The University of Sydney

MATH2008 Introduction to Modern Algebra

(http://www.maths.usyd.edu.au/u/UG/IM/MATH2008/)

Semester2, 2003 — Lecturer: R. Howlett

Computer Tutorial 4

This tutorial is about the vector space of polynomials over $\mathbb{R}$. There is one new MAGMA command you will need to use (`PolynomialAlgebra`) and one new *control structure* (`for ... end for`). And of course you can still use the commands you learnt in previous tutorials (e.g. `RealField VectorSpace print func Sqrt InnerProduct Arccos NullSpace sub KMatrixSpace Transpose Solution`).

Remember that all MAGMA commands end with a semicolon (`;`) (which can go on the next line, if necessary). You should frequently get MAGMA to print the values of the variables you are working with, to see the results of your actions. (e.g. if you define a subspace by a command like
`SS := sub< V | u, v >;`
then you should type `SS;` (or `print SS;`) to find out what MAGMA thinks `SS` is. To see what variables the MAGMA system has defined, type `ShowIdentifiers();`.

1. Use MAGMA to add and multiply the polynomials $f(x) = x^7 - 5x^4 + 2x - 1$ and $g(x) = 3x^3 - 2x^2 + x - 1$. To get started type
`R := RealField();`
`P<x> := PolynomialAlgebra(R);`
(Now MAGMA knows that `P` is the set of all polynomials in the variable `x`.)

- (*i*) The next step is to enter the polynomials themselves. That is,
`f := x^7 - 5*x^4 + 2*x - 1;`
`g := 3*x^3 - 2*x^2 + x - 1;`
- (*ii*) Print out the values of `P`, `f` and `g`.
- (*iii*) Next, print the sum and product of `f` and `g`.
- (*iv*) MAGMA has a function `Eltseq` which returns the sequence of coefficients of a polynomial. To see how it works, type `Eltseq(f);`. Note that the constant term comes first, then the coefficient of x^1, and so on. You can also define a polynomial by entering its sequence of coefficients, and "coercing" this sequence into the polynomial algebra. Test this by typing `h:=P![-1,1,-2,3];` and then `print h;`.

Solution.

```
> R := RealField();
> P<x> := PolynomialAlgebra(R);
> f := x^7 - 5*x^4 + 2*x - 1;
> g := 3*x^3 - 2*x^2 + x - 1;
```



```
> P,f,g;
Univariate Polynomial Ring in x over Real Field
x^7 - 5*x^4 + 2*x - 1
3*x^3 - 2*x^2 + x - 1
> f+g, f*g;
x^7 - 5*x^4 + 3*x^3 - 2*x^2 + 3*x - 2
3*x^10 - 2*x^9 + x^8 - 16*x^7 + 10*x^6 - 5*x^5 + 11*x^4 - 7*x^3
                         + 4*x^2 - 3*x + 1
> Eltseq(f);
[ -1, 2, 0, 0, -5, 0, 0, 1 ]
> h:=P![-1,1,-2,3];
> h;
3*x^3 - 2*x^2 + x - 1
> h eq g;
true
```

2. (*i*) If $f(x)$ is a polynomial then the mathematical notation for the number obtained by putting $x = 3$ (say) is $f(3)$. MAGMA has a function `Evaluate` for this: in MAGMA, if `f` is a polynomial then `Evaluate(f,3)` evaluates `f` at $x = 3$. Evaluate `f`, `g`, `f-g` and `f*g` at $x = 0$, 1 and 10.

- (*ii*) MAGMA can integrate and differentiate polynomials with the commands `Integral` and `Derivative`. We can use `Integral` to define a MAGMA function that returns the inner product of two polynomial functions on the interval $[-1, 1]$, using the inner product defined in lectures, namely $(f, g) = \int_{-1}^{1} f(x)g(x)\,dx$. The MAGMA code you need is

```
polyip := func< a,b| Evaluate(int,1) - Evaluate(int,-1)
             where int:=Integral(a*b) >;
```

  Use `polyip` to find the inner product of the polynomials `f` and `g` of the previous exercise.
- (*iii*) Calculate the *length* of `f` and the *angle* between `f` and `g`. (Recall that in any inner product space the length of an element v is defined to be $\sqrt{(v, v)}$, and the angle between v and w is $\arccos((v, w)/\|v\|\,\|w\|)$. Start by defining `Length:=func< v | Sqrt(polyip(v,v)) >;`).

Solution.

```
> Evaluate(f,0); Evaluate(g,0);
-1
-1
> Evaluate(f+g,0); Evaluate(f*g,0);
-2
1
```

```
> Evaluate(f,1); Evaluate(g,1);
-3
1
> Evaluate(f+g,1); Evaluate(f*g,1);
-2
-3
> Evaluate(f,10); Evaluate(g,10);
9950019
2809
> Evaluate(f+g,10); Evaluate(f*g,10);
9952828
27949603371
> polyip:=func<a,b| Evaluate(int,1) - Evaluate(int,-1) where
int:=Integral(a*b);
> polyip(f,g);
43976/3465
> Length:=func< v | Sqrt(polyip(v,v)) >;
> Length(f);
3.90441345715901922389813612O
> Arccos(polyip(f,g)/(Length(f)*Length(g)));
0.3419620528286725421961978426
```

3. Find the inner products (x^k, x^k) where k ranges from 0 to 10. You can use a MAGMA control structure called the "for loop". You put the commands you want to repeat between the words "`for`" and "`end for`". Your work will look like this:
`> for k in [0..10] do`
`for>` *The command for printing* (x^k, x^k) *goes here. Remember the semicolon.*
`for> end for;`
Notice that after you begin a for loop, MAGMA changes its prompt to remind you that you must finish it. You finish the for loop with the two-word command "`end for`" followed by a semicolon, but there is no semicolon after the `do`.

Solution.

```
> for k in [1..10] do
for> k, polyip(x^k,x^k);
for> end for;
1 2/3
2 2/5
3 2/7
4 2/9
```

```
5 2/11
6 2/13
7 2/15
8 2/17
9 2/19
10 2/21
```


4. For what values of k and m are x^k and x^m orthogonal? To answer this, use MAGMA to deal with the first few cases (say for k and m ranging from 1 to 5) and then make a conjecture.

Hint. You can begin with
`for k in [1..5] do`
`for m in [1..5] do`
...

Solution.

```
> for k in [1..5] do
for> for m in [1..5] do
for|for> print "(x^",k,",x^",m,") = ",polyip(x^k,x^m);
for|for> end for;
for> end for;
(x^ 1 ,x^ 1 ) = 2/3
(x^ 1 ,x^ 2 ) = 0
(x^ 1 ,x^ 3 ) = 2/5
(x^ 1 ,x^ 4 ) = 0
(x^ 1 ,x^ 5 ) = 2/7
(x^ 2 ,x^ 1 ) = 0
  ... etc. ...
(x^ 5 ,x^ 4 ) = 0
(x^ 5 ,x^ 5 ) = 2/11
```

(The print command used above could be replaced by the simpler
`polyip(x^k,x^m);`
but then MAGMA would just print the 25 numbers 2/3, 0, 2/5, ... , and it would not be immediately obvious which inner products have which values. So it is better at least to get MAGMA to print `k` and `m` as well as `(x^k,x^m)` each time:
`k, m, polyip(x^k,x^m);`
You can also get MAGMA to print words or symbols literally by enclosing them in quotes, as done above.)

You should guess from the above output that if k and m are positive integers then $(x^k, x^m) = 0$ if and only if $k+m$ is odd. You should also be be able to prove this mathematically. Indeed,

$$\int_{-1}^{1} x^{k+m}\,dx = \frac{1}{k+m+1}(1^{k+m+1} - (-1)^{k+m+1}),$$

which is zero if $k+m+1$ is even and is $2/(k+m+1)$ if $k+m+1$ is odd.

5. Check that $1+2x^2$ and x^3 are orthogonal and then find the projections of x^4 and x^5 onto the subspace spanned by $1+2x^2$ and x^3. Explain why in both cases you get a multiple of one of the original vectors.

Hint. Remember that the projection p of the vector v onto the subspace with the orthogonal basis a_1, a_2, ... , a_k is

$$p = \frac{(v, a_1)}{(a_1, a_1)} a_1 + \frac{(v, a_2)}{(a_2, a_2)} a_2 + \cdots + \frac{(v, a_k)}{(a_k, a_k)} a_k.$$

In this question, k is 2, $a_1 = 1 + 2x^2$ and $a_2 = x^3$.

Solution.

```
> p := 1+2*x^2;
> q := x^3;
> polyip(p,q);
0
> (polyip(x^4,p)/polyip(p,p))*p+(polyip(x^4,q)/polyip(q,q))*q;
102/329*x^2 + 51/329
> (polyip(x^5,p)/polyip(p,p))*p+(polyip(x^5,q)/polyip(q,q))*q;
7/9*x^3
```

The projection of x^4 is a multiple of p since $(x^4, q) = 0$. And the projection of x^5 is a multiple of q since x^5 is orthogonal to p.

6. (*i*) The aim of this question is to apply the Gram-Schmidt process to the polynomials 1, x, x^2, x^3, x^4, x^5 and x^6, and hence obtain the first seven Legendre polynomials.

A MAGMA input file t4defs.m has been prepared containing the following MAGMA code (which you are not required to understand):

```
polyProjection := func< Q,g | #Q eq 0 select 0*g else
        &+[(polyip(f,g)/polyip(f,f))*f :  f in Q] >;
polyGS := func< Q | [f-polyProjection(Self(),f) : f in Q]>;
```

The function `polyGS` does this: if `B` is a sequence of polynomials then `polyGS(B)` is the orthogonal sequence produced by Gram-Schmidt.

Type `load "t4defs.m";` to get MAGMA to read the input file. Next, define `B` to be the sequence of polynomials consisting of the powers of `x` from `x^0` up to `x^6`. You can do this by

```
B:=[x^0,x^1,x^2,x^3,x^4,x^5,x^6];
```

or more sneakily by

```
B := [x^k :  k in [0..6]];
```

Then apply `polyGS` to `B`.

(*ii*) Actually, Legendre polynomials are usually scaled so that their value at 1 is 1. If we are given a polynomial f and we define a new polynomial g by $g(x) = cf(x)$, where c is a scalar, then of course $g(1) = cf(1)$. So we can make $g(1) = 1$ by putting $c = 1/f(1)$. In MAGMA the corresponding command is `g:= (1/Evaluate(f,1))*f;`. Do this scaling on the polynomials you obtained in Part (*i*).

Solution.

```
> load "t4defs.m";
> B := [ x^k :  k in [0..6] ];
> G := polyGS(B);
> G;
[
    1,
    x,
    x^2 - 1/3,
    x^3 - 3/5*x,
    x^4 - 6/7*x^2 + 3/35,
    x^5 - 10/9*x^3 + 5/21*x,
    x^6 - 15/11*x^4 + 5/11*x^2 - 5/231
]
> for k in [1..7] do G[k] := G[k]/Evaluate(G[k],1);
for> end for;
> G;
[
    1,
    x,
    3/2*x^2 - 1/2,
    5/2*x^3 - 3/2*x,
    35/8*x^4 - 15/4*x^2 + 3/8,
    63/8*x^5 - 35/4*x^3 + 15/8*x,
    231/16*x^6 - 315/16*x^4 + 105/16*x^2 - 5/16
]
```

Note that MAGMA uses the notation `G[1]`, `G[2]` (etc.) for the 1st, 2nd (etc.) terms of a sequence `G`. Thus in this case `G[1]` is the polynomial `1`, `G[2]` is `x`, `G[3]` is `(3/2)x^2-1/2`, and so on.