_1, \dots, e_n$  then  $\text{degree } f \geq e_1 + \dots + e_n$* . To derive this inequality, proceed as before, then recall the unique factorization theorem for polynomials:

$$f(x) = (x - t)^{e_1} g_1(x) = ap_1(x)^{e_1'} \cdots p_m(x)^{e_m'}$$

for some polynomial  $g_1(x)$ , where the rightmost term of the equation is the prime decomposition of  $f$ . Also, write the prime factorization of  $g_1(x)$ :

$$\begin{aligned} g_1(x) &= bq_1(x)^{d_1'} \cdots q_m(x)^{d_m'} \\ (x - t)^{e_1} bq_1(x)^{d_1'} \cdots q_m(x)^{d_m'} &= ap_1(x)^{e_1'} \cdots p_m(x)^{e_m'}. \end{aligned}$$

Since  $(x - t_1)^{e_1+1}$  doesn't divide  $f(x)$ ,  $x - t$  doesn't divide  $g(x)$ , hence  $x - t \neq q_k(x)$  for any  $k$ . Since  $x - t$  is prime,  $x - t = p_k(x)$  and  $e_1 = e_k'$  for some  $k$  by the unique factorization theorem. Now apply this argument to the other roots  $t_2, \dots, t_n$  in turn. Each factor  $x - t_j$  is equal to a distinct prime  $p_k(x)$ , and  $e_j = e_k'$ . Thus

$$f(x) = (x - t_1)^{e_1} \cdots (x - t_n)^{e_n} h(x)$$

for some polynomial  $h(x)$ , hence

$$\text{degree } f = e_1 + \cdots + e_n + \text{degree } h \geq e_1 + \cdots + e_n.$$

According to Taylor's theorem, if  $f$  is an  $n$ th degree polynomial, then

$$f(x) = \sum_{k=0}^n \frac{f^{(k)}(t)}{k!} (x - t)^k$$

for any  $x$  and  $t$ . It follows that  $t$  is a root of  $f$  of order  $e$  just in case  $f^{(k)}(t) = 0$  for each  $k < e$  but  $f^{(e)}(t) \neq 0$ .

So far, this discussion has only yielded limits on the number and degree of the roots of a polynomial. Showing *existence* of a root is another matter. For an *odd*-degree polynomial, that requires a tedious argument with inequalities.

First, note that for any real  $a$  and  $b$  with  $a > 0$ , and any integers  $m$  and  $n$  with  $m > n \geq 0$ , there exists  $t \geq 0$  such that  $ax^m \geq bx^n$  for all  $x \geq t$ . Just take  $t = 0$  if  $b \leq 0$  and  $t = (b/a)^{1/(m-n)}$  otherwise.

Now let  $m$  be an odd integer,  $a_m \neq 0$ , and consider the polynomial

$$f(x) = a_0 + a_1x + \cdots + a_mx^m.$$

To find a root of  $f$ , first apply the result of the previous paragraph to find  $t_0, \dots, t_{m-1} \geq 0$  such that

$$\begin{aligned} x \geq t_0 &\Rightarrow (a_m/m)x^m \geq -a_0x^0 \\ &\vdots \\ x \geq t_{m-1} &\Rightarrow (a_m/m)x^m \geq -a_{m-1}x^{m-1}. \end{aligned}$$

Let  $t = \max(t_0, \dots, t_{m-1})$  and suppose  $x \geq t$ . Then all these inequalities hold. Adding them, you get

$$\begin{aligned} a_mx^m &\geq -a_{m-1}x^{m-1} - \cdots - a_0 \\ f(x) &= a_mx^m + a_{m-1}x^{m-1} + \cdots + a_0 \geq 0. \end{aligned}$$

Thus  $f(x) \geq 0$  when  $x \geq t$ . Next, using the same method, find  $u_0, \dots, u_{m-1} \geq 0$  such that

$$\begin{aligned}
x \geq u_0 &\Rightarrow (a_m/m)x^m \geq +a_0x^0 \\
x \geq u_1 &\Rightarrow (a_m/m)x^m \geq -a_1x^1 \\
&\vdots \\
x \geq u_{m-2} &\Rightarrow (a_m/m)x^m \geq -a_{m-2}x^{m-2} \\
x \geq u_{m-1} &\Rightarrow (a_m/m)x^m \geq +a_{m-1}x^{m-1}.
\end{aligned}
\tag{alternate  $\pm$ }$$

Let  $u = \max(u_0, \dots, u_{m-1})$  and suppose  $x \geq u$ . Then all these inequalities hold. Adding them, you get

$$\begin{aligned}
a_mx^m &\geq a_{m-1}x^{m-1} - a_{m-2}x^{m-2} - \dots - a_1x + a_0 \\
f(-x) &= -a_mx^m + a_{m-1}x^{m-1} - a_{m-2}x^{m-2} - \dots - a_1x + a_0 \leq 0.
\end{aligned}$$

Thus,  $f(-x) \leq 0$  when  $x \geq u$ , hence  $f(-u) \leq 0 \leq f(t)$ . Existence of a root of  $f$  between  $-u$  and  $t$  follows from the intermediate value theorem.

*Even-degree* polynomials need have no roots: for example,  $x^2 + 1$ .

If you must look for a root, you'd like to know beforehand that you can confine your search to a certain interval. *Cauchy's bound* provides that information:

$$0 = a_0 + a_1t + \dots + a_nt^n \text{ implies } |t| \leq 1 + \max_{k < n} \left| \frac{a_k}{a_n} \right|.$$

This inequality is trivial if  $|t| \leq 1$ . To prove it for  $|t| > 1$ , let  $M$  denote that maximum value, apply the triangle inequality, some algebra, and a familiar result about finite geometric series:

$$\begin{aligned}
|a_n||t|^n &= |-a_nt^n| = |a_0 + a_1t + \dots + a_{n-1}t^{n-1}| \\
&\leq |a_0| + |a_1||t| + \dots + |a_{n-1}||t|^{n-1} \\
|t^n| &= \left| \frac{a_0}{a_n} \right| + \left| \frac{a_1}{a_n} \right| |t| + \dots + \left| \frac{a_{n-1}}{a_n} \right| |t|^{n-1} \\
&\leq M + M|t| + \dots + M|t|^{n-1} \\
&= M(1 + |t| + \dots + |t|^{n-1}) \\
&= M \frac{|t|^n - 1}{|t| - 1}.
\end{aligned}$$

Since  $|t| > 1$ ,

$$\begin{aligned}
|t|^n(|t| - 1) &\leq M(|t|^n - 1) \\
|t| - 1 &\leq M(1 - |t|^{-n}) < M.
\end{aligned}$$

That's the desired result.

## Complex polynomial roots

You saw earlier that every odd-degree polynomial has at least one real root, but *even* degree polynomials needn't have any. Gauss' *fundamental theorem of algebra* states that *every* non-constant polynomial  $f$  has a *complex* root—i.e. there exist real numbers  $x$  and  $y$  such that  $f(x + iy) = 0$ , where  $i^2 = -1$ . The most common proof of this famous result involves complex integral calculus, and cannot be given here. Consult the reference by Hille. The following proof was described recently in the reference by Remmert. He attributes it to R. Argand, 1814.

By Cauchy's bound, all roots of a polynomial  $p(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$  lie in the closed disk  $D$  centered at 0 with radius

$$1 + \max_{k < n} \left| \frac{a_k}{a_n} \right|.$$

Since polynomials and the absolute value function are continuous, and any composition of continuous functions is continuous, the real-valued function  $q : z \rightarrow |p(z)|$  is continuous on  $D$ , hence it assumes a minimum value. That is, there exists  $c \in D$  such that for all  $w \in D$ ,  $q(c) \leq q(w)$ . (These properties of continuous functions weren't stated and proved until about fifty years after Argand's work; Remmert says Argand would have thought them obvious, not needing proof.)

Argand shows  $p(c) = 0$  by assuming the opposite and deriving a contradiction. Under that assumption the function  $h : z \rightarrow p(c)^{-1}p(z + c)$  is a nonconstant polynomial. That is, there exist integers  $k$  and  $n$  such that  $1 \leq k \leq n$ , and complex numbers  $b_k, \dots, b_n$  such that  $b_k \neq 0$  and  $h(z) = 1 + b_kz^k + \dots + b_nz^n$  for all complex  $z$ . Find  $d$  such that  $d^k = -1/b_k$ . The polynomial  $g : z \rightarrow b_{k+1}d^{k+1}z + \dots + b_nd^n z^{n-k}$  is continuous and  $g(0) = 0$ , so there exists  $\delta > 0$  such that  $|g(t)| < 1/2$  for all  $t$  such that  $0 \leq t < \delta$ . Select any  $t$  such that  $0 < t < 1, \delta$  and let  $w = dt + c$ . Then

$$\begin{aligned} |h(dt)| &= |1 + b_k(dt)^k + \dots + b_n(dt)^n| \\ &= |1 + b_k d^k t^k + b_{k+1} d^{k+1} t^{k+1} + \dots + b_n d^n t^n| \\ &= |1 - t^k + t^k g(t)| \leq |1 - t^k| + |t^k g(t)| \\ &= 1 - t^k + t^k |g(t)| < 1 - t^k + 1/2 t^k < 1 - 1/2 t^k < 1. \\ q(w) &= |p(dt + c)| = |p(c)h(dt)| = |p(c)| |h(dt)| \\ &< |p(c)| = q(c). \end{aligned}$$

This contradicts the previous paragraph, which established  $q(c) \leq q(w)$ . And that establishes the theorem:  $p$  does have the complex root  $c$ .

Complex roots provide useful information about real polynomial factorization. First, note that if  $z = x + iy$  is a complex root of a real polynomial  $f$ , then its *conjugate*  $z^* = x - iy$  is also a root. To see that, observe first that conjugation preserves sums and products:

$$\begin{aligned} [(u + iv) + (x + iy)]^* &= [(u + x) + i(v + y)]^* = (u + x) - i(v + y) \\ &= (u - iv) + (x - iy) = (u + iv)^* + (x + iy)^* \\ [(u + iv)(x + iy)]^* &= [(ux - vy) + i(uy + vx)]^* \\ &= (ux - vy) - i(uy + vx) = (ux - vy) + i(-uy - vx) \\ &= (u - iv)(x - iy) = (u + iv)^*(x + iy)^*. \end{aligned}$$

Now suppose  $z = x + iy$  and

$$0 = f(z) = a_0 + a_1z + \dots + a_mz^m.$$

Since conjugation doesn't change *real* numbers,

$$\begin{aligned} 0 &= f(z)^* = (a_0)^* + (a_1z)^* + \dots + (a_mz^m)^* \\ &= a_0^* + a_1^*z^* + \dots + a_m^*(z^m)^* \\ &= a_0^* + a_1^*z^* + \dots + a_m^*(z^*)^m \\ &= a_0 + a_1z^* + \dots + a_m(z^*)^m \\ &= f(z^*), \end{aligned}$$

hence  $z^*$  is also a root, as claimed.

Before proceeding further, it's necessary to check the validity of the factor theorem for polynomials with complex coefficients. Namely, a complex number  $t$  is a root of a complex polynomial  $f$  just when  $x - t$  divides  $f$ . You'll find that all the algebra leading to that result applies to complex numbers as well as reals.

Now you can deduce that *every real prime polynomial  $p$  is linear or quadratic*. Argue first the case when  $p$  has a real root  $t$ . Then  $x - t$  divides  $p(x)$ , hence  $p(x) = (x - t)q(x)$  for some polynomial  $q$ . Because  $p$  is prime,  $x - t$  or  $q(x)$  must have the same degree as  $p$ , hence  $q$  must be constant and  $p$  linear. Next, suppose that  $p$  has no real root. By the fundamental theorem of algebra, it has a complex root  $t = u + iv$ . By the factor theorem,  $p(x) = (x - t)q(x)$  for some complex polynomial  $q(x)$ . Setting  $x = t^*$ , you get

$$0 = p(t^*) = (t^* - t)q(t^*) = -2ivq(t^*).$$

Since  $v \neq 0$ ,  $t^*$  is a root of  $q$ , hence  $q(x) = (x - t^*)r(x)$  for some complex polynomial  $r(x)$ . Now compute

$$\begin{aligned} t + t^* &= 2u & tt^* &= (u + iv)(u - iv) = u^2 - i^2v^2 = u^2 + v^2 \\ p(x) &= (x - t)(x - t^*)r(x) = (x - t)(x - t^*)r(x) \\ &= [x^2 - (t + t^*)x + tt^*]r(x) \\ &= [x^2 - 2ux + u^2 + v^2]r(x). \end{aligned}$$

Since all the coefficients of  $p$  are real, those of  $r$  must be real also. Because  $p$  is prime,  $x^2 - 2ux + (u^2 + v^2)$  or  $r(x)$  must have the same degree as  $p$ , hence  $r$  must be constant and  $p$  quadratic.

The previous paragraph, together with the unique factorization theorem for polynomials, implies that *every real polynomial is the product of real linear and quadratic polynomials*.

Since every complex polynomial has a complex root, *the only complex prime polynomials are the linear ones, and every complex polynomial is the product of complex linear polynomials*.

The concept of *order* applies as well to complex roots of complex polynomials. If you modify the algebra at the beginning of the section on real polynomial roots to include the complex case, and apply the fundamental theorem of algebra when necessary, you can show that *the degree of any complex polynomial is the sum of the orders of its roots*.

## References

Einar HILLE, *Analytic function theory*, Volume 1. Blaisdell, 1959.

Donald E. KNUTH, *The art of computer programming*, Volume 2: *Seminumerical algorithms*. Addison-Wesley, 1969.

Lydia I. KRONSJÖ, *Algorithms: Their complexity and efficiency*. Wiley, 1979.

Alexander M. OSTROWSKI, *Solutions of equations and systems of equations*. Academic Press, 1966.

Reinhold REMMERT, Vom Fundamentalsatz der Algebra zum Satz von Gelfand-Mazur, *Mathematische Semesterberichte* 40(1993):63–71.

B.L. van der WAERDEN, *Modern algebra, in part a development from lectures by E. Artin and E. Noether*, translated from the second revised German edition by F. Blum, revised English edition, Volume 1. Ungar, 1953.

**Exercises**

1. Use polynomial division to compute  $q$  and  $r$  with  $f = gq + r$  and degree  $r <$  degree  $g$ , for these cases:
  - a.  $f(x) = x^5 - 1$ ,  $g(x) = x - 1$ ;
  - b.  $f(x) = x^5 + 1$ ,  $g(x) = x - 1$ ;
  - c.  $f(x) = x^5 + 1$ ,  $g(x) = x^2 + 1$ .
  
2. In some common computer language, write a function Horner that uses Horner's algorithm to compute the value  $y$  of the  $N$ th degree polynomial  $P(x)$  at  $x = T$  and the coefficients of the  $N - 1$ st degree polynomial  $Q(x)$ , where  $P(x) = (x - T)Q(x) + y$ . Its input arguments should be  $N$ ,  $P$ , and  $T$ ; it should output  $Q$  and return  $y$  as function value. Represent  $P$  and  $Q$  by the arrays  $\langle P_0, \dots, P_N \rangle$  and  $\langle Q_0, \dots, Q_N \rangle$  of their coefficients. Take care with the cases  $N \leq 0$ . Test the function by using it to generate a table of values of  $x$ ,  $P(x)$ ,  $P'(x)$  for  $x = -1.0, -0.9, \dots, 1.0$ , where  $P(x) = 2.5x^3 - 1.5x$ .
  
3. If you use Horner's algorithm to divide  $f(x)$  by  $x - t$  to get the coefficients of the quotient polynomial  $q(x)$ , then  $q(t) = f'(t)$ . Now use the same algorithm to divide  $q(x)$  by  $x - t$ , obtaining coefficients of another quotient polynomial  $r(x)$ . How is  $r(t)$  related to  $f''(t)$ ? Generalize to third and fourth derivatives, etc.
  
4. Factor  $t^6 + t^5 + t^4 - t^2 - t - 1$  into real linear and quadratic factors.

For the remaining problems, consult the references by Knuth and Kronsjö.

6. Find an algorithm for evaluating a polynomial and its derivative that's more efficient than applying Horner's twice.
  
7. Verify the preprocessing algorithm described in these notes, determine how it was discovered, and investigate similar results for polynomials of other degrees.
  
8. How does the accuracy of the results  $b_0, \dots, b_{n-1}$  and  $f(t)$  of Horner's algorithm depend on that of its inputs  $a_0, \dots, a_n$  and  $t$ ?

### Appendix: Proof that Horner's algorithm is optimal

In order to prove the result, mentioned earlier, that no algorithm for evaluating polynomials requires fewer additions or multiplications than Horner's, you need a fairly precise notion of algorithm. The following level of detail seems sufficient. In this context, an algorithm accepts some input parameters, then performs steps of the form

$$v \leftarrow p \qquad v \leftarrow p + q \qquad v \leftarrow p \times q$$

where  $v$  is a variable and  $p$  and  $q$  are operands. An operand is a variable, a constant, or a parameter. The result of the last step is the value computed by the algorithm. You can depict an algorithm as a tree. For example, if  $a$ ,  $b$ , and  $c$  are parameters and  $u$  and  $v$  are variables, then you can arrange the computation  $v \leftarrow b^2 - 4ac$  like this:

$$\begin{array}{ccc}
 & a & c \\
 & \backslash & / \\
 & v \leftarrow a \times c & \\
 & | & \\
 b & & \\
 | & & \\
 u \leftarrow b \times b & & v \leftarrow 4 \times v \\
 \backslash & & / \\
 & v \leftarrow u - v & 
 \end{array}$$

The leaves are labeled by parameters. A step is said to *involve* a parameter if it lies below a leaf with that label.

The algorithms considered here accept parameters  $a_0, \dots, a_n$  and  $x$  and compute the value  $y = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ . Horner's algorithm uses  $n$  additions and  $n$  multiplications. The proof shows that no such algorithm can use fewer.

Before starting the proof, it's convenient to discuss some very simple facts required there. First, note that any algorithm of this type must include an addition or multiplication involving  $a^n$ : otherwise,  $y$  would be independent of that parameter.

The second preliminary fact is that any such algorithm must include a multiplication involving  $a_n$ . If it didn't, there'd be at least one addition involving  $a_n$ . Consider the first. There could be no subsequent multiplications. Thus, there'd exist a positive integer  $k$  and polynomial  $p$  such that for all  $a_0, \dots, a_n$  and  $x$ ,

$$y = ka_n + p(a_0, \dots, a_{n-1}, x).$$

Replace  $y$  by its definition to get

$$\begin{aligned}
 a_0 + \dots + a_nx^n &= ka_n + p(a_0, \dots, a_{n-1}, x) \\
 a_0 + \dots + a_{n-1}x^{n-1} - p(a_0, \dots, a_{n-1}, x) &= ka_n - a_nx^n
 \end{aligned}$$

for all  $a_0, \dots, a_n$  and  $x$ . This is impossible, because the left hand side is independent of  $a_n$ , but given any  $x$ , you can choose values of  $a_n$  that change the right hand side.

The final preliminary fact is that any such algorithm must also include an addition involving  $a_n$ . If it didn't, there'd be at least one multiplication involving  $a_n$ . Consider the first. There could be no subsequent additions. Thus, there'd exist a positive integer  $k$  and a polynomial  $p$  such that for all  $a_0, \dots, a_n$  and  $x$ ,

$$y = a_n^k p(a_0, \dots, a_{n-1}, x).$$

Replace  $y$  by its definition to get

$$\begin{aligned} a_0 + \dots + a_n x^n &= a_n^k p(a_0, \dots, a_{n-1}, x) \\ a_0 + \dots + a_{n-1} x^{n-1} &= a_n^k p(a_0, \dots, a_{n-1}, x) - a_n x^n. \end{aligned}$$

for all  $a_0, \dots, a_n$  and  $x$ . This is impossible, because you can choose values of  $a_0, \dots, a_{n-1}$  and  $x$  so that the left hand side is not zero, then set  $a_n = 0$ .

You can now prove the optimality result by recursion on  $n$ . It's true trivially for  $n = 0$ . Suppose it holds for a certain  $n$ . Let  $A$  be an algorithm that computes  $a_0 + \dots + a_{n+1} x^{n+1}$ . It must include an addition involving  $a_{n+1}$ ; consider the first. That must have the form  $v \leftarrow a_{n+1} + w$ . Construct a new algorithm  $A'$  by changing this step to  $v \leftarrow w$ , then replacing all remaining  $a_{n+1}$  references by references to 0. Since  $A'$  computes  $a_0 + \dots + a_n x^n$ , it must include at least  $n$  additions, hence  $A$  must have included at least  $n + 1$ . Now consider multiplications.  $A$  must include at least one involving  $a_{n+1}$ ; consider the first. That must have the form  $v \leftarrow a_{n+1} \times w$ . Construct a new algorithm  $A''$  by changing this step to  $v \leftarrow 0$ , then replacing all remaining  $a_{n+1}$  references by references to 0. Since  $A''$  computes  $a_0 + \dots + a_n x^n$ , it must include at least  $n$  multiplications, hence  $A$  must have included at least  $n + 1$ .

Thus Horner's algorithm is optimal, in the sense that none of the same type is faster. It's also known that any optimal algorithm performs essentially *the same*  $n$  additions and multiplications as Horner's.