

## 2 Chapter: Groups

Let  $U(n)$  be the set of all positive integers less than  $n$  and relatively prime to  $n$ . The set  $U(n)$  is a group under multiplication modulo  $n$ .

For this chapter you will need the file named “ulist”. (Thanks goes to Alexander Hulpke for pointing out this revised version of “ulist”.) Fetch the file “ulist” from the web site for this manual and place the file in the gap4r4 folder. The appendix at the end of this chapter contains a print out of the file “ulist.” To use the file “ulist” while in GAP type:

```
gap> Read("ulist");
```

This command reads in a copy of the file “ulist”.

*Careful:* If you exit GAP and then reenter GAP you will need to read in the file “ulist” again, if you want to continue to use it. For any  $n$  the file “ulist” contains a function, also called `ulist`, which will list the elements of  $U(n)$ . For example:

```
gap> ulist(100);
[ ZmodnZObj( 1, 100 ), ZmodnZObj( 3, 100 ), ZmodnZObj( 7, 100 ),
  ZmodnZObj( 9, 100 ), ZmodnZObj( 11, 100 ), ZmodnZObj( 13, 100 ),
  ZmodnZObj( 17, 100 ), ZmodnZObj( 19, 100 ), ZmodnZObj( 21, 100 ),
  ZmodnZObj( 23, 100 ), ZmodnZObj( 27, 100 ), ZmodnZObj( 29, 100 ),
  ZmodnZObj( 31, 100 ), ZmodnZObj( 33, 100 ), ZmodnZObj( 37, 100 ),
  ZmodnZObj( 39, 100 ), ZmodnZObj( 41, 100 ), ZmodnZObj( 43, 100 ),
  ZmodnZObj( 47, 100 ), ZmodnZObj( 49, 100 ), ZmodnZObj( 51, 100 ),
  ZmodnZObj( 53, 100 ), ZmodnZObj( 57, 100 ), ZmodnZObj( 59, 100 ),
  ZmodnZObj( 61, 100 ), ZmodnZObj( 63, 100 ), ZmodnZObj( 67, 100 ),
  ZmodnZObj( 69, 100 ), ZmodnZObj( 71, 100 ), ZmodnZObj( 73, 100 ),
  ZmodnZObj( 77, 100 ), ZmodnZObj( 79, 100 ), ZmodnZObj( 81, 100 ),
  ZmodnZObj( 83, 100 ), ZmodnZObj( 87, 100 ), ZmodnZObj( 89, 100 ),
  ZmodnZObj( 91, 100 ), ZmodnZObj( 93, 100 ), ZmodnZObj( 97, 100 ),
  ZmodnZObj( 99, 100 ) ]
```

The output `ZmodnZObj( 3, 100 )`, for example, means the element 3 mod 100.

### Exercises

2.1 Using GAP determine the size of  $U(n)$  for  $n = 9, 27, 81, 243, 5, 25, 125$ . Make a conjecture about the size of  $U(p^k)$  where  $p$  is a prime not equal to 2 and  $k$  is a positive integer. Do not count the elements in  $U(n)$ , instead use `Size` to make GAP count the elements for you! [Gallian, Chapter 2, Computer Exercise 2]

2.2 Using GAP determine the size of  $U(n)$  for  $n = 18, 54, 162, 486, 50, 250, 98, 242$ . Make a conjecture about the relationship between the size of  $U(2p^k)$  and the size of  $U(p^k)$  where  $p$  is a prime

not equal to 2. [Gallian, Chapter 2, Computer Exercise 2]

2.3 Let  $r$  and  $s$  be relatively prime integers. Use **GAP** to help you make a conjecture about the size of  $U(rs)$  in terms of the sizes of  $U(r)$  and  $U(s)$ .

2.4 Recall from Chapter 0 how to do modular arithmetic in **GAP**. Use the function **Gcdex** to find the inverses of the elements in  $U(100)$ . For example, to find the inverse of 3 in  $U(100)$  use **Gcdex(3,100)**.

The command **GL(n,p)** creates the general linear group of  $n \times n$  invertible matrices with entries in  $\mathbf{Z}_p$  and **SL(n,p)** creates the special linear group of  $n \times n$  invertible matrices with entries in  $\mathbf{Z}_p$  and determinate equal to one. For example the following creates **GL(3, Z<sub>5</sub>)** and **SL(3, Z<sub>5</sub>)**:

```
gap> g:= GL(3,5);
GL(3,5)
gap> s:= SL(3,5);
SL(3,5)
```

We can use our **Size** command to find the number of elements (the order) in these groups:

```
gap> Size(g);
1488000
gap> Size(s);
372000
```

### *Exercises*

2.5 Find the number of elements in **GL(2, Z<sub>p</sub>)** and **SL(2, Z<sub>p</sub>)** for  $p = 3, 5, 7$  and 11. What relationship do you see between the orders of **GL(2, Z<sub>p</sub>)** and **SL(2, Z<sub>p</sub>)** and  $p-1$ ? Does this relationship hold for  $p = 2$ ? Based on these examples does it appear that  $p$  always divides the order of **SL(2, Z<sub>p</sub>)**? What about  $p-1$ ? What about  $p+1$ ? Guess a formula for the order of **SL(2, Z<sub>p</sub>)**. Guess a formula for the order of **GL(2, Z<sub>p</sub>)**. [Gallian, Chapter 2, Computer Exercise 4].

### **Appendix for Chapter 2**

The following is the file “ulist” which is used in this chapter:

```
ulist:= function(n)
local s,i,o;
o:= One(Integers mod n);
s:= n-> Filtered([1..n-1], i -> Gcd(i,n) = 1);
return s(n)*o;
end;
```