Operations on ideals

Deadline: Monday 27 March, 4.00pm.

This final set of exercises is in two contrasting parts. The first part is essentially the degree exam for April 2015, the aim is to give you timetabled revision and feedback. The only difference is that Q1 has been altered to take account of the switch from Maple to Axiom; you can thus take this modified version as a model for exams to come. The second part consists of two example constructions of ideals together with results about them and how they relate to varieties. You are then asked to prove some corresponding results about a third construction and use one of these in a little practical work with Axiom.

Naturally the exam questions have a certain amount of bookwork. In answering these do not just copy out large chunks of the course notes (if you do you will lose credit). Just give a straightforward answer in your own words, keep it simple and direct. (I have left the questions intact since the aim is to give you practice in answering genuine exam questions.) Note that in the exam you have a choice of two out of three questions. This applies to this exercise. Just as for the exam, if you attempt all three questions I will mark your three attempts and give you credit for the best two. In the actual exam it is a very bad idea to attempt more than two questions due to time constraints. However for the purposes of this exercise you might wish to attempt all three if you have time.

The marking will be carried out as follows: the exam part will be marked as normal (each question out of 25) so that your maximum mark here is 50. This will then be halved to give a score out of 25. (I will also give the mark out of 25 for each question you attempt as part of feedback.) The other part has a maximum score of 25 and each sub-part will be marked out of the indicated sub-total (see below). The two separate marks will then be added together and scaled to be out of 40 (the mark being rounded to the nearest integer). This ensures that the total for the three sets of exercises of the course is out of 100 (recall that the first was out of 20 while the other two are out of 40).

Submission: Submit your handwritten answers, stapled at the top left corner, to all the exercises to the ITO. There is no electronic submission for these exercises.

Good Scholarly Practice: Please remember the University requirement as regards all assessed work for credit. Details and advice about this can be found at:

http://web.inf.ed.ac.uk/infweb/admin/policies/academic-misconduct

and links from there. Note that, in particular, you are required to take reasonable measures to protect your assessed work from unauthorised access. For example, if you put any such work on a public repository then you must set access permissions appropriately (generally permitting access only to yourself, or your group in the case of group practicals).

§1. The April 2015 Exam. Q1 and Q3c(ii) have been altered. Instructions to candidates: Answer any two questions. All questions carry equal weight.

1. (a) Explain what is meant by a canonical representation and a normal representation of objects. [4 marks]

(b) Consider members of $\mathbb{Q}(x)$ where $x$ is an indeterminate over $\mathbb{Q}$. Explain how to represent these in a canonical form (you are not required to prove this is the case) and discuss a disadvantage of doing so (give an example to illustrate the disadvantage). [4 marks]

(c) Consider two polynomials $f, g$ in one indeterminate $x$ and with integer coefficients. We want to find isolating intervals for the common real roots of $f, g$. The following Axiom code aims to do this.
commonRoots(f:UP(x,INT),g:UP(x,INT)):List(Record(left:FRAC INT,right:FRAC INT))==
if f=0 and g=0 then
  error "Both arguments are 0!"
else if f=0 then
  realZeros(g)
else if g=0 then
  realZeros(f)
else
  F:=realZeros(f)
  R:=[]
  for r in F repeat
    if hasRoot(g,r) then R:=cons(r,R)
  R

Note the following:
• The function realZeros is built into Axiom and returns isolating intervals for the
  real roots of its argument. It returns a list of records of the type shown, a record
  [left=a, right=b] represents the open interval (a,b).
• The procedure hasRoot returns true if the first argument (a polynomial in one
  indeterminate) has a real root in r (which represents an isolating interval using
  the convention stated above). Otherwise it returns false. The code for this is not
  shown as only the functionality is relevant.

i. State an alternative declaration of the types for the function commonRoots that
leaves the header of the function simply as commonRoots(f,g)==. [2 marks]

ii. It is suggested that for an interval (a, b) with a < b we can implement the call to
hasRoot(g,[left=a,right=b]) by returning true if the sign of g(a) is different
from that of g(b); false otherwise. Is this correct? Justify your answer with either
a brief argument or a counterexample. Assume for this part that neither a nor b is
a root of g. [3 marks]

iii. In fact we want the code to return intervals that not only contain all the common
roots of f and g but also serve as isolating intervals with respect to each polynomial.
Thus each interval should contain exactly one root of f and exactly one root of g
(the same root of course). Explain why the code, as presented, does not do this
and how you would fix the problem. Your solution must retain the overall approach
of finding isolating intervals for the roots of f (i.e., any changes to the code must
apply to the body of the loop). You may use any standard algorithms provided
you mention them by name and ensure that any relevant conditions are satisfied. [8 marks]

iv. The approach taken by the supplied code is naive and inefficient. Explain how to
find the necessary intervals by replacing f and g with a single polynomial and then
finding isolating intervals for its roots. Prove the correctness of your approach. [4 marks]

2. Throughout this question k is a field, x an indeterminate over k and f, g ∈ k[x] not both 0.

   (a) Give the general definition of a gcd of two elements in a unique factorisation domain
   (UFD). State how any two gcd’s are related (no proof is required). Use the definition
to prove that gcd(f,g) is any polynomial of highest possible degree that divides both
f and g. Explain how to make the gcd unique. [6 marks]

   (b) State the Euclidean Algorithm for finding gcd(f,g), i.e., explain how to find a sequence
of remainders r_0, ..., r_n (where r_0 = f, r_1 = g) such that gcd(f,g) = r_n.
Comment briefly on the practicality of the algorithm for the cases when k is Q and
when k is Z_p for a prime number p that fits into a computer word. [5 marks]

   (c) Let q_1, ..., q_n be the sequence of quotients obtained by the Euclidean algorithm with
inputs f, g (the remainders being as above). Explain how to amend the algorithm
so that it computes two sequences \( s_0, \ldots, s_n \) and \( t_0, \ldots, t_n \) of polynomials with the property that \( r_i = s_i f + t_i g \) for \( 0 \leq i \leq n \). \[4 \text{ marks}\]

(d) From now on we assume that \( k = \mathbb{Q} \), \( g \) is an irreducible polynomial and \( g \) does not divide \( f \). Recall that this means that there is a polynomial \( h \) such that

\[
fh = 1 \mod g
\]

holds in \( \mathbb{Q}[x] \).

i. Explain how to find polynomials \( F, G \in \mathbb{Z}[x] \) such that equation \( (\dagger) \) is equivalent to requiring the existence of polynomials \( H, Q \in \mathbb{Z}[x] \) and a non-zero integer \( a \) such that

\[
FH = a + GQ.
\]

holds in \( \mathbb{Z}[x] \).

ii. For a prime number \( p \) and a polynomial \( A \in \mathbb{Z}[x] \) let \( A_p \) be the polynomial in \( \mathbb{Z}_p[x] \) obtained by taking the coefficients of \( A \) modulo \( p \). Assume that \( p \) does not divide \( \gcd(\text{lcm}(F), \text{lcm}(G)) \). Prove that there exist \( H, Q \) and \( a \) with \( p \) not dividing \( a \) such that equation \( (\dagger) \) holds if and only if \( \gcd(F_p, G_p) = 1 \), where the \( \gcd \) is computed in \( \mathbb{Z}_p[x] \). \[5 \text{ marks}\]

3. (a) State Hilbert’s Nullstellensatz and explain its significance for systems of polynomial equations over the complex numbers.

Give a counterexample to show that the Nullstellensatz does not hold for systems over the real numbers. \[7 \text{ marks}\]

(b) Let \( x_1, x_2, \ldots, x_n, z \) be indeterminates over the complex numbers \( \mathbb{C} \) and \( p, p_1, \ldots, p_m \in \mathbb{C}[x_1, \ldots, x_n] \). 

i. Use Hilbert’s Nullstellensatz to show that \( p \) vanishes whenever \( p_1, \ldots, p_m \) do if and only if

\[
1 \in (p_1, \ldots, p_m, pz - 1),
\]

where \((p_1, \ldots, p_m, pz - 1)\) denotes the ideal generated by the given polynomials. \[6 \text{ marks}\]

ii. Describe an algorithm for deciding if \( V(p, p_1, \ldots, p_m) = V(p_1, \ldots, p_m) \). \[3 \text{ marks}\]

(c) Now let \( k \) be a field and \( X \) a finite non-empty set of \( n \) indeterminates over \( k \). A 2-nomial is a polynomial from \( k[X] \) of the form \( a_1u_1 + a_2u_2 \) where \( a_1, a_2 \in k \) and \( u_1, u_2 \) are power products. We assume that an admissible order < on the power products has been fixed and \( F \) is a non-empty finite set of 2-nomials none of which is 0. Let \( G \) be the Gröbner basis of the ideal \((F)\) obtained by running the Buchberger algorithm with input \( F \).

i. Prove that \( G \) consists of 2-nomials.

Does every Gröbner basis of \((F)\) consist of 2-nomials? Justify your answer briefly. \[5 \text{ marks}\]

ii. Assume now that the admissible order is a degree then lexicographic one and suppose that \( d = \max\{ \deg f \mid f \in F \} \). It is conjectured that for all \( g \in G \) we have \( \deg g \leq d \). Prove that this is false by providing a counter example. \[4 \text{ marks}\]

§2. Operations on Ideals: the Algebra–Geometry Dictionary. Let \( k \) be a field and \( X = \{x_1, x_2, \ldots, x_n\} \) a set of indeterminates over \( k \). We have seen how to associate special subsets (i.e., varieties) of \( k^n \) with ideals of \( k[X] \) and vice versa. It is reasonable to ask such questions as: let \( V_1, V_2 \) be varieties, are \( V_1 \cap V_2 \) and \( V_1 \cup V_2 \) also varieties? Naturally the answers to such questions are related to an understanding of operations on ideals. We discuss the two cases cited in detail as preparation for the exercises.

Throughout we let \( I, J \) be ideals of \( k[X] \).

Definition 2.1 The sum of \( I \) and \( J \) is given by

\[
I + J = \{ f + g \mid f \in I, g \in J \}.
\]
Lemma 2.1 \( I + J \) is an ideal and \( V(I + J) = V(I) \cap V(J) \).

Proof \( I + J \) is not empty since it contains 0. Now suppose that \( h \in I + J \) and \( p \in k[X] \). Then \( h = f + g \) for some \( f \in I \) and \( g \in J \). Thus \( ph = pf + pg \in I + J \) since \( pf \in I \) and \( pg \in J \) (as they are both ideals). Finally if \( h_1, h_2 \in I + J \) then \( h_1 = f_1 + g_1 \) and \( h_2 = f_2 + g_2 \) for some \( f_1, f_2 \in I \) and \( g_1, g_2 \in J \). Thus \( h_1 - h_2 = (f_1 - f_2) + (g_1 - g_2) \in I + J \) again because \( I \) and \( J \) are ideals and are thus closed under subtraction. This proves that \( I + J \) is an ideal.

To prove that \( V(I + J) = V(I) \cap V(J) \) let \( a \in V(I + J) \). Then \( a \in V(I) \) since \( I \subseteq I + J \) and similarly \( a \in V(J) \). Thus \( a \in V(I) \cap V(J) \). For the reverse inclusion suppose that \( a \in V(I) \cap V(J) \). This means that \( f(a) = 0 \) for all \( f \in I \) and \( g(a) = 0 \) for all \( g \in J \). It follows that \( (f + g)(a) = 0 \) for all \( f \in I \) and \( g \in J \), i.e., \( h(a) = 0 \) for all \( h \in I + J \) and thus \( a \in V(I + J) \) as required. \( \square \)

Definition 2.2 The product of \( I \) and \( J \) is given by
\[
IJ = \{ fg \mid f \in I, g \in J \}.
\]

Note that in this definition we cannot just define the product as the set of all \( fg \) with \( f \in I \) and \( g \in J \) because this is (in general) not closed under sums. The correct definition takes all finite sums of such products, i.e.,
\[
IJ = \{ f_1g_1 + f_2g_2 + \cdots + f_mg_m \mid f_i \in I, g_i \in J, \text{ for } m \geq 1 \text{ and } 1 \leq i \leq m \}.
\]

It is worth noting here that we have \( IJ \subseteq I \) and \( IJ \subseteq J \) (can you see why?). We will return to this below.

Lemma 2.2 \( IJ \) is an ideal and \( V(IJ) = V(I) \cup V(J) \).

Proof \( IJ \) is not empty since it contains 0. Suppose that \( h \in IJ \) and \( p \in k[X] \). Then \( h = f_1g_1 + f_2g_2 + \cdots + f_mg_m \) for some \( f_1, f_2, \ldots, f_m \in I \) and \( g_1, g_2, \ldots, g_m \in J \). Thus \( ph = (pf_1)g_1 + (pf_2)g_2 + \cdots + (pf_m)g_m \in IJ \) since \( I \) is an ideal and thus closed under multiplication by elements of \( k[X] \). Finally if \( h_1, h_2 \in IJ \) then \( h_1 = f_1g_1 + f_2g_2 + \cdots + f_1g_1 + f_2g_2 + \cdots + f_2g_2 \) for some \( f_1 \in I \) and \( g_1 \in J \). Thus
\[
h_1 - h_2 = f_1g_1 + f_2g_2 + \cdots + f_1g_1 + f_2g_2 + \cdots + f_2g_2 \in IJ,
\]

since \( IJ \) is closed under negation.

To prove that \( V(IJ) = V(I) \cup V(J) \) let \( a \in V(IJ) \). Then \( f(a)g(a) = 0 \) for all \( f \in I \) and \( g \in J \). If \( f(a) = 0 \) for all \( f \in I \) then \( a \in V(I) \). Otherwise there is some \( f \in I \) such that \( f(a) \neq 0 \) in which case \( g(a) = 0 \) for all \( g \in J \) (since \( f(a), g(a) \in k, \) a field) and so \( a \in V(J) \). Thus \( a \in V(I) \cup V(J) \). For the reverse inclusion let \( a \in V(I) \cup V(J) \) and suppose w.l.o.g. that \( a \in V(I) \). Then \( f(a) = 0 \) for all \( f \in I \) and it follows that \( h(a) = 0 \) for all \( h \in IJ \) since \( h \) is a finite sum of products of members of \( I \) and \( J \). It follows that \( a \in V(IJ) \). \( \square \)

An obvious and important question to ask in connection with these constructions is how to find a basis for the constructed ideal given bases for \( I \) and \( J \). This turns out to be easy for these cases.

Lemma 2.3 Let \( f_1, f_2, \ldots, f_r \) be a basis for \( I \) and \( g_1, g_2, \ldots, g_s \) a basis for \( J \). Then

1. \( f_1, f_2, \ldots, f_r, g_1, g_2, \ldots, g_s \) is a basis for \( I + J \).
2. \( f_ig_j \) for \( 1 \leq i \leq r \) and \( 1 \leq j \leq s \) is a basis for \( IJ \).

Proof For the basis of \( I + J \) let \( H = (f_1, \ldots, f_r, g_1, \ldots, g_s) \). Clearly \( H \) contains both \( I \) and \( J \) as subsets and thus \( I + J \subseteq H \) since \( H \) is closed under addition. The reverse inclusion is obvious since the generators of \( H \) are contained in \( I + J \).

For the basis of \( IJ \) it is clear that the ideal \( H = (f_ig_j, 1 \leq i \leq r, 1 \leq j \leq s) \) is contained in \( IJ \). For the reverse inclusion it suffices to show that any product \( fg \) with \( f \in I \) and \( g \in J \) is in \( H \). We have \( f = \sum_{i=1}^r p_if_i \) and \( g = \sum_{j=1}^s q_jg_j \) for some polynomials \( p_i, q_j \). Thus \( fg = \sum_{i=1}^r \sum_{j=1}^s p_iq_jf_ig_j \) and this is in \( H \) as required. \( \square \)
The two examples given above are relatively straightforward and the question of finding bases is easily settled. We now move to look at a rather more intricate case.

§2.1. Intersection of Ideals. Since ideals are (special kinds of) sets we can take their intersection.

Exercise 2.1 Prove that $I \cap J$ is an ideal. \hfill [5 marks]

Note that we always have $IJ \subseteq I \cap J$ because $IJ \subseteq I$ and $IJ \subseteq J$ as observed above. However, equality need not hold as can be seen by taking $I = J = (x)$ in $k[x]$: we have $IJ = (x^2)$ but $I \cap J = (x)$ and $(x^2) \subset (x)$ since $x^2 \in (x)$ but $x \notin (x^2)$ [why?].

Exercise 2.2 Prove that $V(I \cap J) = V(I) \cup V(J)$. \hfill [5 marks]

Thus $V(IJ)$ and $V(I \cap J)$ are the same. We will see that finding a basis for $I \cap J$ from bases of $I$ and $J$ is quite hard. So why bother with this harder concept? The simple example given above (with $I = J = (x)$) provides a clue. Although $IJ$ and $I \cap J$ have the same zero sets as sets the latter gives more refined information. In our simple example the variety is just \{0\} but $IJ$ has this as a repeated root whereas $I \cap J$ captures it as a simple root. Recall that if $k$ is algebraically closed, Hilbert’s Nullstellensatz tells us that $f \in I \cap J$ if and only if $f^s \in I$ for some $s \geq 1$. The radical of $I$, denoted by $\sqrt{I}$, is defined to be all $f \in k[X]$ such that $f^s \in I$ for some $s \geq 1$; this is an ideal (the proof is not hard). The Nullstellensatz can now be restated as $\sqrt{I} = I \cap J$. It is thus reasonable to look for constructions that behave well in terms of taking radicals. $I \cap J$ is very well behaved in this regard whereas $IJ$ is not (we always have $\sqrt{I \cap J} = \sqrt{I} \cap \sqrt{J}$ but we need not have $\sqrt{IJ} = \sqrt{I} \sqrt{J}$ as the simple example shows).

We now discuss the question of finding a basis for $I \cap J$. Let $t$ be a new indeterminate over $k$. We will be considering elements of $k[t]$, $k[X]$ and of $k[X,t]$. In order to keep things clear we will use the argument based notation $u(t)$, $g(X)$ and $h(X,t)$ for the three types of elements (note that $h(X,t)$ need not involve $t$ or even members of $X$, the notation just indicates the possible indeterminates that it might involve). For $u(t) \in k[t]$ we will use $u(t)I$ to denote the ideal of $k[X,t]$ generated by $\{u(t)h \mid h \in I\}$ (remember that $I$ is an ideal of $k[X]$). The following result is fairly straightforward to prove, we omit the proof just to save a little space.

Lemma 2.4 Suppose that $I$ is generated by $f_1(X), f_2(X), \ldots, f_r(X)$ as an ideal of $k[X]$.

1. The ideal $u(t)I$ of $k[X,t]$ is generated by $u(t)f_1(X), u(t)f_2(X), \ldots, u(t)f_r(X)$.

2. If $h(X,t) \in u(t)I$ and $a \in k$ then $h(X,a) \in I$.

The two simple observations of the preceding lemma are very helpful in the following.

Exercise 2.3 Prove that $I \cap J = (tI + (1-t)J) \cap k[X]$. (Hint: Use the preceding Lemma as part of your proof.) \hfill [10 marks]

Having established the main result we can dispense with the clumsy argument based notation for elements. Suppose that $I$ is generated by $f_1, f_2, \ldots, f_r$ and $J$ by $g_1, g_2, \ldots, g_s$ (as ideals of $k[X]$).

Then the ideal $tI + (1-t)J$ of $k[X,t]$ is generated by $tf_1, \ldots, tf_r, (1-t)g_1, \ldots, (1-t)g_s$, this follows from Lemma 2.4 and Lemma 2.3. We are now ready to produce an algorithm for finding a basis for $I \cap J$.

Choose a lexicographic order with $t$ greater than $x_1, x_2, \ldots, x_n$ (these can be ordered in any way). Compute a Gröbner basis $G$ for $tI + (1-t)J$ using this order. The elements of $G$ that do not involve $t$ are a basis (actually a Gröbner basis) for $I \cap J$. To see this let $H$ be the elements of $G$ chosen as described (i.e., $H = G \cap k[X]$), it follows from the exercise that $H \subseteq I \cap J$.

Since $G$ is a Gröbner basis for $tI + (1-t)J$ and $I \cap J \subseteq tI + (1-t)J$ it follows that every element $f \in I \cap J$ reduces to 0 w.r.t. $G$. Of course the power products of any such $f$ are all free of $t$. On the other hand if we consider an element of $G$ that involves $t$ then its leading power product will also involve $t$ (because the order is lexicographic and $t$ is the largest indeterminate). Thus no such element can ever be used in reducing $f$ to 0, i.e., $f$ reduces to 0 w.r.t. $H$. Thus $H$ is a Gröbner
basis for \( I \cap J \) and it is a simple exercise (which you are advised to do) to see that a Gröbner basis for an ideal is a basis for it (i.e., it generates the ideal).

As a simple example consider \( I = (x^3y) \) and \( J = (xy^2) \); it is easy to see that \( I \cap J = (x^3y^2) \). We verify this by following the algorithm using Axiom’s function \texttt{groebner} with \( y <_L x <_L t \). The basis we obtain is \([tx^3y, txy^2 - xy^2, x^3y^2]\) and so \( H = [x^3y^2] \) as expected.

Note that in Axiom we tell the function \texttt{groebner} which order to use by giving the polynomials the appropriate type. In our case this is

\[
\text{DistributedMultivariatePolynomial([t,x,y],Integer)}
\]

which, mercifully, can be abbreviated to \texttt{DMP([t,x,y],INT)}. You can declare the (same) type of several variables in one go, e.g., \((f1,f2,g1,g2):DMP([t,x,y],INT)\). The procedure \texttt{groebner} takes a list of polynomials as its argument so the type of this must be \texttt{List(DMP([t,y,x],INT))}.

You can ensure this type in various ways but do check the type before calling the procedure, probably the safest thing is to assign the list to a variable and look at the type that Axiom returns (for a single variable \( B \) you can just use \( B:⟨\text{type}⟩:=⟨\text{value}⟩ \)).

**Exercise 2.4** We use polynomials with coefficients from \( \mathbb{Q} \) throughout.

1. Let \( I = (x^2 + y^2 - 1, xy) \) and \( J = (y - x, xy - 1) \). Find a basis for \( I \cap J \). Note that \( \mathbf{V}(I) \) consists of four points and \( \mathbf{V}(J) \) consists of two points (use the Axiom function \texttt{solve}). Find these points and check that the variety of your computed basis is the union of the points (this does not prove that the basis is correct but it does provide a reasonable check). For your answer just write down the basis.

2. Let \( I = (x) \), \( J_1 = (x^2, y) \) and \( J_2 = (x^2, xy, y^2) \). Find \( I \cap J_1 \) and show that this is the same as \( I \cap J_2 \). Since the three ideals are all generated by power products you should be able to see easily that the generators for the intersection belong to all the ideals (why?). For your answer just write down the common basis of the two intersections.

[5 marks]

**Notes.** If you have computed a basis \( G \) with the extra indeterminate \( t \) you can pick out the polynomials that do not involve \( t \) with the simple function

\[
\text{freeOf}(t,L)==
\]

\[
\text{R:=[]}
\]

\[
\text{for f in L repeat if not member?(t,variables(f)) then R:=cons(f,R) }
\]

\[
\text{R}
\]

You could try other examples, especially ones for which you can describe the varieties easily. An obvious source is to take linear polynomials with finitely many solutions (one set to generate \( I \) and another to generate \( J \)). Naturally the algorithm works in general but things get complicated: even here you can do some checking, e.g., the generators for \( I \cap J \) should be in \( I \) and in \( J \) and this can be checked using Gröbner bases for \( I \) and \( J \). In Axiom you can do this using the procedure \texttt{normalForm}, if \( G \) is the Gröbner basis then the normal form of \( f \) is obtained by \texttt{normalForm(f,G)}.

Kyriakos Kalorkoti, March 2017