THE UNIVERSITY OF SYDNEY

MATH2008 Introduction to Modern Algebra

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

Semester 2, 2003 Lecturer: R. Howlett

Computer Tutorial 7

The following commands will be used in today's tutorial. `Sym`, `Alt`, `Order`, `#`, `Set`, `Stabilizer`, `diff`, `meet`, `if ... then ... end if`.

1. Let G be the group of all permutations of $\{1, 2, 3, 4\}$ and let H be the stabilizer of 2 in G. That is, H consists of all permutations in G that leave 2 fixed. These groups can be set up in MAGMA with the commands

```
G := Sym(4);
H := Stabilizer(G,2);
```

(*i*) What are the orders of H and G?

(*ii*) Print all the elements of H. (In MAGMA the command `Set(H)` will produce the elements of H.)

(*iii*) Choose any element $y \in G$ and create the set obtained by multiplying every element of H by y (on the right). This is written as Hy and called a *right coset* of H. The MAGMA command to produce this set and name it `C` is
`C := { h*y : h in H };`
Print `C` and compare it with `H`. Is it the same size? Is it the same as `H`? How many permutations do they have in common? Does `y` belong to `C`?

(*iv*) Choose x in `C` (with $x \neq y$) and form the coset
`D := { h*x : h in H };`
Compare `D` with `C`. How many elements do they have in common?

(*v*) Is there a common property that the elements of `C` share? (Hint. Examine `2^x` for all `x` in `C`.)

(*vi*) Now let's be a bit more systematic. We shall create cosets $C1$, $C2$, ... , until every element of G is in one of these cosets. Begin by setting $C1$ equal to C. The MAGMA command is
`C1 := C;`
In MAGMA, if `X` and `Y` are sets then `X diff Y` is the set of all elements of `X` that are not in `Y`. (Think of "diff" as meaning "different from".) Define
`Z := Set(G) diff C1;`
and print the elements of `Z`. Now choose any element of `Z` (call it `y2`) and form the coset
`C2 := { h*y2 : h in H };`
Now redefine `Z := Z diff C2`, so that now `Z` consists of the elements that are not in either `C1` or `C2`, choose some `y3` in `Z`, and form its coset `C3`. Keep going like this until every element of G is in one of your cosets.

(a) How many cosets do you have?

(b) What is the size of each of your cosets?

(c) How much overlap is there between your cosets? (If `X` and `Y` are sets, their intersection is given by `X meet Y`.)

(d) Does the original subgroup `H` appear in your list of cosets? Why is that?

Solution.

```
> G := Sym(4);
> H := Stabilizer(G,2);
> #G, #H;
24 6
> Set(H);
{
    Id(H),
    (1, 4, 3),
    (3, 4),
    (1, 3, 4),
    (1, 3),
    (1, 4)
}
```

```
> y:=G!(1,2);
> C:={ h*y : h in H };
> C;
{
    (1, 3, 4, 2),
    (1, 4, 3, 2),
    (1, 2),
    (1, 4, 2),
    (1, 2)(3, 4),
    (1, 3, 2)
}
> // C, H both have 6 elements
> // y is an element of C
```

Whatever choice you make for the element y, it will always turn out that `C` has 6 elements, the same as `H`. If `y` happens to be in `H` then you will find that `C = H`, otherwise `C` and `H` will have no elements in common.

```
> x := G!(1,4,2);
> D := { h*x : h in H };
> D;
{
    (1, 4, 3, 2),
    (1, 3, 4, 2),
    (1, 2),
    (1, 4, 2),
    (1, 3, 2),
    (1, 2)(3, 4)
}
```

```
> D eq C;
true
> for z in C do
for> print 2^z;
for> end for;
1
1
1
1
1
1
```

It will always be the case that `D = C`, no matter which element `x` in `C` you choose. This is a general fact about cosets: if H is a subgroup and $C = Hy$ any right coset of H, then $Hx = C$ for all elements $x \in C$.

In this particular example (for the element `y` that was chosen) the coset `C` consists of all elements `z` in `G` such that `2^z = 1`. There are three other cosets that could have been obtained by choosing `y` differently:

(a) the set of all `z` with `2^z = 2`;

(b) the set of all `z` with `2^z = 3`;

(c) the set of all `z` with `2^z = 4`.

Observe that the first of these is equal to `H` (by the definition of `H`).

```
> C1 := C;
> Z := Set(G) diff C1;
> Z;
{
    (2, 3, 4),
    (1, 4)(2, 3),
    (2, 4),
    (1, 3, 4),
    (1, 3, 2, 4),
    (1, 2, 4, 3),
    (3, 4),
    (1, 2, 3, 4),
    (1, 2, 4),
    (1, 3),
    (1, 4, 2, 3),
    (2, 4, 3),
    (1, 3)(2, 4),
    (1, 2, 3),
    Id(G),
    (1, 4, 3),
    (2, 3),
    (1, 4)
}
> y2 := G!(2,3,4);
> C2 := { h*y2 : h in H };
> Z := Z diff C2;
> Z;
{
    (1, 3)(2, 4),
    (1, 2, 4, 3),
    (1, 3, 4),
    (1, 4, 3),
    (1, 3),
    (3, 4),
    (1, 2, 4),
    (2, 4),
    (1, 4),
    (1, 3, 2, 4),
    Id(G),
    (2, 4, 3)
}
> y3 := G!(1,3)(2,4);
> C3 := { h*y3 : h in H };
> Z := Z diff C3;
> Z;
{
    Id(G),
    (1, 4, 3),
    (3, 4),
    (1, 3, 4),
    (1, 3),
    (1, 4)
}
> y4 := Id(G);
> C4 := { h*y4 : h in H };
> Z := Z diff C4;
> Z;
{}
> C1 meet C2, C1 meet C3,
> C1 meet C4;
{}
{}
{}
> C2 meet C3, C2 meet C4,
> C3 meet C4;
{}
{}
{}
> C4 eq Set(H);
true
```

There are four cosets, they have six elements each, and they do not overlap at all. A subgroup is always a right coset of itself: indeed $Hh = H$ whenever h is an element of the subgroup H. In our example the coset `C4` is equal to `H`.

2. Let G be a cyclic group generated by an element of order 12. For example, `G := PermutationGroup< 12 | (1,2,3,4,5,6,7,8,9,10,11,12) >;` (This is the same as `G:=sub<Sym(12) | (1,2,3,4,5,6,7,8,9,10,11,12)>`)

(i) Print the elements of G and determine the order of each element.

(ii) Check that in this group two elements that have the same order always generate the same cyclic subgroup of G.

(iii) Which elements of G generate all of G? Hint: Try the MAGMA code

```
for x in G do
if sub< G | x > eq G then print x; end if;
end for;
```

Solution.

```
> G:=PermutationGroup<12|(1,2,3,4,5,6,7,8,9,10,11,12)>;
> for g in G do
for> print g,"has order",Order(g);
for> end for;
Id(G)
has order 1
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)
has order 12
(1, 3, 5, 7, 9, 11)(2, 4, 6, 8, 10, 12)
has order 6
(1, 4, 7, 10)(2, 5, 8, 11)(3, 6, 9, 12)
has order 4
(1, 5, 9)(2, 6, 10)(3, 7, 11)(4, 8, 12)
has order 3
(1, 6, 11, 4, 9, 2, 7, 12, 5, 10, 3, 8)
has order 12
(1, 7)(2, 8)(3, 9)(4, 10)(5, 11)(6, 12)
has order 2
(1, 8, 3, 10, 5, 12, 7, 2, 9, 4, 11, 6)
has order 12
(1, 9, 5)(2, 10, 6)(3, 11, 7)(4, 12, 8)
has order 3
(1, 10, 7, 4)(2, 11, 8, 5)(3, 12, 9, 6)
has order 4
(1, 11, 9, 7, 5, 3)(2, 12, 10, 8, 6, 4)
has order 6
(1, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2)
has order 12
> for x in G do
for> for y in G do
for|for> if Order(y) eq Order(x) then
for|for|if> print sub < G | x > eq sub < G | y >;
for|for|if> end if;
for|for> end for;
for> end for;
true
 ... 27 similar lines omitted ...
true
```

Since the order of an element is always the same as the order of the cyclic subgroup it generates, an element x in `G` will generate the whole of `G` if and only if the order of x is 12. We have already seen that there are exactly four such elements. (They happen to all be 12 cycles.)

```
> for x in G do
for> if sub< G | x > eq G then print x;
for|if> end if; end for;
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)
(1, 6, 11, 4, 9, 2, 7, 12, 5, 10, 3, 8)
(1, 8, 3, 10, 5, 12, 7, 2, 9, 4, 11, 6)
(1, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2)
```

3. The *alternating* group Alt(n) consists of all even permutations of $\{1, 2, \ldots, n\}$. (A permutation is even if its diagram has an even number of line crossings. An equivalent condition is that the permutation can be expressed as a product of an even number of transpositions (i, j).) Alt(n) is a subgroup of Sym(n) and contains exactly half the elements of Sym(n).

(i) Check that for $n = 5$ the alternating group is half the size of the symmetric group. The MAGMA command to create it is
`A := Alt(5);`

(ii) For each $n \in \{1, 2, 3, 4, 5, 6\}$, find the number of elements of `A` of order n.

(iii) Find two elements of `A` of order 2 whose product has order 3, and find the order of the subgroup they generate.

(iv) Find two elements of `A` of order 2 whose product has order 5, and find the order of the subgroup they generate.

(v) Find two elements of `A` of order 2 whose product has order 2, and find the order of the subgroup they generate.

Solution.

```
> S := Sym(5);
> A := Alt(5);
> #S, #A;
120 60
> Ord1 := { x : x in A | Order(x) eq 1 };
> Ord2 := { x : x in A | Order(x) eq 2 };
> Ord3 := { x : x in A | Order(x) eq 3 };
> Ord4 := { x : x in A | Order(x) eq 4 };
> Ord5 := { x : x in A | Order(x) eq 5 };
> Ord6 := { x : x in A | Order(x) eq 6 };
> #Ord1, #Ord2, #Ord3, #Ord4, #Ord5, #Ord6;
1 15 20 0 24 0
```



```
> for x in Ord2 do
for> for y in Ord2 do
for|for> if Order(x*y) eq 5 then
for|for|if> x, y;
for|for|if> #sub< A | x, y>;
for|for|if> break x;
for|for|if> end if;
for|for> end for;
for> end for;
(1, 4)(2, 3)
(1, 2)(4, 5)
10
> for x in Ord2 do
for> for y in Ord2 do
for|for> if Order(x*y) eq 3 then
for|for|if> x, y;
for|for|if> #sub< A | x, y>;
for|for|if> break x;
for|for|if> end if;
for|for> end for;
for> end for;
(1, 4)(2, 3)
(2, 3)(4, 5)
6
> for x in Ord2 do
for> for y in Ord2 do
for|for> if Order(x*y) eq 2 then
for|for|if> x, y;
for|for|if> #sub< A | x, y>;
for|for|if> break x;
for|for|if> end if;
for|for> end for;
for> end for;
(1, 4)(2, 3)
(1, 2)(3, 4)
4
```

In this code, the command `break x` tells MAGMA not to continue testing any more values of `x`. If we had just said `break` then MAGMA would have terminated the inner loop (the `y` loop) only.

You could not have been expected to know the `break` command, since it had never been mentioned. But you can get the answer quickly enough by just printing out all the elements of order 2 and systematically computing the orders of products of two elements of order 2 until suitable pairs are found. This can also be done rather quickly with pen and paper, making no use of MAGMA at all.

```
> Ord2;
{
    (1, 4)(2, 3),
    (1, 2)(4, 5),
    (1, 3)(4, 5),
    (2, 3)(4, 5),
    (1, 5)(3, 4),
    (1, 5)(2, 4),
    (1, 5)(2, 3),
    (1, 3)(2, 5),
    (2, 5)(3, 4),
    (1, 2)(3, 4),
    (1, 2)(3, 5),
    (1, 3)(2, 4),
    (2, 4)(3, 5),
    (1, 4)(2, 5),
    (1, 4)(3, 5)
}
```

```
> t:=A!(1,4)(2,3);
> Order(t*A!(1,2)(4,5));
5
> Order(t*A!(1,3)(4,5));
5
> Order(t*A!(2,3)(4,5));
3
> Order(t*A!(1,5)(3,4));
5
> Order(t*A!(1,5)(2,4));
5
> Order(t*A!(1,5)(2,3));
3
> Order(t*A!(1,3)(2,5));
5
> Order(t*A!(2,5)(3,4));
5
> Order(t*A!(1,2)(3,4));
2
```

4. If G is a group and y is an element of G then the set $\{x \in G \mid xy = yx\}$ is called the *centralizer* of y in G. That is, the centralizer of y is the set of all elements of G that commute with y. It is always a subgroup of G. The MAGMA command to create it is
`C := Centralizer(G,y);`
Try the following example;
`G := Sym(8);`
`y := G!(1, 3, 2, 7)(4, 6, 8, 5);`
Get MAGMA to print the centralizer, its order and all of its elements.

Solution.

```
> G:=Sym(8);
> y := G!(1, 3, 2, 7)(4, 6, 8, 5);
> C := Centralizer(G,y);
> C;
Permutation group C acting on a set of cardinality 8
Order = 32 = 2^5
    (1, 4)(2, 8)(3, 6)(5, 7)
    (4, 6, 8, 5)
> Set(C);
{
    (1, 3, 2, 7),
    (1, 7, 2, 3)(4, 8)(5, 6),
```



```
    (1, 5, 7, 8, 2, 6, 3, 4),
    (1, 4, 2, 8)(3, 6, 7, 5),
    (1, 6, 7, 4, 2, 5, 3, 8),
    (1, 2)(3, 7),
    (1, 5, 2, 6)(3, 4, 7, 8),
    (4, 6, 8, 5),
    (1, 4)(2, 8)(3, 6)(5, 7),
    (1, 6, 3, 8, 2, 5, 7, 4),
    (1, 8, 7, 6, 2, 4, 3, 5),
    (1, 3, 2, 7)(4, 6, 8, 5),
    (1, 4, 7, 5, 2, 8, 3, 6),
    (1, 6)(2, 5)(3, 8)(4, 7),
    (4, 5, 8, 6),
    (1, 7, 2, 3),
    (1, 2)(3, 7)(4, 6, 8, 5),
    (4, 8)(5, 6),
    (1, 3, 2, 7)(4, 5, 8, 6),
    (1, 3, 2, 7)(4, 8)(5, 6),
    (1, 8)(2, 4)(3, 5)(6, 7),
    (1, 2)(3, 7)(4, 5, 8, 6),
    (1, 5)(2, 6)(3, 4)(7, 8),
    (1, 2)(3, 7)(4, 8)(5, 6),
    (1, 8, 2, 4)(3, 5, 7, 6),
    (1, 7, 2, 3)(4, 6, 8, 5),
    (1, 8, 3, 5, 2, 4, 7, 6),
    (1, 4, 3, 6, 2, 8, 7, 5),
    (1, 6, 2, 5)(3, 8, 7, 4),
    (1, 5, 3, 4, 2, 6, 7, 8),
    Id(C),
    (1, 7, 2, 3)(4, 5, 8, 6)
}
```