# Using *Mathematica* to Explore Abstract Algebra

Al Hibbard (Central College - hibbarda@central.edu)
Ken Levasseur (UMass-Lowell - Kenneth_Levasseur@uml.edu)

**http://www.central.edu/eaam/AAOverview2.nb**

## Startup

First, we load the `Master` package, which will load in all of the names that are used in the *AbstractAlgebra* packages. **(Note that these packages can be freely downloaded from the EAAM web site.)**

```
Needs["AbstractAlgebra`Master`"]
```

Since we will first consider groups, we switch the structure to `Group`.

```
SwitchStructureTo[Group]
```

```
Group
```

## The Basic Structures

There are three basic *Mathematica* data structures used in *AbstractAlgebra*: the `Groupoid`, `Ringoid`, and `Morphoid`. These are generalizations of groups, rings and morphisms. We will use groupoid, ringoid and function when we refer to the mathematical counterparts to the corresponding *Mathematica* data structures.

# Groupoids

A `Groupoid` consists of a set of elements and a “binary” operation (the image of two inputs from the space does not need to belong to this space). One means of creating one of these is with the `FormGroupoid` function.

```
G = FormGroupoid[{0, 2, 1, 4, 6}, Times, GroupoidName → "ex.1"]
```

```
Groupoid[{0, 2, 1, 4, 6}, -Operation-]
```

We can easily extract the operation and elements of any groupoid.

```
Operation[G]
Elements[G]
```

```
Times
```

```
{0, 2, 1, 4, 6}
```

We are often interested in whether a groupoid has an identity element or not.

```
HasIdentityQ[G]
```

```
True
```

Many functions can take on additional `Modes`, such as `Textual` or `Visual`.

```
HasIdentityQ[G, Mode → Visual]
```

ex.1 x ∗ y

| x \ y | 0 | 2 | 1 | 4 | 6 |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 4 | 2 | 8 | 12 |
| 1 | 0 | 2 | 1 | 4 | 6 |
| 4 | 0 | 8 | 4 | 16 | 24 |
| 6 | 0 | 12 | 6 | 24 | 36 |

red: left ident.

ex.1 x ∗ y

| x \ y | 0 | 2 | 1 | 4 | 6 |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 4 | 2 | 8 | 12 |
| 1 | 0 | 2 | 1 | 4 | 6 |
| 4 | 0 | 8 | 4 | 16 | 24 |
| 6 | 0 | 12 | 6 | 24 | 36 |

red: right ident.

```
True
```

Another group axiom to consider is whether all the elements have inverses. Here we illustrate the `Textual` mode.

```
HasInversesQ[G, Mode → Textual]
```

Given a Groupoid G, we say an element g in G has an inverse h if G has an identity e and g ∗ h = h ∗ g = e (where ∗ indicates the operation).

The Groupoid ex.1 contains some elements without inverses. For example, 0 does NOT have an inverse.

```
False
```

Closure is another required property of being a group. Here is the `Visual` mode of this Boolean function.

```
ClosedQ[G, Mode → Visual]
```

```
All the elements marked with yellow are original
  elements in the set. Those in red are from outside.
```

ex.1 x ⋆ y

| x \ y | 0 | 2 | 1 | 4 | 6 |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 4 | 2 | 8 | 12 |
| 1 | 0 | 2 | 1 | 4 | 6 |
| 4 | 0 | 8 | 4 | 16 | 24 |
| 6 | 0 | 12 | 6 | 24 | 36 |

```
False
```

Finally, the last required property is associativity. Here is the `Textual` mode.

```
AssociativeQ[G, Mode → Textual]
```

```
Given a structured set S (Groupoid or Ringoid), we
  say the operation * is associative if for every g, h, and
  k in S we have (g*h)*k = g*(h*k), where * is the operation.

In this case, ex.1 is associative.

Consider the following table illustrating random
  triples that associate.Pay attention to the last two columns.
```

| i | j | k | (i*j)*k | i*(j*k) |
|---|---|---|---|---|
| 2 | 0 | 6 | 0 | 0 |
| 2 | 6 | 4 | 48 | 48 |
| 4 | 2 | 4 | 32 | 32 |
| 6 | 6 | 2 | 72 | 72 |
| 4 | 0 | 2 | 0 | 0 |
| 0 | 6 | 4 | 0 | 0 |
| 2 | 2 | 4 | 16 | 16 |
| 0 | 2 | 2 | 0 | 0 |
| 2 | 4 | 6 | 48 | 48 |
| 6 | 4 | 2 | 48 | 48 |

```
True
```

Instead of testing the axiomatic properties individually, we can also test these together with one function.

```
GroupQ[G]
```

```
False
```

The Cayley table is a tool that can reveal a number of interesting properties regarding a group.

```
CayleyTable[G, Mode → Visual]
```

For each element, a different color is used. The entries in the table corresponding to the elements are then colored and labeled accordingly.

ex.1 x * y

| x \ y | 0 | 2 | 1 | 4 | 6 |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 4 | 2 | 8 | 12 |
| 1 | 0 | 2 | 1 | 4 | 6 |
| 4 | 0 | 8 | 4 | 16 | 24 |
| 6 | 0 | 12 | 6 | 24 | 36 |

```
{{0, 0, 0, 0, 0}, {0, 4, 2, 8, 12},
 {0, 2, 1, 4, 6}, {0, 8, 4, 16, 24}, {0, 12, 6, 24, 36}}
```

The following illustrates how an operation makes sense on the right cosets of {0, 4} in $\mathbb{Z}_8$. This also shows a quotient group.

```
gr1 = RightCosets[Z[8], {0, 4}, Mode → Visual, Output → Graphics];
```

Z[8]                                        x + y

| x \ y | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 |
| 4 | 4 | 0 | 5 | 1 | 6 | 2 | 7 | 3 |
| 1 | 1 | 5 | 2 | 6 | 3 | 7 | 4 | 0 |
| 5 | 5 | 1 | 6 | 2 | 7 | 3 | 0 | 4 |
| 2 | 2 | 6 | 3 | 7 | 4 | 0 | 5 | 1 |
| 6 | 6 | 2 | 7 | 3 | 0 | 4 | 1 | 5 |
| 3 | 3 | 7 | 4 | 0 | 5 | 1 | 6 | 2 |
| 7 | 7 | 3 | 0 | 4 | 1 | 5 | 2 | 6 |

By specifying Output → Graphics, we indicate that we want the graphic as the output, not the actual Cayley table.

```
gr2 = CayleyTable[Z[4], Mode → Visual, Output → Graphics];
```

For each element, a different color is used. The entries in the table corresponding to the elements are then colored and labeled accordingly.

Z[4] x + y

| x \ y | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 0 | 0 | 1 | 2 | 3 |
| 1 | 1 | 2 | 3 | 0 |
| 2 | 2 | 3 | 0 | 1 |
| 3 | 3 | 0 | 1 | 2 |

Putting the two side-by-side makes it clear to what group this quotient group $\mathbb{Z}_8 / \{0, 4\}$ is isomorphic.

```
Show[GraphicsArray[{gr1, gr2}]];
```

Z[8] x + y

| x \ y | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 |
| 4 | 4 | 0 | 5 | 1 | 6 | 2 | 7 | 3 |
| 1 | 1 | 5 | 2 | 6 | 3 | 7 | 4 | 0 |
| 5 | 5 | 1 | 6 | 2 | 7 | 3 | 0 | 4 |
| 2 | 2 | 6 | 3 | 7 | 4 | 0 | 5 | 1 |
| 6 | 6 | 2 | 7 | 3 | 0 | 4 | 1 | 5 |
| 3 | 3 | 7 | 4 | 0 | 5 | 1 | 6 | 2 |
| 7 | 7 | 3 | 0 | 4 | 1 | 5 | 2 | 6 |

Z[4] x + y

| x \ y | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 0 | 0 | 1 | 2 | 3 |
| 1 | 1 | 2 | 3 | 0 |
| 2 | 2 | 3 | 0 | 1 |
| 3 | 3 | 0 | 1 | 2 |

The following indicates that $\langle\{3, 2, 1\}\rangle$ in $S_3$ is not normal.

```
NormalQ[H = SubgroupGenerated[Symmetric[3], {3, 2, 1}], Symmetric[3]]
```

```
False
```

Because of this lack of normality, the product of cosets is not a well-defined operation, as illustrated here by the failure of having square blocks for products.

```
LeftCosets[Symmetric[3], H, Mode → Visual];
```

```
KEY for S[3]: label used → element: {g1 → {3, 2, 1}, g2 → {1, 2, 3},
  g3 → {2, 3, 1}, g4 → {1, 3, 2}, g5 → {3, 1, 2}, g6 → {2, 1, 3}}
```

S[3] x * y

| x \ y | g1 | g2 | g3 | g4 | g5 | g6 |
|---|---|---|---|---|---|---|
| g1 | g2 | g1 | g6 | g5 | g4 | g3 |
| g2 | g1 | g2 | g3 | g4 | g5 | g6 |
| g3 | g4 | g3 | g5 | g6 | g2 | g1 |
| g4 | g3 | g4 | g1 | g2 | g6 | g5 |
| g5 | g6 | g5 | g2 | g1 | g3 | g4 |
| g6 | g5 | g6 | g4 | g3 | g1 | g2 |

# Ringoids

Since we now wish to consider rings, we switch our structure.

```
SwitchStructureTo[Ring]
```

```
Ring
```

`FormRingoid` works in a fashion analogous to `FormGroupoid`. The required parameters are the list of elements, the addition operation and the multiplication operation. Options can be added afterwards.

```
R = FormRingoid[{0, 2, 1, 4, 6}, Plus, Times, FormatElements → True,
FormatOperator → False]
```

```
Ringoid[{-Elements-}, Plus, Times]
```

`RingQ` is similar to `GroupQ`; upon the first failure, it returns `False`.

```
RingQ[R]
```

```
False
```

Since there are two operations, we need to view the Cayley tables of both operations.

```
CayleyTables[R, Mode → Visual]
```

Add. x + y

| x \ y | 0 | 2 | 1 | 4 | 6 |
|---|---|---|---|---|---|
| 0 | 0 | 2 | 1 | 4 | 6 |
| 2 | 2 | 4 | 3 | 6 | 8 |
| 1 | 1 | 3 | 2 | 5 | 7 |
| 4 | 4 | 6 | 5 | 8 | 10 |
| 6 | 6 | 8 | 7 | 10 | 12 |

Mult. x * y

| x \ y | 0 | 2 | 1 | 4 | 6 |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 4 | 2 | 8 | 12 |
| 1 | 0 | 2 | 1 | 4 | 6 |
| 4 | 0 | 8 | 4 | 16 | 24 |
| 6 | 0 | 12 | 6 | 24 | 36 |

```
{{{0, 2, 1, 4, 6}, {2, 4, 3, 6, 8}, {1, 3, 2, 5, 7},
  {4, 6, 5, 8, 10}, {6, 8, 7, 10, 12}}, {{0, 0, 0, 0, 0}, {0, 4, 2, 8, 12},
  {0, 2, 1, 4, 6}, {0, 8, 4, 16, 24}, {0, 12, 6, 24, 36}}}
```

Here we form the extension ring of polynomials over the Boolean ring on {1, 2, 3} (whose elements are subsets of the powerset of {1,2,3} with `Addition` being the symmetric difference and `Multiplication` being interesection) and choose a random polynomial of degree 2 that is monic (leading coefficient is the unity).

```
RandomElement[PolynomialsOver[BooleanRing[3]], 2, Monic → True]
```

$\{1, 2, 3\} + \{1, 2\}\, x + \{1, 2, 3\}\, x^2$

```
FullForm[%]
```

```
AbstractAlgebra`Master`Private`poly[List[
  AbstractAlgebra`Master`Private`ringoid[List[List[], List[3], List[2],
    List[2, 3], List[1], List[1, 3], List[1, 2], List[1, 2, 3]], Function[
    Union[Complement[Slot[1], Slot[2]], Complement[Slot[2], Slot[1]]]],
   Function[Intersection[Slot[1], Slot[2]]],
   List[List[], List[], List[], List[], List[],
    List[Rule[RingoidName, "Bool[3]"], Rule[RingoidDescription,
      "the boolean Ring on {1,...,3}"], Rule[WideElements, True]]]],
  LeftToRight, x, False], List[List[1, 2, 3], List[1, 2], List[1, 2, 3]]]
```

Next we consider a random 3 by 3 matrix whose elements come from the lattice ring on the divisors of 12 (with operation LCM/GCD for the addition and GCD for the multiplication.

```
RandomElement[MatricesOver[LatticeRing[12], 3]] // MatrixForm
```

$$\begin{pmatrix} 3 & 12 & 4 \\ 12 & 4 & 1 \\ 6 & 12 & 12 \end{pmatrix}$$

The third type of ring extension is the ring of functions over a ring; here we use $\mathbb{Z}_{12}$.

```
RandomElement[FunctionsOver[ZR[12]]]
```

```
Func[1, 11, 9, 3, 9, 5, 2, 10, 9, 2, 1, 4]
```

As a last example here, we form the Galois field of order 9.

```
GF[9]
```

```
Ringoid[{0, x, 2 x, 1, 1 + x, 1 + 2 x, 2, 2 + x, 2 + 2 x},
 -Addition-, -Multiplication-]
```

# Morphoids

To form a `Morphoid`, the parameters are a (pure) function and then either two groupoids or two ringoids. (The function has the first structure as the domain and the second as the codomain.)

```
f = FormMorphoid[Mod[#, 6]&, Z[12], Z[6]]
```

```
Morphoid[Mod[#1, 6] &, -Z[12]-, -Z[6]-]
```

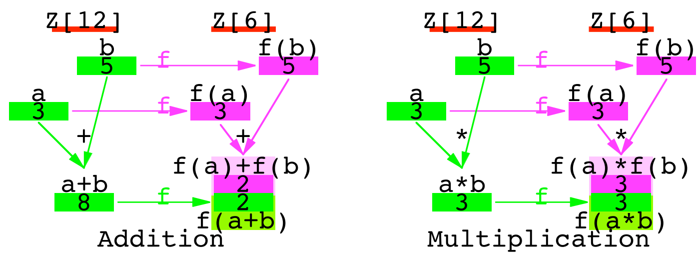
The `MorphismQ` function determines if this is a (ring) homomorphism.

```
MorphismQ[f]
```

```
True
```

To see visually why the operations are preserved for the pair (3, 5), try the following.

```
PreservesQ[f, {3, 5}, Mode → Visual]
```




```
True
```

We now switch back to groups.

```
SwitchStructureTo[Group]
```

```
Group
```

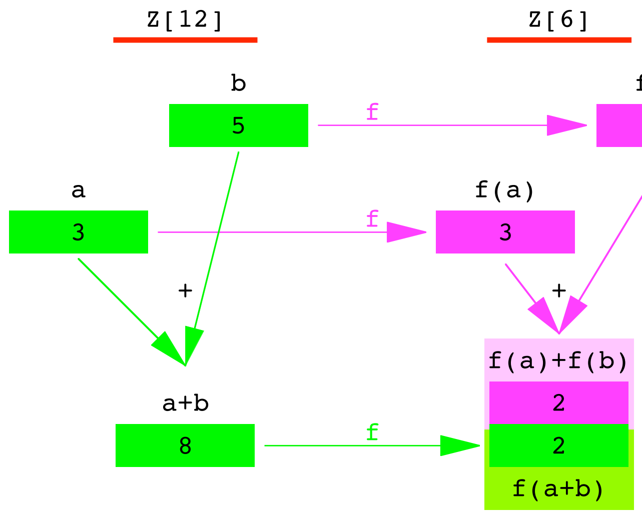
At this point, we now build a group homomorphism (with the same definition).

```
g = FormMorphoid[Mod[#, 6]&, Z[12], Z[6]]
```

```
Morphoid[Mod[#1, 6] &, -Z[12]-, -Z[6]-]
```

We see different results now that we are working with groups.

```
PreservesQ[g, {3, 5}, Mode → Visual]
```




```
True
```

Sometimes morphisms are more easily set up by matching how we want the elements to line up.

```
FormMorphoidSetup[D[4], Z[8]];
```

| Domain | | | Codomain |
|---|---|---|---|
| 1 | 1 | 1 | 0 |
| Rot | 2 | 2 | 1 |
| Rot^2 | 3 | 3 | 2 |
| Rot^3 | 4 | 4 | 3 |
| Ref | 5 | 5 | 4 |
| Rot**Ref | 6 | 6 | 5 |
| Rot^2**Ref | 7 | 7 | 6 |
| Rot^3**Ref | 8 | 8 | 7 |

Suppose that we want to send the first element in the domain to the first element in the codomain, the second element in the domain to the third element in the codomain, the third to the fifth and so on.

```
h = FormMorphoid[{1, 3, 5, 7, 2, 4, 6, 8}, D[4], Z[8]]
```

```
Morphoid[{1 → 0, Rot → 2, Rot^2 → 4, Rot^3 → 6, Ref → 1,
  Rot ** Ref → 3, Rot^2 ** Ref → 5, Rot^3 ** Ref → 7}, -D[4]-, -Z[8]-]
```

Here we see that this is not a homomorphism on the whole group, but note that we can see a homormorphism from the rotational subgroup to the set {0, 2, 4, 6}.

```
MorphismQ[h, Mode → Visual]
```

```
The table entry corresponding to the computation a∗b in the
  domain of the morphoid is colored if and only if the pair
  {a,b} is preserved by the morphoid; i.e., f(a∗b) = f(a)∗f(b)

KEY for D[4]: label used → element: {g1 → 1, g2 → Rot, g3 → Rot^2, g4 →
  Rot^3, g5 → Ref, g6 → Rot**Ref, g7 → Rot^2**Ref, g8 → Rot^3**Ref}
```

D[4] x ⋆ y

| x \ y | g1 | g2 | g3 | g4 | g5 | g6 | g7 | g8 |
|---|---|---|---|---|---|---|---|---|
| g1 | g1 | g2 | g3 | g4 | g5 | g6 | g7 | g8 |
| g2 | g2 | g3 | g4 | g1 | g6 | g7 | g8 | g5 |
| g3 | g3 | g4 | g1 | g2 | g7 | g8 | g5 | g6 |
| g4 | g4 | g1 | g2 | g3 | g8 | g5 | g6 | g7 |
| g5 | g5 | g8 | g7 | g6 | g1 | g4 | g3 | g2 |
| g6 | g6 | g5 | g8 | g7 | g2 | g1 | g4 | g3 |
| g7 | g7 | g6 | g5 | g8 | g3 | g2 | g1 | g4 |
| g8 | g8 | g7 | g6 | g5 | g4 | g3 | g2 | g1 |

```
False
```

# Help Browser

We have implemented full documentation into the Help Browser. Before using, you need to download and install from http://www.central.edu/eaam/, choose *Rebuild Help Index* from the *Help* menu and then access it from the `AddOns` button.

# Exploring Abstract Algebra with Mathematica

## Overview

The packages in *AbstractAlgebra* form the foundation for a series of 14 group labs and 13 ring labs designed to help students conceptualize abstract algebra. These are combined with documentation for *AbstractAlgebra* in a book entitled *Exploring Abstract Algebra with Mathematica* (EAAM) published by TELOS/Springer-Verlag (January, 1999—ISBN 0-387-98619-7).

## Group Labs

Group Lab 1. *Using Symmetry to Uncover a Group*—This lab explores the underlying definition of a group by looking at the symmetries of an equilateral triangle.

Group Lab 2. *Determining the Symmetry Group of a Given Figure*—The focus of this lab is to determine the symmetry group of a figure chosen randomly from a list of regular polygons and "cyclic" objects.

Group Lab 3. *Is This a Group?*—This lab randomly presents a Cayley table of one of 20 "possible groups." The goal is to determine which of the defining properties of a group are reflected in the Cayley table to determine which are groups.

Group Lab 4. *Let's Get These Orders Straight*—This lab looks at the order of an element and its inverse, the distribution of the orders of the elements in $\mathbb{Z}_n$, investigates the probability that an element in $\mathbb{Z}_n$ has order $n$ and also explores the group $U_n$ (the units in $\mathbb{Z}_n$).

Group Lab 5. *Subversively Grouping Our Elements*—This lab explores the notion of a subgroup, including looking at the subgroups of $\mathbb{Z}_n$ and $U_n$, calculating the probability that a random subset of $\mathbb{Z}_n$ is a subgroup, and determining what elements in a subset are necessary so that the closure yields the whole group.

Group Lab 6. *Cycling Through the Groups*—Here we focus on the notion of a cyclic group and its subgroup structure. We also look at determining when the direct sum of $\mathbb{Z}_m$ and $\mathbb{Z}_n$ is a cyclic group.

Group Lab 7. *Permutations*—This lab looks at the definitions of a permutation, cycle and transposition, how to perform computations with each of these structures, and explores their properties and relations to each other. We also look at some applications of permutations.

Group Lab 8. *Isomorphisms*—Here we look at the definition of an isomorphism and then use various visual mechanisms to try to determine when two groups are or are not isomorphic. A `Morphoid` is introduced here.

Group Lab 9. *Automorphisms*—In this lab, we look at the group of automorphisms of $\mathbb{Z}_n$ and also look at inner automorphisms.

Group Lab 10. *Direct Products*—The notion of direct products (sums) are introduced and we determine the order of elements in a direct product. We also try to determine when the direct product of cyclic groups is itself cyclic. We also look for isomorphisms between some $U_n$ groups.

Group Lab 11. *Cosets*—This lab explores the definition and properties of cosets.

Group Lab 12. *Normality and Factor groups*—A normal group is defined, explored, and then used to define and explore factor groups.

Group Lab 13. *Homomorphisms*—This lab explores group homomorphisms.

Group Lab 14. *Rotational Groups of Regular Polyhedra*—Here we look at how to generate the rotational groups of several polyhedra.

# Ring Labs

Ring Lab 1. *An Introduction to Rings and* `Ringoids`—This lab introduces some of the definitions and properties of rings.

Ring Lab 2. *An Introduction to Rings, Part 2*—This lab continues looking at basic ring concepts, focusing on units and zero divisors and leading to the definition of an integral domain and field.

Ring Lab 3. *An Ideal Part of Rings*—This explores the notion of an ideal and properties related to it.

Ring Lab 4. *What Does* $\mathbb{Z}[i]/\langle a + b\,i \rangle$ *Look Like?*—This lab focuses on the Gaussian integers modulo an ideal generated by some Gaussian integer.

Ring Lab 5. *Ring Homomorphisms*—This lab looks at the definition of a ring homomorphism, the First Isomorphism Theorem, and the Chinese Remainder Theorem.

Ring Lab 6. *Polynomial Rings*—Some basic properties of polynomial rings are introduced and explored.

Ring Lab 7. *Factoring and Irreducibility*—What does it mean to factor a polynomial? Various definitions and techniques are introduced, including the Rational Root Theorem, the Mod $p$ Irreducibility Test, and Eisenstein's Criterion.

Ring Lab 8. *Roots of Unity*—This lab focuses on the polynomial $x^n - 1$ and explores graphically the zeros of this polynomial, in particular seeing how the zeros are related to the factors and how the group $U_n$ arises out of this.

Ring Lab 9. *Cyclotomic Polynomials*—This lab focuses on cyclotomic polynomials and the many properties related to them.

Ring Lab 10. *Quotient Rings of Polynomials*—The notion of a quotient ring over a polynomial is introduced in this lab.

Ring Lab 11. *Quadratic Field Extensions*—This lab continues the last one by looking more closely at quotient rings modulo a quadratic polynomial where the result is a field.

Ring Lab 12. *Factoring in* $\mathbb{Z}\left[\sqrt{d}\right]$—This lab focuses on the rings $\mathbb{Z}\left[\sqrt{d}\right]$ and pursues the notion of divisibility and factoring in such rings. Several rings are illustrated as failing being a UFD.

Ring Lab 13. *Finite Fields*—This lab continues the ideas formulated in lab 11 by looking at Galois fields and properties related to them.

# GroupCalculator

See our web page for a group calculator to download it. (For now, start with a clean kernel, clearing out any previous *AbstractAlgebra* definitions.) By `clicking here`, one will go to the Help Browser, from which a click there will bring up a workable calculator.

# More Groupoids

We mostly work with groups here.

```
SwitchStructureTo[Group]
```

```
Group
```

There are a number of options for controlling how groupoids, ringoids and morphoids are formed.

```
Options[FormGroupoid]
```

```
{CayleyForm → OutputForm, FormatElements → False,
 FormatOperator → True, Generators → {}, GroupoidDescription → ,
 GroupoidName → TheGroup, IsAGroup → False, KeyForm → InputForm,
 MaxElementsToList → 50, WideElements → False}
```

We can form the permutation group on any set of elements.

```
H = PermutationGroup[{α, β, γ}]
```

```
Groupoid[{{α, β, γ}, {α, γ, β}, {β, α, γ}, {β, γ, α}, {γ, α, β}, {γ, β, α}},
 -Operation-]
```

Here is the Cayley table of the group just formed, using a Key since the elements are too wide for the table.

```
CayleyTable[H, Mode → Visual, KeyForm → StandardForm];
```

KEY for TheGroup: label used → element: {g1 → {α, β, γ}, g2 → {α, γ, β}, g3 → {β, α, γ}, g4 → {β, γ, α}, g5 → {γ, α, β}, g6 → {γ, β, α}}

TheGroup x * y

| x \ y | g1 | g2 | g3 | g4 | g5 | g6 |
|---|---|---|---|---|---|---|
| g1 | g1 | g2 | g3 | g4 | g5 | g6 |
| g2 | g2 | g1 | g5 | g6 | g3 | g4 |
| g3 | g3 | g4 | g1 | g2 | g6 | g5 |
| g4 | g4 | g3 | g6 | g5 | g1 | g2 |
| g5 | g5 | g6 | g2 | g1 | g4 | g3 |
| g6 | g6 | g5 | g4 | g3 | g2 | g1 |

We form a list of some groups, to be used below.

```
someGroups = {Z[5], Dihedral[4], Symmetric[3], U[15]}
```

```
{Groupoid[{0, 1, 2, 3, 4}, Mod[#1 + #2, 5] &], Groupoid[
  {1, Rot, Rot^2, Rot^3, Ref, Rot ** Ref, Rot^2 ** Ref, Rot^3 ** Ref}, -Operation-],
 Groupoid[{{1, 2, 3}, {1, 3, 2}, {2, 1, 3}, {2, 3, 1}, {3, 1, 2}, {3, 2, 1}},
  -Operation-], Groupoid[{1, 2, 4, 7, 8, 11, 13, 14}, Mod[#1 #2, 15] &]}
```

Most functions can take a list of arguments, as shown here with CayleyTable.

```
CayleyTable[someGroups, Mode → Visual];
```

KEY for D[4]: label used → element: {g1 → 1, g2 → Rot, g3 → Rot^2, g4 → Rot^3, g5 → Ref, g6 → Rot**Ref, g7 → Rot^2**Ref, g8 → Rot^3**Ref}

KEY for S[3]: label used → element: {g1 → {1, 2, 3}, g2 → {1, 3, 2}, g3 → {2, 1, 3}, g4 → {2, 3, 1}, g5 → {3, 1, 2}, g6 → {3, 2, 1}}

Z[5] x + y

| x \ y | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | 0 | 1 | 2 | 3 | 4 |
| 1 | 1 | 2 | 3 | 4 | 0 |
| 2 | 2 | 3 | 4 | 0 | 1 |
| 3 | 3 | 4 | 0 | 1 | 2 |
| 4 | 4 | 0 | 1 | 2 | 3 |

D[4] x * y

| x \ y | g1 | g2 | g3 | g4 | g5 | g6 | g7 | g8 |
|---|---|---|---|---|---|---|---|---|
| g1 | g1 | g2 | g3 | g4 | g5 | g6 | g7 | g8 |
| g2 | g2 | g3 | g4 | g1 | g6 | g7 | g8 | g5 |
| g3 | g3 | g4 | g1 | g2 | g7 | g8 | g5 | g6 |
| g4 | g4 | g1 | g2 | g3 | g8 | g5 | g6 | g7 |
| g5 | g5 | g8 | g7 | g6 | g1 | g4 | g3 | g2 |
| g6 | g6 | g5 | g8 | g7 | g2 | g1 | g4 | g3 |
| g7 | g7 | g6 | g5 | g8 | g3 | g2 | g1 | g4 |
| g8 | g8 | g7 | g6 | g5 | g4 | g3 | g2 | g1 |

S[3] x * y

| x \ y | g1 | g2 | g3 | g4 | g5 | g6 |
|---|---|---|---|---|---|---|
| g1 | g1 | g2 | g3 | g4 | g5 | g6 |
| g2 | g2 | g1 | g5 | g6 | g3 | g4 |
| g3 | g3 | g4 | g1 | g2 | g6 | g5 |
| g4 | g4 | g3 | g6 | g5 | g1 | g2 |
| g5 | g5 | g6 | g2 | g1 | g4 | g3 |
| g6 | g6 | g5 | g4 | g3 | g2 | g1 |

U[15] x * y

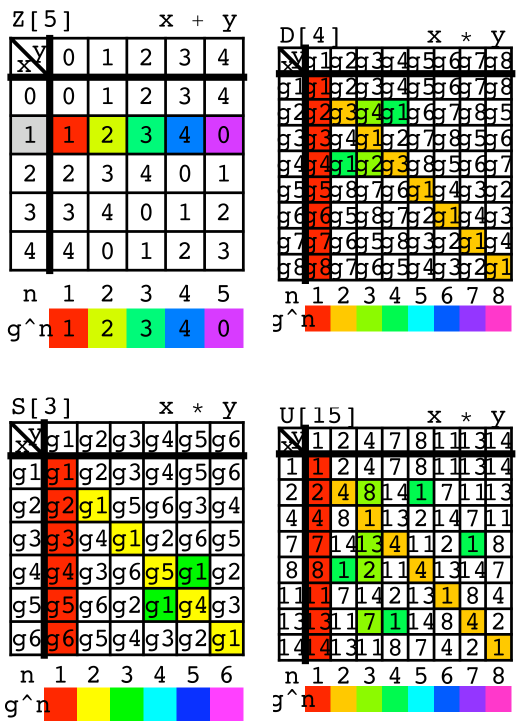
| x \ y | 1 | 2 | 4 | 7 | 8 | 11 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 2 | 4 | 7 | 8 | 11 | 13 | 14 |
| 2 | 2 | 4 | 8 | 14 | 1 | 7 | 11 | 13 |
| 4 | 4 | 8 | 1 | 13 | 2 | 14 | 7 | 11 |
| 7 | 7 | 14 | 13 | 4 | 11 | 2 | 1 | 8 |
| 8 | 8 | 1 | 2 | 11 | 4 | 13 | 14 | 7 |
| 11 | 11 | 7 | 14 | 2 | 13 | 1 | 8 | 4 |
| 13 | 13 | 11 | 7 | 1 | 14 | 8 | 4 | 2 |
| 14 | 14 | 13 | 11 | 8 | 7 | 4 | 2 | 1 |

Here is a visualization of why the following groups are or are not cyclic.

```
CyclicQ[someGroups, Mode → Visual]
```

KEY for D[4]: label used → element: {g1 → 1, g2 → Rot, g3 → Rot^2, g4 → Rot^3, g5 → Ref, g6 → Rot**Ref, g7 → Rot^2**Ref, g8 → Rot^3**Ref}

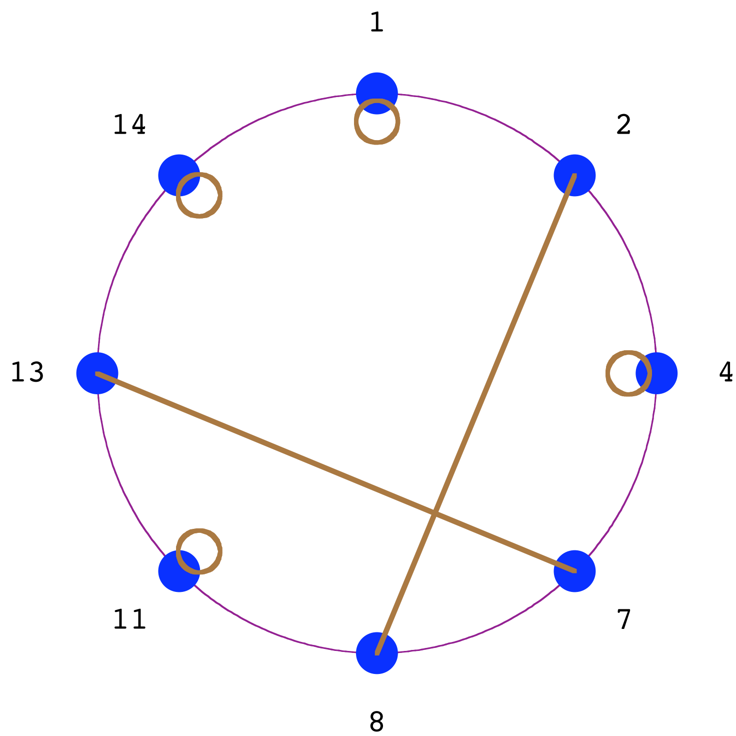
KEY for S[3]: label used → element: {g1 → {1, 2, 3}, g2 → {1, 3, 2}, g3 → {2, 1, 3}, g4 → {2, 3, 1}, g5 → {3, 1, 2}, g6 → {3, 2, 1}}




```
{True, False, False, False}
```

Loops indicate self-inversive elements, while lines connect other inverses.

```
Inverses[U[15], Mode → Visual]
```




```
{{1, 1}, {2, 8}, {4, 4}, {7, 13}, {11, 11}, {14, 14}}
```

We can form the direct product of any number of groupoids.

```
G = DirectProduct[Z[5], U[4]]
```

```
Groupoid[{{0, 1}, {0, 3}, {1, 1}, {1, 3}, {2, 1},
  {2, 3}, {3, 1}, {3, 3}, {4, 1}, {4, 3}}, -Operation-]
```

Here we choose 2 random elements from this group, each of which are pairs.

```
{g, h} = RandomElements[G, 2]
```

```
{{4, 1}, {0, 1}}
```

We can apply the group operation to these elements as follows.

```
Operation[G][g,h]
```

```
{4, 1}
```

Here is a nonsense groupoid formed by specifying the "group" table.

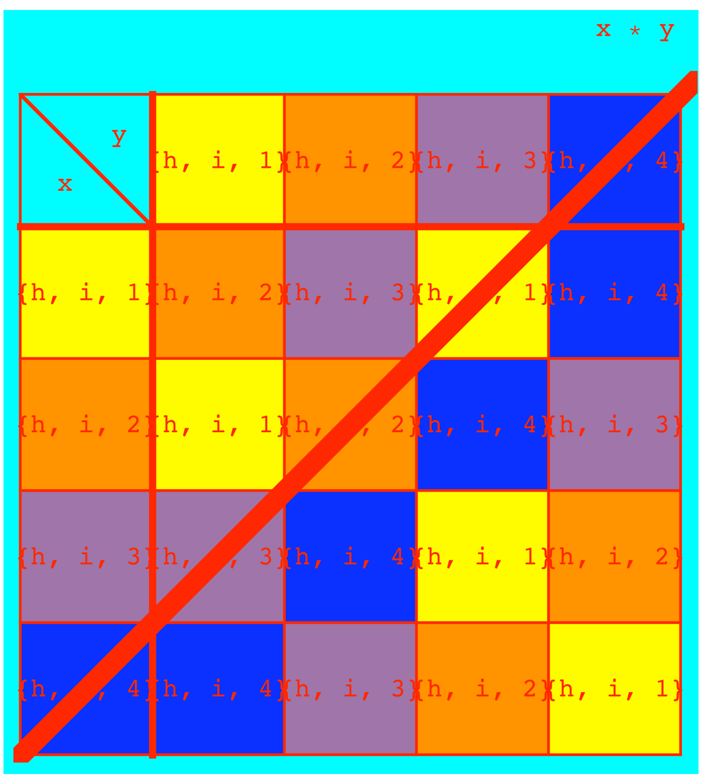
```
H = FormGroupoidByTable[{b, a, a ** b, a^b},
  {{a, a ** b, b, a^b}, {b, a, a^b, a ** b}, {a ** b, a^b, b, a}, {a^b, a ** b, a, b}},
  "*", WideElements → True]
```

```
Groupoid[{b, a, a ** b, a^b}, -Operation-]
```

The CayleyTable function has a large number of options, as well as the ability to take Graphics options.

```
CayleyTable[H, Mode → Visual, ShowName → False, VarToUse → "hi",
 KeyForm → FullForm, Background → Cyan, CayleyForm → Characters,
 Epilog → {RGBColor[1, 0, 0], Thickness[0.02], Line[{{-1, 0}, {5, 6}}]}]
```

```
KEY for TheGroup: label used → element: {hi1 → b, hi2 →
  a, hi3 → NonCommutativeMultiply[a, b], hi4 → Power[a, b]}
```

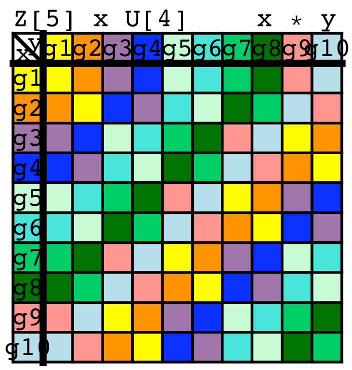



```
{{a, a ** b, b, a^b}, {b, a, a^b, a ** b}, {a ** b, a^b, b, a}, {a^b, a ** b, a, b}}
```

Each groupoid in CayleyTable can receive different options.

```
CayleyTable[{G, H}, {{ShowBodyText → False}, {ShowKey → False}},
  Mode → Visual];
```

```
KEY for Z[5] x U[4]: label used → element: {g1 → {0, 1},
  g2 → {0, 3}, g3 → {1, 1}, g4 → {1, 3}, g5 → {2, 1}, g6 →
  {2, 3}, g7 → {3, 1}, g8 → {3, 3}, g9 → {4, 1}, g10 → {4, 3}}
```




TheGroup x * y

| x \ y | g1 | g2 | g3 | g4 |
|---|---|---|---|---|
| g1 | g2 | g3 | g1 | g4 |
| g2 | g1 | g2 | g4 | g3 |
| g3 | g3 | g4 | g1 | g2 |
| g4 | g4 | g3 | g2 | g1 |

We can work with Gaussian integers reduced some modulus.

```
Z[4, I]
```

```
Groupoid[{0, i, 2 i, 3 i, 1, 1 + i, 1 + 2 i, 1 + 3 i, 2,
  2 + i, 2 + 2 i, 2 + 3 i, 3, 3 + i, 3 + 2 i, 3 + 3 i}, -Operation-]
```

The TwistedZ is an interesting groupoid that is sometimes a group.

```
SubgroupQ[ {0, 2 , 8}, TwistedZ[13]]
```

```
True
```

The SubgroupQ function takes multiple requests in the following fashion.

```
SubgroupQ[{{ {0, 3}, Z[5]}, { {1, 4}, U[9]}}, Mode → Visual]
```

All the elements marked with yellow are original elements in the set. Those in red are from outside.

Z[5] x + y

| x\y | 0 | 3 | 1 | 2 | 4 |
|---|---|---|---|---|---|
| 0 | 0 | 3 | 1 | 2 | 4 |
| 3 | 3 | 1 | 4 | 0 | 2 |
| 1 | 1 | 4 | 2 | 3 | 0 |
| 2 | 2 | 0 | 3 | 4 | 1 |
| 4 | 4 | 2 | 0 | 1 | 3 |

U[9] x * y

| x\y | 1 | 4 | 2 | 5 | 7 | 8 |
|---|---|---|---|---|---|---|
| 1 | 1 | 4 | 2 | 5 | 7 | 8 |
| 4 | 4 | 7 | 8 | 2 | 1 | 5 |
| 2 | 2 | 8 | 4 | 1 | 5 | 7 |
| 5 | 5 | 2 | 1 | 7 | 8 | 4 |
| 7 | 7 | 1 | 5 | 8 | 4 | 2 |
| 8 | 8 | 5 | 7 | 4 | 2 | 1 |

```
{False, False}
```

Given the set {1,4} of the group $\mathbb{Z}_9$, the following shows how the closure of this set is built up in three iterations.

```
Closure[Z[9], {1, 4}, ReportIterations → True]
```

```
{Groupoid[{1, 4, 2, 5, 8, 3, 6, 0, 7}, Mod[#1 + #2, 9] &],
 {3, {{1, 4}, {1, 4, 2, 5, 8}, {1, 4, 2, 5, 8, 3, 6, 0, 7}}}}
```

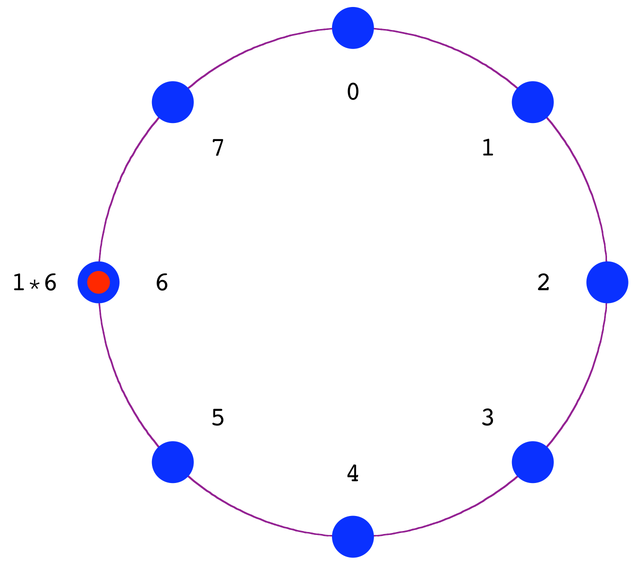
One may want the elements to be canonically sorted.

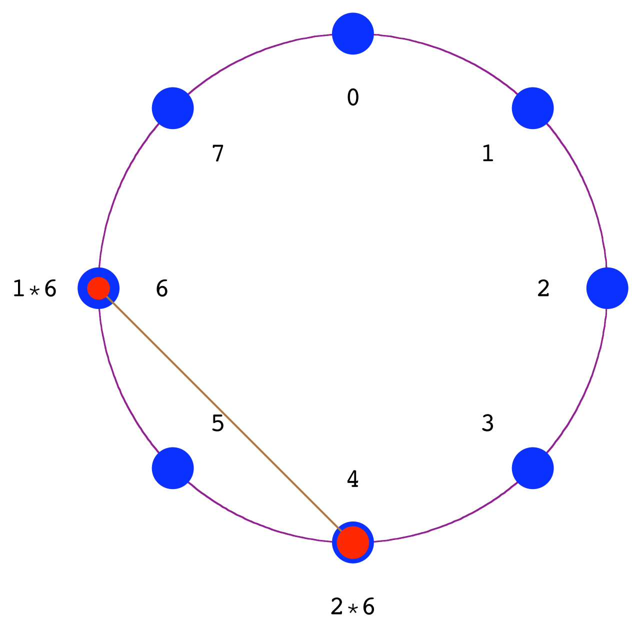
```
Closure[Z[9], {1, 4}, Sort → True]
```

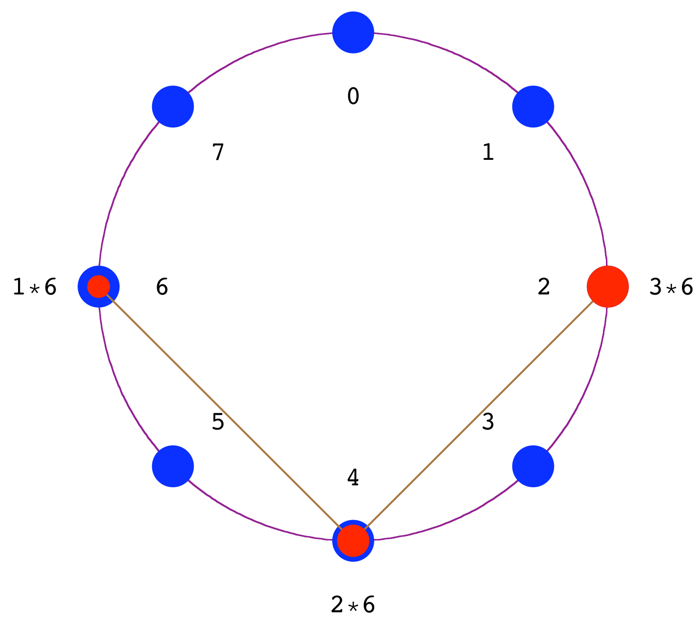
```
Groupoid[{0, 1, 2, 3, 4, 5, 6, 7, 8}, Mod[#1 + #2, 9] &]
```

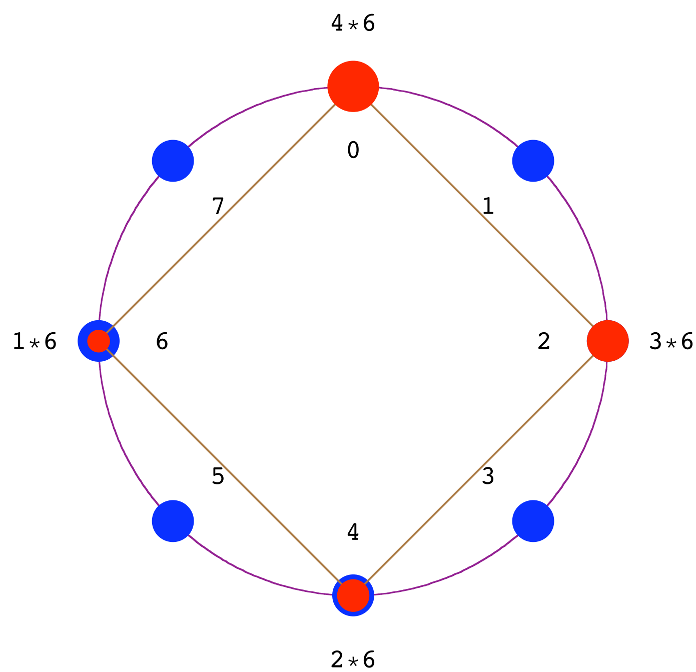
Here is a animation indicating the subgroup generated by 6 in the group $\mathbb{Z}_8$.

```
SubgroupGenerated[Z[8], 6, Mode → Visual]
```


0
1
2
3
4
5
6
7
1 * 6


0
1
2
3
4
5
6
7
1 * 6
2 * 6

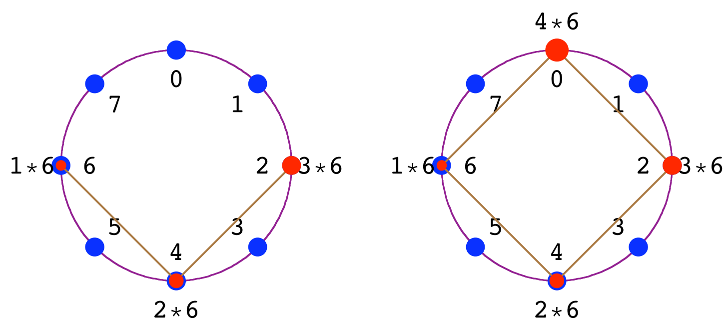



```
Groupoid[{6, 4, 2, 0}, Mod[#1 + #2, 8] &]
```

Here is the same but using a GraphicsArray for its display.

```
SubgroupGenerated[Z[8], 6, Mode → Visual, Output → GraphicsArray]
```



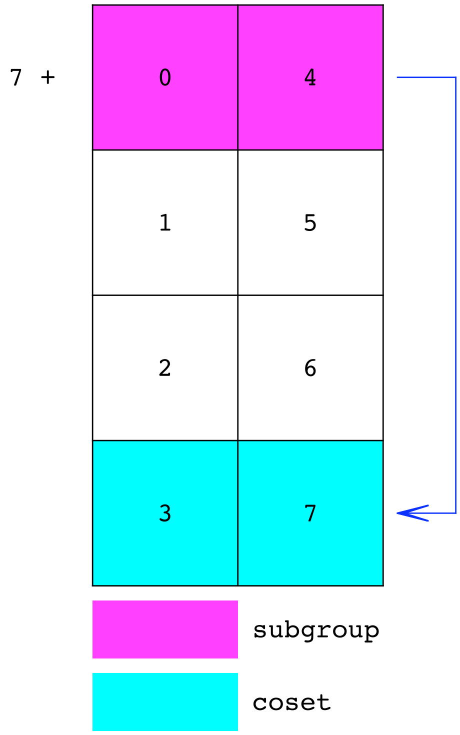

We can find all cyclic subgroups of any group.

```
CyclicSubgroups[D[4]]
```

```
{Groupoid[{1}, -Operation-], Groupoid[{1, Ref}, -Operation-],
 Groupoid[{1, Rot^2}, -Operation-], Groupoid[{1, Rot ** Ref}, -Operation-],
 Groupoid[{1, Rot^2 ** Ref}, -Operation-],
 Groupoid[{1, Rot^3 ** Ref}, -Operation-],
 Groupoid[{1, Rot, Rot^2, Rot^3}, -Operation-]}
```

Here is a visualization showing the left coset 7 + {0, 4} in the group $\mathbb{Z}_8$.

```
LeftCoset[Z[8], {0, 4}, 7, Mode → Visual]
```




```
{7, 3}
```

This illustrates how an operation makes sense on the following right cosets. This also shows a quotient group.

```
gr1 = RightCosets[Z[8], {0, 4}, Mode → Visual, Output → Graphics];
```

Z[8] x + y

| x \ y | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 |
| 4 | 4 | 0 | 5 | 1 | 6 | 2 | 7 | 3 |
| 1 | 1 | 5 | 2 | 6 | 3 | 7 | 4 | 0 |
| 5 | 5 | 1 | 6 | 2 | 7 | 3 | 0 | 4 |
| 2 | 2 | 6 | 3 | 7 | 4 | 0 | 5 | 1 |
| 6 | 6 | 2 | 7 | 3 | 0 | 4 | 1 | 5 |
| 3 | 3 | 7 | 4 | 0 | 5 | 1 | 6 | 2 |
| 7 | 7 | 3 | 0 | 4 | 1 | 5 | 2 | 6 |

By specifying Output → Graphics, we indicate that we want the graphic as the output, not the actual Cayley table.

```
gr2 = CayleyTable[Z[4], Mode → Visual, Output → Graphics];
```

Z[4] x + y

| x \ y | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 0 | 0 | 1 | 2 | 3 |
| 1 | 1 | 2 | 3 | 0 |
| 2 | 2 | 3 | 0 | 1 |
| 3 | 3 | 0 | 1 | 2 |

Putting the two side-by-side makes it clear to what group this quotient group $\mathbb{Z}_8 / \{0, 4\}$ is isomorphic.

```
Show[GraphicsArray[{gr1, gr2}]];
```

Z[8] x + y

| x \ y | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 |
| 4 | 4 | 0 | 5 | 1 | 6 | 2 | 7 | 3 |
| 1 | 1 | 5 | 2 | 6 | 3 | 7 | 4 | 0 |
| 5 | 5 | 1 | 6 | 2 | 7 | 3 | 0 | 4 |
| 2 | 2 | 6 | 3 | 7 | 4 | 0 | 5 | 1 |
| 6 | 6 | 2 | 7 | 3 | 0 | 4 | 1 | 5 |
| 3 | 3 | 7 | 4 | 0 | 5 | 1 | 6 | 2 |
| 7 | 7 | 3 | 0 | 4 | 1 | 5 | 2 | 6 |

Z[4] x + y

| x \ y | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 0 | 0 | 1 | 2 | 3 |
| 1 | 1 | 2 | 3 | 0 |
| 2 | 2 | 3 | 0 | 1 |
| 3 | 3 | 0 | 1 | 2 |

The following indicates that ⟨{3, 2, 1}⟩ in $S_3$ is not normal.

```
NormalQ[H = SubgroupGenerated[Symmetric[3], {3, 2, 1}], Symmetric[3]]
```

```
False
```

Because of this lack of normality, the product of cosets is not a well-defined operation, as illustrated here by the failure of having square blocks for products.

```
LeftCosets[Symmetric[3], H, Mode → Visual];
```

```
KEY for S[3]: label used → element: {g1 → {3, 2, 1}, g2 → {1, 2, 3},
  g3 → {2, 3, 1}, g4 → {1, 3, 2}, g5 → {3, 1, 2}, g6 → {2, 1, 3}}
```

S[3] x * y

| x \ y | g1 | g2 | g3 | g4 | g5 | g6 |
|---|---|---|---|---|---|---|
| g1 | g2 | g1 | g6 | g5 | g4 | g3 |
| g2 | g1 | g2 | g3 | g4 | g5 | g6 |
| g3 | g4 | g3 | g5 | g6 | g2 | g1 |
| g4 | g3 | g4 | g1 | g2 | g6 | g5 |
| g5 | g6 | g5 | g2 | g1 | g3 | g4 |
| g6 | g5 | g6 | g4 | g3 | g1 | g2 |

Since {0, 4} is normal in $\mathbb{Z}_8$, we can form the quotient group.

```
QuotientGroup[Z[8], {0, 4}]
```

```
QuotientGroup::NS :
 This quotient group uses NS to represent the normal subgroup
   {0, 4} that you specified. Use CosetToList to convert
   this coset representation to a list of elements.
```

```
Groupoid[{NS, 1 + NS, 2 + NS, 3 + NS}, -Operation-]
```

Here is a Cayley table of this group, using a different form and set of representatives for the representation of the elements.

```
CayleyTable[QuotientGroup[Z[8], {0, 4}, Form → Representatives,
  Representatives → {4, 1, 6, 3}], Mode → Visual]
```

Z[8]/NS x + y

| x \ y | 4 | 1 | 6 | 3 |
|---|---|---|---|---|
| 4 | 4 | 1 | 6 | 3 |
| 1 | 1 | 6 | 3 | 4 |
| 6 | 6 | 3 | 4 | 1 |
| 3 | 3 | 4 | 1 | 6 |

```
{{4, 1, 6, 3}, {1, 6, 3, 4}, {6, 3, 4, 1}, {3, 4, 1, 6}}
```

The same group is shown here using a coset list for each element.

```
CayleyTable[QuotientGroup[Z[8], {0, 4}, Form → CosetLists],
 Mode → Visual]
```

```
KEY for Z[8]/NS: label used → element:
  {g1 → {0, 4}, g2 → {1, 5}, g3 → {2, 6}, g4 → {3, 7}}
```

Z[8]/NS x + y

| x \ y | g1 | g2 | g3 | g4 |
|---|---|---|---|---|
| g1 | g1 | g2 | g3 | g4 |
| g2 | g2 | g3 | g4 | g1 |
| g3 | g3 | g4 | g1 | g2 |
| g4 | g4 | g1 | g2 | g3 |

```
{{{0, 4}, {1, 5}, {2, 6}, {3, 7}}, {{1, 5}, {2, 6}, {3, 7}, {0, 4}},
 {{2, 6}, {3, 7}, {0, 4}, {1, 5}}, {{3, 7}, {0, 4}, {1, 5}, {2, 6}}}
```

This visulalization shows that 4 is the group exponent for the group $U_{15}$.

```
GroupExponent[U[15], Mode → Visual]
```

| powers \ elements | 1 | 2 | 4 | 7 | 8 | 11 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|
| 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 1 | 8 | 4 | 13 | 2 | 11 | 7 | 14 |
| 2 | 1 | 4 | 1 | 4 | 4 | 1 | 4 | 1 |
| 1 | 1 | 2 | 4 | 7 | 8 | 11 | 13 | 14 |

```
4
```

GenerateGroupoid is another means of forming a groupoid.

```
G = GenerateGroupoid[{{{2, 1}, {1, 1}}}, Mod[#1.#2, 3] &,
  WideElements -> True]
```

```
Groupoid[{{{1, 0}, {0, 1}}, {{1, 2}, {2, 2}},
  {{2, 0}, {0, 2}}, {{2, 1}, {1, 1}}}, -Operation-]
```

# More Ringoids

# More Morphoids

# And other stuff

# Not satisfied? Some More!