We would like to generalize the division algorithm to polynomials in several variables. In the single variable case, it suffices to be able to divide by a single polynomial

\[ g(x) = q(x)f(x) + r(x) \]

with \( \deg r(x) < \deg f(x) \), in the sense that even if we want to determine, given \( g(x), f_1(x), f_2(x) \), whether it’s possible to write

\[ g(x) = q_1(x)f_1(x) + q_2(x)f_2(x) \]

we can use Euclid’s algorithm to find the gcd \( f(x) \) of \( f_1(x) \) and \( f_2(x) \) and write it as \( f(x) = a_1(x)f_1(x) + a_2(x)f_2(x) \), and then simply divide \( g(x) \) by \( f(x) \).

Another way of saying this is that every ideal \( \langle f_1(x), \ldots, f_k(x) \rangle \subseteq R[x] \) is in fact generated by a single element \( \langle f_1(x), \ldots, f_k(x) \rangle = \langle f(x) \rangle \), and both computing this element (using Euclid’s algorithm) and testing whether \( g(x) \) is divisible by it can be carried out by dividing by single polynomials.

The situation is much more complicated when dealing with polynomials in several variables. For example, the ideal \( \langle x, y \rangle \subseteq R[x, y] \) can not be generated by a single element, and while \( x \) and \( y \) have no common factors, it is not possible to write \( 1 = q_1(x, y)x + q_2(x, y)y \). As such, we won’t be able to limit ourselves to dividing by a single polynomial.

Another condition that we’ll have to change in the multivariable case is our requirement that \( \deg r(x) < \deg f(x) \). For example, if we tried to write

\[ x^2y + xy^2 + z^3 = q_1(x, y)x + q_2(x, y)y + r(x, y), \]

it seems like whatever choices of \( q_1 \) and \( q_2 \) we make we’ll always be stuck with a \( z^3 \) term in the remainder, which has a larger degree than the polynomials \( x \) and \( y \) that we’re dividing by.

**The division algorithm**

We now describe what we can do over \( R[x_1, \ldots, x_n] \) by essentially just following the usual division algorithm for single variable polynomials. Since the single variable division algorithm involves the leading terms of various polynomials, we fix a monomial order on \( R[x_1, \ldots, x_n] \) and use it to define \( LT(f) \) to be the leading term of a polynomial \( f \) according to that monomial order.

Now suppose we are given a polynomial \( g \) that we are trying to divide by polynomials \( f_1, \ldots, f_k \in R[x_1, \ldots, x_n] \) to write

\[ g = q_1f_1 + \cdots + q_nf_n + r. \]

If \( LT(g) \) is divisible by \( LT(f_i) \), then we can add \( \frac{LT(g)}{LT(f_i)}f_i \) to \( q_i \) and subtract \( \frac{LT(g)}{LT(f_i)}f_i \) from \( g \) to cancel out the leading term of \( g \), leaving it with a a smaller leading term. In this way, we eventually arrive at a polynomial whose leading term is not divisible by any of the \( LT(f_i) \), and so we must move that leading term to the remainder. We continue in this way: we either cancel out \( LT(g) \) if it is divisible by some \( LT(f_i) \) or we just move it to the remainder if it isn’t. Since the leading term decreases at each step, this process must terminate eventually with no terms left of \( g \), and at that point every term of the remainder we’re left with will not be divisible by any of the \( LT(f_i) \).
Theorem (Division algorithm in \( \mathbb{R}[x_1, \ldots, x_n] \)). Fix a monomial order. Let \( f_1, \ldots, f_k \in \mathbb{R}[x_1, \ldots, x_n] \) be given. Then every \( g \in \mathbb{R}[x_1, \ldots, x_n] \) can be expressed as

\[
g = q_1 f_1 + \cdots + q_k f_k + r
\]

with \( q_1, \ldots, q_k, r \in \mathbb{R}[x_1, \ldots, x_n] \) and either \( r = 0 \) or every term of \( r \) is not divisible by the leading term of any of the \( f_i \). Also, the leading monomial of each \( q_i f_i \) is no greater than that of \( f \).

The division algorithm is useful, but it doesn’t give us everything that we want by itself. For example, the polynomial \( y^2 - y^3 \) is in the ideal \( \langle x - y^2, x - y^3 \rangle \), but if we try to divide \( y^2 - y^3 \) by \( x - y^2 \) and \( x - y^3 \) in lex order, nothing happens: we get \( q_1 = q_2 = 0 \) and \( r = y^2 - y^3 \), so the division algorithm is not telling us that it is in the ideal.

For now, we’ll solve this problem by defining it away. For an ideal \( I \subseteq \mathbb{R}[x_1, \ldots, x_n] \), we say that \( f_1, \ldots, f_k \in I \) are a Gröbner basis for \( I \) if

\[
LT(I) = \langle LT(f_1), \ldots, LT(f_k) \rangle
\]

where \( \langle LT(I) \rangle = \langle LT(f) : f \in I \rangle \) is the monomial ideal generated by the leading terms of all the polynomials in \( I \).

In our example above, \( \langle x - y^2, x - y^3 \rangle \) is not a Gröbner basis for \( I = \langle x - y^2, x - y^3 \rangle \) since \( y^2 = LT(y^2 - y^3) \) is in \( LT(I) \) but not in \( \langle LT(x - y^2), LT(x - y^3) \rangle \). For a Gröbner basis though, the division algorithm does what we want it to.

Proposition. Fix a monomial order. Suppose \( f_1, \ldots, f_k \in \mathbb{R}[x_1, \ldots, x_n] \) are a Gröbner basis for the ideal \( I = \langle f_1, \ldots, f_n \rangle \) that they generate and suppose \( g \in \mathbb{R}[x_1, \ldots, x_n] \). Let \( r \) be the remainder upon division of \( g \) by \( (f_1, \ldots, f_k) \). Then

1. \( g = q_1 f_1 + \cdots + q_k f_k \) has a solution \( (q_1, \ldots, q_k) \in \mathbb{R}[x_1, \ldots, x_n]^k \) if and only if \( r = 0 \), and
2. \( r \) is unique in the sense that if \( g = f + \tilde{r} \) with \( f \in I \) and no term of \( \tilde{r} \) is divisible by any \( LT(f_i) \), then \( r = \tilde{r} \).

In particular, changing the order of the \( f_i \) does not change \( r \).

Proof. The first statement is the special case of the second when \( r = 0 \). To prove the second, we just note that \( r - \tilde{r} \in I \), so that if \( r \neq \tilde{r} \), then \( LT(r - \tilde{r}) \in \langle LT(I) \rangle \). But every term of \( r - \tilde{r} \), and in particular the leading term, is not divisible by any of the \( LT(f_i) \), so that

\[
LT(r - \tilde{r}) \notin \langle LT(f_1), \ldots, LT(f_k) \rangle,
\]

which contradicts our assumption that \( f_1, \ldots, f_k \) is a Gröbner basis. \( \square \)

Thus we now “know” how to determine whether a given polynomial \( g \) is in the ideal \( I \): we find a Gröbner basis for \( I \) and then just use the division algorithm. At this point though, we’ve never even shown that any particular set of polynomials is a Gröbner basis (you’re asked to show this in a very simple example on the homework). What we really want to be able to do is start with an arbitrary (finite) set of generators for an ideal and find a Gröbner basis for it.\(^\text{1}\)

\( ^\text{1}\)A Gröbner basis will always exist because the Hilbert basis theorem tells us that \( \langle LT(I) \rangle \) is generated by finitely many elements \( LT(f_i) \). This doesn’t tell us anything about how to find one though.