A String of Pearls: Proofs of Fermat's Little Theorem

Hing-Lun Chan[1] and Michael Norrish[2]

[1] joseph.chan@anu.edu.au
Australian National University
[2] Michael.Norrish@nicta.com.au
Canberra Research Lab., NICTA*;
also, Australian National University

Abstract. We discuss mechanised proofs of Fermat's Little Theorem in a variety of styles, focusing in particular on an elegant combinatorial "necklace" proof that has not been mechanised previously. What is elegant in prose turns out to be long-winded mechanically, and so we examine the effect of explicitly appealing to group theory. This has pleasant consequences both for the necklace proof, and also for the direct number-theoretic approach.

1 Introduction

Fermat's Little Theorem is a famous result in basic number theory. When p is prime, then

$$a^p \equiv a \pmod{p} \qquad \text{for any natural number } a.$$

Though resources like Wikipedia [15] provide an extensive range of proofs of this result, it seems that standard practice in interactive proof assistants (*e.g.* Hurd *et al.* [9]) is to use Euler's generalisation, which is number-theoretic. There is good reason for this: the number theory required is actually quite simple, making it easy to establish the result without needing a great deal of background theory. This paper shows, however, how a number of other proofs, some with interesting ideas, can be performed mechanically.

The simplest such proof (the necklace) is based on combinatorics over lists. It is relatively straightforward to mechanise (we use the HOL4 proof assistant [13], which has built-in support for lists), but aspects of the proof become smoother when it is rephrased in the language of group theory. This required background does not represent a particularly onerous burden for mechanisation. Indeed, the more group theory one has to hand, the more polished the proof becomes.

We also examine the effect of using group theory in number-theoretic approaches.

Overview The rest of the paper is structured as follows. In Sections 2 and 3, we describe both the standard number-theoretic proof, and Golomb's combinatorial necklace proof [4], and their mechanisation. In Section 4, we discuss how the required (rather basic) group theory is mechanised, before showing how this theory can be applied to the necklace proof (in Section 5), and to the number-theoretic proof (in Section 6). We conclude in Section 7, including a comparison of the different approaches in terms of their complexity.

* NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

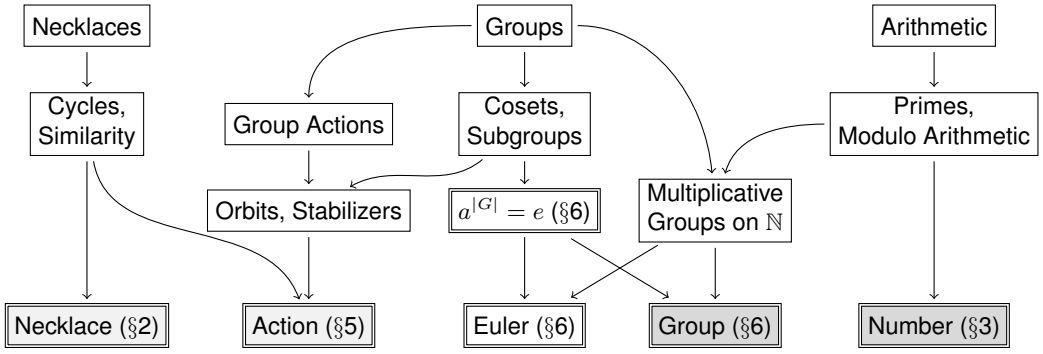

Fig. 1. Theory dependencies for proofs of Fermat's Little Theorem. Double-lined boxes indicate significant results discussed in the corresponding section of the paper. The leftmost Necklace and rightmost Number are direct proofs; others use group theory. Combinatoric results (in light gray) are of $a^p \equiv a \pmod{p}$, which is equivalent (when $0 < a < p$) to number-theoretic results (in dark gray) of $a^{p-1} \equiv 1 \pmod{p}$. Euler is $a^{\varphi(n)} \equiv 1 \pmod{n}$.

Figure 1 gives a graphical summary of the logical dependencies underlying all of the proofs we discuss. The graph falls into three parts:

- The leftmost route shows Golomb's necklace proof, a combinatorial proof based on rotations of lists.
- The rightmost route shows an elementary number-theoretic proof commonly found in textbooks.
- The central paths are of various group-theoretic proofs. The first path from the left is the application of group theory (*via* the notions of action, orbit, and stabilizer) to the necklace proof. The others show the proof of the group-theoretic analogue of Fermat's Little Theorem, followed by the derivation of specific results in the domain of natural numbers.

HOL4 Notation and Theorems All statements appearing with a turnstile ($\vdash$) are HOL4 theorems, automatically pretty-printed to LaTeX from the relevant theory in the HOL4 development. Notation specific to this paper is explained as it is introduced. Otherwise, HOL4 supports a notation that is a generally pleasant combination of quantifiers ($\forall$, $\exists$) and functional programming (λ for function abstraction, juxtaposition for function application).

Lists are written between square brackets, *e.g.,* `[1; 2]`. The length of a list ℓ is written $|\ell|$. The concatenation of ℓ_1 and ℓ_2 is written ℓ_1 `++` ℓ_2. Sets are written between braces, also allowing comprehensions such as $\{x \mid x < 6\}$. Sets also support standard operations such as cardinality (also written with vertical bars: $|\{3; 5\}| = 2$), union (`∪`), intersection (`∩`), and difference (`\`). We write `IMAGE` f s for the image of the set s under function f, and `BIJ` f s_1 s_2 means that function f is a bijection between sets s_1 and s_2. The term R `equiv_on` s means that R is an equivalence relation on the set s, and `partition` R s denotes the set of subsets of s that are partitions with respect to an equivalence relation R.

Our Contribution As already noted, Fermat's Little Theorem has been mechanised a number of times before, *e.g.*, in Coq [10], ACL2 [12] and HOL Light [6]. The minimal group theory we used and mechanised is also very standard.[1] Our contribution is the mechanisation of the necklace proof, in direct and

[1] The Orbit-Stabiliser theorem has not been mechanised before in HOL, but this is a minor contribution given the existing work in other systems such as Coq.

group-theoretic styles (we believe both to be entirely novel). We also compare these proofs with the standard number-theoretic approaches.

Availability HOL4 proof scripts can be found at `http://bitbucket.org/jhlchan/hol/src`. The linearised scripts (discussed in Section 7) are those beginning with prefix `All` in the `fermat` directory. Proofs as they were developed (in various separate theories) are laid out in sub-directories below `fermat`.

2 The Necklace Proof

Leonard Eugene Dickson, in his authoritative treatise *History of the Theory of Numbers* [3, Chapter 3], thoroughly documented all known proofs of this Fermat's result, up to 1919. Among them was this nice combinatorial proof by Julius Petersen in 1872:

> *Take p elements from q with repetitions in all ways, that is, in q^p ways. The q sets with elements all alike are not changed by a cyclic permutation of the elements, while the remaining $q^p - q$ sets are permuted in sets of p [when p is prime]. Hence p divides $q^p - q$.*

This idea is the basis of the *Necklace proof* of S. W. Golomb [4], which has since been re-discovered or discussed by many others (*e.g.*, Smyth [14], Rouse [11], Evans [7], and Conrad [2]).

The necklaces of Golomb's proof are the p elements drawn from a set of cardinality q. Where Peterson has cyclic permutations, Golomb's version adds the image of rotating beads on a necklace (Figure 2).

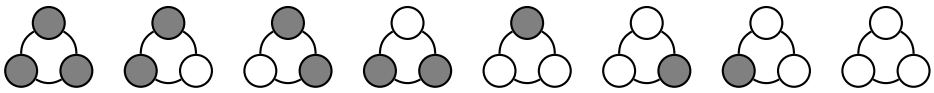

Fig. 2. Necklaces with 3 beads in 2 colours. The first and last are *monocoloured* necklaces. The other *multicoloured* necklaces are divided into 2 parts: those with one white bead and those with two white beads. Multicoloured necklaces in each part can cycle only among themselves. Note that, for 3 beads, each part consists of 3 necklaces. Hence the number of multi-2-coloured necklaces with 3 beads (which is $2^3 - 2 = 8 - 2 = 6$) is divisible by 3.

2.1 Necklaces and Colours

Consider the set N of necklaces of length n (*i.e.*, n beads) with a colours (*i.e.*, a choices for a bead's colour). Since a bead can have any of the a colours, and there are n beads in total, the total number of necklaces is $|N| = a^n$, or $|N| = a^n$. Of these necklaces, the *monocoloured* necklaces are those with the same colour for all beads; the others are *multicoloured* necklaces.

Let S (for single) denote the set of monocoloured necklaces, and M (for multiple) denote the multicoloured necklaces. Clearly, $N = S \cup M$, and $S \cap M = \emptyset$ — that is, S and M are disjoint, and they form a partition of the set of necklaces N. Since there is only 1 monocoloured necklace for each colour, the number of monocoloured necklaces $|S|$ is just a. Given that the two types of necklaces partition the whole set, the number of multicoloured necklaces $|M|$ is equal to $|N| - |S| = a^n - a$.

HOL Implementation Let (`necklace` n a) be the set of necklaces of length n with a colours. The HOL definition is

$\vdash$ `necklace` n a = $\{\ell \mid |\ell| = n \wedge$ `set` $\ell \subseteq$ `count` $a\}$

Our necklaces' beads are just natural numbers, and the definition requires that the "colours" of the necklace are simply drawn from the set (`count` a): the set of natural numbers less than a.

Simple properties of the set (`necklace` n a) readily follow from the definition:

$\vdash$ `FINITE (necklace` n a`)`
$\vdash$ $|$`necklace` n $a|$ = a^n

The monocoloured and multicoloured necklaces are defined thus:

$\vdash$ `monocoloured` n a = $\{\ell \mid \ell \in$ `necklace` n $a \wedge$ `(`$\ell \neq$ `[] ⇒ SING (set` ℓ`))}`
$\vdash$ `multicoloured` n a = `necklace` n $a \setminus$ `monocoloured` n a

where `SING (set` ℓ`)` means that the set of list elements (`set` ℓ) is a singleton. The cardinality results for these sets are straightforward:

$\vdash$ 0 $<$ $n \Rightarrow$ $|$`monocoloured` n $a|$ = a
$\vdash$ 0 $<$ $n \Rightarrow$ $|$`multicoloured` n $a|$ = $a^n - a$

In order to show that the last expression $a^n - a$ is divisible by n when length n is prime, we need to know something more about the multicoloured necklaces, especially how an equivalence relation involving cyclic permutations partitions the set.

2.2 Cycles

Necklaces are represented by lists of length n. Following the imagery, it is natural to think of them as being joined from end to end. We define a `cycle` operation on lists:

$\vdash$ `cycle` n ℓ = `FUNPOW (`$\lambda\ell$`. DROP 1` ℓ `++ TAKE 1` ℓ`)` n ℓ

The expression `DROP` n ℓ discards the first n elements of list ℓ, returning whatever remains, while `TAKE` n ℓ returns the first n elements (for the empty list `[]`, `TAKE` and `DROP` both return `[]`). By putting $n = 1$, this is chopping off the first bead, shifting it to the other end and adding it back. Therefore `DROP 1` ℓ `++ TAKE 1` ℓ represents a rotation by 1 bead position. Then `FUNPOW` just repeats this operation n times.[2] These elementary facts about `cycle` follow immediately:

$\vdash$ `cycle 0` ℓ = ℓ (`CYCLE_0`)
$\vdash$ `cycle` n `(cycle` m ℓ`) = cycle (`n + m`)` ℓ (`CYCLE_ADD`)

Applying `cycle` on a necklace results in another necklace, of the same length and colours:

$\vdash$ $\ell \in$ `necklace` n $a \Rightarrow \forall k$. `cycle` k $\ell \in$ `necklace` n a
$\vdash$ $|$`cycle` n $\ell|$ = $|\ell|$
$\vdash$ `set (cycle` n ℓ`) = set` ℓ

As a result, `cycle` of a monocoloured necklace is still monocoloured, and `cycle` of a multicoloured necklace is still multicoloured, as expected.

[2] This definition of `cycle` n ℓ using `FUNPOW` makes sense for all n, whereas a definition using `TAKE` n and `DROP` n would only work when $n \leq |\ell|$.

We can reason about cycles with modular arithmetic:

$\vdash \ell \neq [] \Rightarrow \texttt{cycle}\ n\ \ell = \texttt{cycle}\ (n \bmod |\ell|)\ \ell$ (CYCLE_MOD_LENGTH)
$\vdash \ell \neq [] \Rightarrow \texttt{cycle}\ m\ (\texttt{cycle}\ n\ \ell) = \texttt{cycle}\ ((m + n) \bmod |\ell|)\ \ell$

And ultimately, a cycle can come full-circle, in multiples, or can be undone by another cycle:

$\vdash \texttt{cycle}\ |\ell|\ \ell = \ell$ (CYCLE_BACK)
$\vdash \texttt{cycle}\ n\ \ell = \ell \Rightarrow \forall m.\ \texttt{cycle}\ (m \times n)\ \ell = \ell$ (CYCLE_MULTIPLE_BACK)
$\vdash n \leq |\ell| \Rightarrow \texttt{cycle}\ (|\ell| - n)\ (\texttt{cycle}\ n\ \ell) = \ell$ (CYCLE_INV)

Already, one can see the possible connections to group theory.

2.3 Similarity and Partitions

We shall say two necklaces ℓ_1, ℓ_2 are *similar*, denoted $\ell_1 \ == \ \ell_2$, when:

$\vdash \ell_1 == \ell_2 \iff \exists n.\ \ell_2 = \texttt{cycle}\ n\ \ell_1$

That is, ℓ_1 can cycle to ℓ_2 because they consist of the same beads in cyclic order.
The following properties of `(==)` follow from properties of `cycle`:

$\vdash \ell == [] \vee [] == \ell \iff \ell = []$
$\vdash \ell_1 == \ell_2 \Rightarrow |\ell_1| = |\ell_2|$

With a little more effort, the fact that `(==)` is an equivalence relation can be proved:

$\vdash \ell == \ell$
$\vdash \ell_1 == \ell_2 \Rightarrow \ell_2 == \ell_1$
$\vdash \ell_1 == \ell_2 \wedge \ell_2 == \ell_3 \Rightarrow \ell_1 == \ell_3$

The key for reflexivity is `CYCLE_0`, for symmetry is `CYCLE_INV`, for transitivity is `CYCLE_ADD`.
Let us denote the equivalence classes under `(==)` by `associates`:

$\vdash \texttt{associates}\ x = \{y \mid x == y\}$

As `(==)` is an equivalence relation, the `associates` partition the set of necklaces. This partitioning has a particularly simple structure when the necklace length n is prime.

2.4 Multicoloured Necklaces with Prime Length

First, an important result about values that "`cycle` back" and their greatest common divisor:

Theorem 1. *If two values m, n can cycle back, the value* `gcd m n` *can also cycle back.*

$\vdash \texttt{cycle}\ m\ \ell = \ell \wedge \texttt{cycle}\ n\ \ell = \ell \Rightarrow \texttt{cycle}\ (\texttt{gcd}\ m\ n)\ \ell = \ell$

Proof. If n = `0`, then `cycle (gcd m 0) ℓ` $=$ `cycle m ℓ` $= \ell$ by assumption. Otherwise, we can use Bézout's identity, called `LINEAR_GCD` in HOL library, which states that if $n \neq$ `0`, then there exist p and q such that $p \times n = q \times m +$ `gcd m n`, and reason:

	`cycle (gcd m n) ℓ`	
$=$	`cycle (gcd m n) (cycle (q × m) ℓ)`	by `CYCLE_MULTIPLE_BACK`
$=$	`cycle (gcd m n + q × m) ℓ`	by `CYCLE_ADD`
$=$	`cycle (p × n) ℓ`	by `LINEAR_GCD`
$=$	ℓ	by `CYCLE_MULTIPLE_BACK`

□

A distinguishing feature of monocoloured necklaces is:

Theorem 2. *A necklace ℓ is monocoloured iff* `cycle 1` $\ell = \ell$.

$\vdash$ `0` $< n \wedge$ `0` $< a \wedge \ell \in$ `necklace` $n\ a \Rightarrow$
`(`$\ell \in$ `monocoloured` $n\ a \iff$ `cycle 1` $\ell = \ell$`)`

Proof. A monocoloured necklace ℓ has all beads the same colour, so shifting 1 bead makes no difference, hence `cycle 1` $\ell = \ell$. Conversely, given `cycle 1` $\ell = \ell$, applying `CYCLE_MULTIPLE_BACK`, $\ell =$ `cycle 2` $\ell =$ `cycle 3` $\ell = \ldots$. As lists, head of ℓ is the first bead, head of `cycle 1` ℓ is the second bead, head of `cycle 2` ℓ is the third bead, *etc*. Since these cycle lists are all the same, and equal lists mean equal heads, all beads have the same colour, making the necklace ℓ monocoloured. □

We proceed to find the size of `associates` of multicoloured necklaces with prime length:

Theorem 3. *For multicoloured necklaces ℓ with prime $|\ell| = p$, the* `cycle` *map from* `count` p *to* `associates` ℓ *is injective.*

$\vdash$ `prime` $p \wedge \ell \in$ `multicoloured` $p\ a \Rightarrow$
`INJ (`λn`.` `cycle` $n\ \ell$`) (count` p`) (associates` ℓ`)`

Proof. This is to show that, for all $x < p$ and $y < p$, `cycle` $x\ \ell =$ `cycle` $y\ \ell \Rightarrow x = y$. Suppose this is not the case. Then there are $x \neq y$ such that there is a common $\ell' =$ `cycle` $x\ \ell =$ `cycle` $y\ \ell$. Note that both necklaces ℓ' and ℓ are multicoloured with same length p (Section 2.2). Without loss of generality, assume $x < y$. Then $y = d + x$, where difference $d > 0$ and $d < p$ (since both $x < p$ and $y < p$). Hence `cycle` $d\ \ell' =$ `cycle` d `(cycle` $x\ \ell$`) =` `cycle (`d `+` x`)` $\ell =$ `cycle` $y\ \ell = \ell'$. With `cycle` $d\ \ell' = \ell'$, and from `CYCLE_BACK` we have `cycle` $p\ \ell' = \ell'$, hence `cycle (gcd` $d\ p$`)` $\ell' = \ell'$ by Theorem 1. But `gcd` $d\ p = 1$ for prime $p, 0 < d < p$. This implies the multicoloured necklace ℓ' has `cycle 1` $\ell' = \ell'$, which is a contradiction in view of Theorem 2. □

Theorem 4. *For multicoloured necklaces ℓ with $|\ell| = n$ (prime or non-prime), the* `cycle` *map from* `count` n *to* `associates` ℓ *is surjective.*

$\vdash \ell \in$ `multicoloured` $n\ a \Rightarrow$ `SURJ (`λk`.` `cycle` $k\ \ell$`) (count` n`) (associates` ℓ`)`

Proof. This is because, if a necklace ℓ' is similar to ℓ, there is a k such that $\ell' =$ `cycle` $k\ \ell$. By `CYCLE_MOD_LENGTH`, $\ell' =$ `cycle (`$k \mod n$`)` ℓ, and so it is in the range of `count` n. □

Theorem 5. *For multicoloured necklaces ℓ with prime $|\ell| = p$, their associates have size p.*

$\vdash$ `prime` $p \wedge \ell \in$ `multicoloured` $p\ a \Rightarrow |$`associates` $\ell| = p$

Proof. Since the `cycle` map is surjective in general (Theorem 4), and injective when the necklace length is prime (Theorem 3), there is a bijection between `count` p and `associates` ℓ for multicoloured necklaces ℓ when $|\ell| = p$ is prime. The result follows from this bijection between finite sets. □

This leads directly to the following mechanisation of the necklace proof of

Theorem 6. *Fermat's Little Theorem.*

$\vdash$ `prime` $p \Rightarrow p$ `divides` $a^p - a$

Proof. For prime p, the multicoloured necklaces $\ell \in$ `multicoloured` $p\ a$ are *"permuted in sets of p"*, as claimed by Julius Petersen (Section 2), since $|$`associates` $\ell| = p$ by Theorem 5. Recall that `associates` ℓ are the equivalence classes of `(==)` on `multicoloured` $p\ a$ (Section 2.3). Since equivalence classes form a partition, and here they all have the same size p, we have:

$$|\texttt{multicoloured}\ p\ a| = p \times |\texttt{partition (==) (multicoloured}\ p\ a\texttt{)}|$$

As p is prime, $0 < p$, and $|$`multicoloured` $p\ a| = a^p - a$ (Section 2.1). Combining these results we have p `divides` $a^p - a$ by definition of `divides`. □

3 Direct Number-Theoretic Proof

The proof of Fermat's Little Theorem given in most textbooks, and also that given in various theorem-proving systems, is number-theoretic, based on properties of modulo arithmetic. In particular, modulo prime multiplication has some special properties. The first one, usually referred to as Euclid's Lemma, is that a prime divides a product iff the prime divides a factor. In terms of modulo arithmetic, this is:

$\vdash$ `prime` $p \Rightarrow (x \times y \equiv 0 \pmod{p} \iff x \equiv 0 \pmod{p} \vee y \equiv 0 \pmod{p})$

Thus, left-cancellation of a non-zero factor is possible in prime modulo arithmetic:

$\vdash$ `prime` $p \wedge x \times y \equiv x \times z \pmod{p} \wedge x \not\equiv 0 \pmod{p} \Rightarrow y \equiv z \pmod{p}$

Definition 1. *Let the* residues *of prime p be the non-zero numbers less than p:* $\{1 \ldots p-1\}$.

Take any a from the residues of p, and consider the various values of $a \times x \mod p$, for all x also a residue of p. In HOL, this is denoted by a *row* operation:

Definition 2. $\vdash$ `row` $p\ a\ x = a \times x \mod p$

Theorem 7. *The row products form a permutation of the residues for prime modulo.*

$\vdash$ `prime` $p \wedge a \in \{1 \ldots p-1\} \Rightarrow \{1 \ldots p-1\} =$ `IMAGE (row` $p\ a$`)` $\{1 \ldots p-1\}$

Proof. The `IMAGE` on the right-hand side is equivalent to $\{a \times x \mod p \mid 1 \leq x \wedge x < p\}$. The possible remainders under modulo p are $0, 1, \ldots p-1$. Since a prime p has no proper factors, and both a and x are less than p, the product $a \times x$ cannot be the prime p, nor any multiple of the prime p. Hence the remainder, $a \times x \mod p$ cannot be zero, making this result one of the residues of p. The possible values are distinct because if $a \times x \equiv a \times y \pmod{p}$, then $x \equiv y \pmod{p}$ by left-cancellation of non-zero a. So the right-hand side, the row products, is just a permutation of the left-hand side, the residues of p. □

This is the key for the number-theoretic proof of

Theorem 8. *Fermat's Little Theorem (equivalent form[3])*

$\vdash$ `prime` $p \wedge a \in \{1 \ldots p-1\} \Rightarrow a^{p-1} \equiv 1 \pmod{p}$

Proof. Consider multiplying all numbers (denoted by the symbol $\prod$) from each of these finite sets:
(1) the residues $\{1 \ldots p-1\}$ and (2) its row products `IMAGE (row` $p\ a$`)` $\{1 \ldots p-1\}$.
Clearly, the first one is a factorial:

$\vdash \prod \{1 \ldots p-1\} =$ `(`$p-1$`)!`

For the second one, since for numbers the order of multiplication does not affect the product, all the factors a of (`row` $p\ a$) (Definition 2) can be collected together, so we have:

$\vdash$ `prime` $p \wedge a \in \{1 \ldots p-1\} \Rightarrow$
$\prod$ `(IMAGE (row` $p\ a$`)` $\{1 \ldots p-1\}$`)` $\equiv a^{p-1} \times$ `(`$p-1$`)!` $\pmod{p}$

As the underlying sets are the same due to permutation (Theorem 7), the two products under modulo p are identical:

$\vdash$ `prime` $p \wedge a \in \{1 \ldots p-1\} \Rightarrow$ `(`$p-1$`)!` $\equiv a^{p-1} \times$ `(`$p-1$`)!` $\pmod{p}$

A prime p has no proper factor, and `(`$p-1$`)!` has of all the numbers less than p, so `(`$p-1$`)!` $\not\equiv 0 \pmod{p}$. Applying non-zero left-cancellation of modulo p multiplication gives Fermat's Little Theorem. □

[3] To show $a^p \equiv a \pmod{p}$ for all a, it is sufficient to show this for $0 \leq a < p$, the possible remainders under modulo p. The case $a = 0$ is trivial. The case $0 < a < p$ has `gcd` $a\ p = 1$, since p is prime. This allows left-cancellation of non-zero a on both sides, giving the equivalent form $a^{p-1} \equiv 1 \pmod{p}$.

4 Group Theory

The combinatorial necklace proof and the number-theoretic proof may appear unrelated, but there is an underlying algebra behind both proofs, that of group theory. The algebra gives us:

- the cycles and similarities in the necklace proof;
- the factor cancellation in the number-theoretic proof; and
- an insight, allowing a modest generalisation.

The discussion that follows is an expansion of this theme.

We mechanise the necessary theorems from group theory following an existing mechanisation in the HOL distribution by Joe Hurd [9].[4] We have a predicate `Group` g on a record of four fields (the group operation ($x \times y$), the inverse (x^{-1}), the identity (e) and the carrier set). We abuse notation to allow `G` and `H` to stand for the carrier sets of groups g and h respectively:

$$\begin{array}{l}\vdash \texttt{Group}\ g \iff \\ \quad e \in \texttt{G} \wedge (\forall x\ y\texttt{::(G)}.\ x \times y \in \texttt{G}) \wedge (\forall x\texttt{::(G)}.\ x^{-1} \in \texttt{G}) \wedge \\ \quad (\forall x\texttt{::(G)}.\ e \times x = x) \wedge (\forall x\texttt{::(G)}.\ x^{-1} \times x = e) \wedge \\ \quad \forall x\ y\ z\texttt{::(G)}.\ x \times y \times z = x \times (y \times z)\end{array}$$

The double-colon notation (*e.g.*, $\forall x\ \ y\texttt{::(G)}.\ P\ x\ y$) is a restriction on all the preceding bound variables (x and y here) requiring them to be in the set `G`.

Typical results in this mechanisation appear with the `Group` predicate as a side-condition.

$$\begin{array}{l}\vdash \texttt{Group}\ g \Rightarrow \forall x\ y\ z\texttt{::(G)}.\ x \times y = x \times z \iff y = z \\ \vdash \texttt{Group}\ g \Rightarrow \forall x\ y\ z\texttt{::(G)}.\ x \times y = z \iff x = z \times y^{-1} \\ \vdash \texttt{Group}\ g \Rightarrow \forall x\texttt{::(G)}.\ (x^{-1})^{-1} = x\end{array}$$

This is perhaps not the slickest possible presentation of abstract algebra, even within the constraints of HOL4's simple type theory, but it is both well-understood and sufficient for our purposes.

Group *exponentiation* is defined *via* primitive recursion, giving us the usual properties:

$$\begin{array}{l}\vdash \texttt{Group}\ g \wedge x \in \texttt{G} \Rightarrow x^0 = e \\ \vdash \texttt{Group}\ g \wedge x \in \texttt{G} \Rightarrow x^1 = x \\ \vdash \texttt{Group}\ g \wedge x \in \texttt{G} \Rightarrow x^{m \times n} = (x^m)^n \\ \vdash \texttt{Group}\ g \wedge x \in \texttt{G} \Rightarrow (x^n)^{-1} = (x^{-1})^n\end{array}$$

We write $h \leq g$ to mean that h is a subgroup of g, and define the *coset* of a set X with respect to a group element a (normally written aX) to be

$$\vdash \texttt{coset}\ g\ X\ a = \texttt{IMAGE}\ (\lambda z.\ a \times z)\ X$$

The cosets of a subgroup's carrier are important because of these standard results:

Theorem 9. *Subgroup cosets partition the group's carrier set, by the following equivalence relation:*

$$\vdash \texttt{Group}\ g \wedge h \leq g \Rightarrow \texttt{coset}\ g\ \texttt{H}\ \texttt{equiv\_on}\ \texttt{G}$$

Theorem 10. *Each coset of a subgroup is in bijection with the subgroup itself.*

$$\vdash \texttt{Group}\ g \wedge h \leq g \wedge a \in \texttt{G} \Rightarrow \texttt{BIJ}\ (\lambda x.\ a \times x)\ \texttt{H}\ (\texttt{coset}\ g\ \texttt{H}\ a)$$

[4] The source code of a prior HOL mechanisation of group theory by Elsa L. Gunter [5] is not generally available.

This bijection allows determination of the size of subgroup cosets:

Theorem 11. *For a finite subgroup, the size of its coset equals the size of subgroup itself:*

$\vdash$ `Group` $g \wedge h \leq g \wedge a \in$ `G` $\wedge$ `FINITE H` $\Rightarrow$ `|coset` g `H` a`|` $=$ `|H|`

Therefore the subgroup cosets partition consists of equal-size chunks, leading to Lagrange's Identity:

$\vdash$ `FiniteGroup` $g \wedge h \leq g \Rightarrow$ `|G|` $=$ `|H|` $\times$ `|partition (coset` g `H) G|`

and Lagrange's Theorem on cardinality of subgroups:

$\vdash$ `FiniteGroup` $g \wedge h \leq g \Rightarrow$ `|H| divides |G|`

5 Group Theory Applied to the Necklace Proof

The group-theoretic version of the necklace proof requires a little more theory than the basic development of the preceding section. We shall use the group $\mathbb{Z}_n^+$, which is the additive group over the natural numbers less than n. This group's binary operation is addition modulo n, and its identity is zero.

5.1 Group Actions

Definition 3. *Let g be a group over a set of elements of type α, X be a set of elements of type β, and (infix) $\circ$ a function of type α -> β -> β. The mapping $(\circ)$ is called a* group action *from g to X, (written* `action (∘) g X`*), if these three conditions are satisfied:*

- *Closure:* $a \in G \wedge x \in X \Rightarrow a \circ x \in X$
- *Identity:* $x \in X \Rightarrow e \circ x = x$
- *Composition:* $a, b \in G \wedge x \in X \Rightarrow a \circ (b \circ x) = (a \times b) \circ x$

The HOL definition is

```
⊢ action (∘) g X ⟺
    ∀x.
      x ∈ X ⇒
        (∀a::(G). a ∘ x ∈ X) ∧ e ∘ x = x ∧
        ∀a b::(G). a ∘ b ∘ x = (a × b) ∘ x
```

The set X above is called the *target*. We can picture a target point $x \in X$ being *acted upon* by the group elements. Alternatively, we say that point x can *reach* another point $a \circ x$ for some $a \in G$. If $a \circ x = x$, we say that the group element a leaves the point x *fixed*. This leads to the following:

Definition 4. *For $x \in X$, the set of target points it can reach form its* **`orbit`**.

Definition 5. *For $x \in X$, the set of group elements that leave it fixed form its* **`stabilizer`**.

For example, $\mathbb{Z}_n^+$ acts on the set of necklaces of length n, with `cycle` being an action from $\mathbb{Z}_n^+$ to the necklaces. Each monocoloured necklace always cycles to itself. Thus its orbit consists of itself only, and its stabilizer is all of the group's carrier. For each multicoloured necklace, cycling brings it to another (similar) multicoloured necklace. Since $|\mathbb{Z}_n^+| = n$, its orbit contains at most n reachable points in the target. Generally, more reachable points give a larger orbit, and the corresponding stabilizer is smaller. In the extreme case when the orbit has n distinct target points, the stabilizer contains just the group identity.

This is a hint that the sizes of orbits and stabilizers may have a relationship — an issue we shall explore.

HOL Implementation The HOL definitions of these concepts pick up multiple parameters, so that, for example, the `reach` relation is not simply a binary notion but has to include explicit parameters for the action and the governing group. Similar extra parameters are required for `orbit` and `stabilizer` definitions:

$\vdash$ `reach (∘) g x y` $\iff \exists a.\ a \in$ `G` $\wedge\ a \circ x = y$
$\vdash$ `orbit (∘) g X x` $= \{y \mid y \in X \wedge$ `reach (∘) g x y` $\}$
$\vdash$ `stabilizer (∘) g x` $= \{a \mid a \in g.$`carrier` $\wedge\ a \circ x = x\}$

For presentational reasons, we shall assume fixed operation $(\circ)$, group g and target set X in much of what follows, and use the following abbreviations in prose and HOL theorems:

- *orbit* x for `orbit (∘) g X` x, and
- *stabilizer* x for `stabilizer (∘) g` x.

5.2 Action Basics

Properties of group actions blend nicely with properties of groups, as shown by these basic results.

Theorem 12. *Reachability is an equivalence relation on the target set.*

$\vdash$ `Group g` $\wedge$ `action (∘) g X` $\Rightarrow$ `reach (∘) g equiv_on X`

Proof. Let $x \sim y$ stand for `reach (∘) g x y`. By action identity: $e \circ x = x$, hence $x \sim x$, or `reach` is *reflexive*. If $a \in G$ moves point x to y: $a \circ x = y$, then $a^{-1} \in G$ moves y to x: $a^{-1} \circ y = a^{-1} \circ (a \circ x) = (a^{-1} \times a) \circ x = e \circ x = x$, hence $x \sim y \Rightarrow y \sim x$, or `reach` is *symmetric*. If $a \circ x = y, b \circ y = z$, then by action composition: $(b \times a) \circ x = b \circ (a \circ x) = b \circ y = z$, hence $x \sim y \wedge y \sim z \Rightarrow x \sim z$, or `reach` is *transitive*. Thus `reach` is an equivalence relation. □

The *orbits* are equivalence classes of `reach`, and they form a partition of the target set X. Another characterisation of `orbit` using the action mapping $(\circ)$ is:

$\vdash$ `Group g` $\wedge$ `action (∘) g X` $\wedge\ x \in X \Rightarrow$ *orbit* $x = \{a \circ x \mid a \in$ `G`$\}$

Theorem 13. *The stabilizer of a point in the target set forms a subgroup.*

$\vdash$ `action (∘) g X` $\wedge\ x \in X \wedge$ `Group g` $\Rightarrow$ `StabilizerGroup (∘) g x` $\leq g$

Proof. If two elements $a, b \in G$ fix a point $x \in X$, *i.e.*, $a \circ x = x$ and $b \circ x = x$, then by action composition: $(a \times b) \circ x = a \circ (b \circ x) = a \circ x = x$. Therefore, the stabilizer is a closed subset of G. The identity e is in the stabilizer by action identity: $e \circ x = x$. If a is in the stabilizer, its inverse a^{-1} is also in the stabilizer since $a^{-1} \circ x = a^{-1} \circ (a \circ x) = (a^{-1} \times a) \circ x = e \circ x = x$. Hence the stabilizer is a subgroup. □

5.3 Orbit-Stabilizer Theorem

Consider a point $x \in X$. Its orbit is the set of points reachable through the action of all group elements $a \in G$. If all action points $a \circ x$ are distinct, only $e \circ x = x$ fixes x, hence its stabilizer consists of the group identity e only, the smallest possible subgroup. For example, let $G = \{a, b, c, d, e, f\}$ with e being the identity, and $X = \{x, y, z, \dots\}$. An example of such a group action is shown in Figure 3.

What happens if not all action points are distinct? This is interesting:

Theorem 14. *If action points coincide:* $a \circ x = b \circ x$*, the quotient* $a^{-1} \times b$ *lies in the stabilizer of* x*.*

$\vdash$ `Group g` $\wedge$ `action (∘) g X` $\wedge\ x \in X \Rightarrow$
$\quad \forall a\ b$`::(G).` $a \circ x = b \circ x \iff a^{-1} \times b \in$ *stabilizer* x

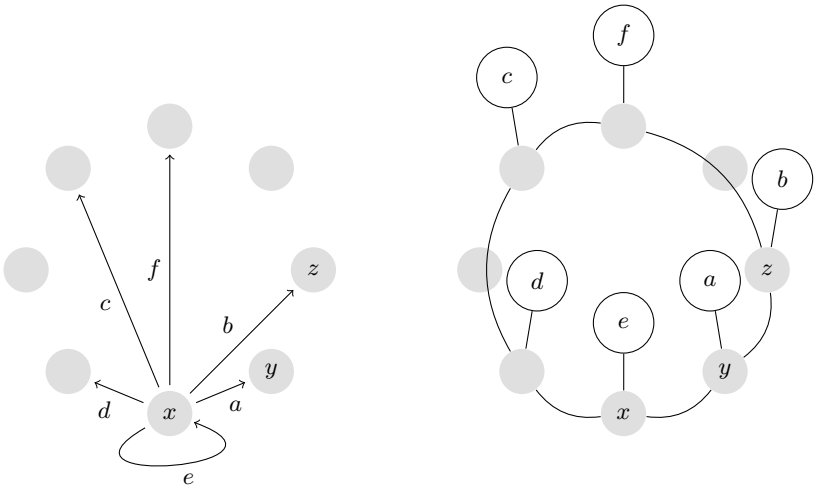

Fig. 3. If a group action, shown on the left, maps a point $x \in X$ to distinct reachable points, each reachable point corresponds to only one element in G. These reachable points can be joined together by arcs, shown on the right, giving the $orbit\ x$. The "balloon" over each $y \in orbit\ x$ contains the group element which acts on x to reach y. Note the balloon over x is $stabilizer\ x$, in this case just $\{e\}$, corresponds to the self-loop over x on the left.

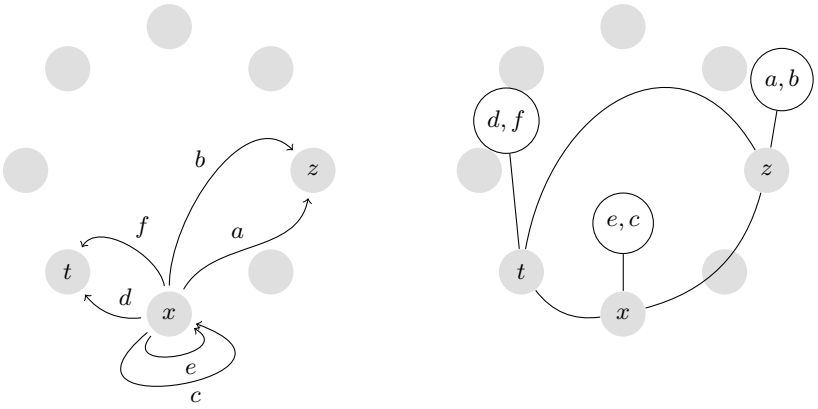

Fig. 4. If $stabilizer\ x$, shown on the left as self-loops over x, has two elements $\{e, c\}$, then every $z \in orbit\ x$ is reachable by two group elements: $z = a \circ x = a \circ (c \circ x) = (a \times c) \circ x = b \circ x$ where $b = a \times c$, and $b \neq a$ since $c \neq e$. On the right are shown the two group elements for every point in $orbit\ x$ inside its "balloon". The balloon over x is $stabilizer\ x$; the other balloons are cosets of $stabilizer\ x$.

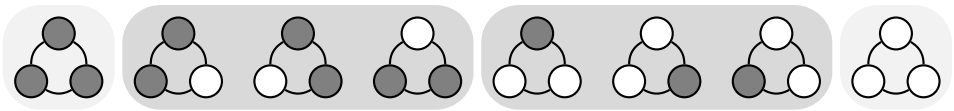

Fig. 5. Orbits of necklaces with 3 beads in 2 colours under cycle action by $\mathbb{Z}_3^+$ are shown as background round rectangles. Light gray orbits of monocoloured ones always have size $= 1$. Dark gray orbits of multicoloured ones always have size > 1. By Orbit-Stabilizer theorem, orbit size divides $|\mathbb{Z}_3^+| = 3$, hence multicoloured orbit size $= 3$.

Proof. From left to right, apply the action a^{-1} to both sides of $a \circ x = b \circ x$. Thus $x = (a^{-1} \times b) \circ x$. From right to left, we have $(a^{-1} \times b) \circ x = x$. Apply the action a to both sides of this equation, deriving $b \circ x = a \circ x$ as required. □

Furthermore $a^{-1} \times b$ is some $c \in G$ by closure property of the group. By uniqueness of group inverses, $c \neq e$ if $a \neq b$. Hence for distinct a, b with $a \circ x = z = b \circ x$, there is an element $c \neq e$ and $c \in stabilizer\ x$. Note that $a^{-1} \times b = c$ implies $b = a \times c$. So if, for example (Figure 4), $stabilizer\ x$ is indeed just $\{e, c\}$, then the set of group elements enabling x to reach z, *i.e.* $\{a, b\} = \{a \times e, a \times c\} = a\ \{e, c\}$, is a coset of $stabilizer\ x$.

Similar reasoning shows that, for any point $y \in orbit\ x$, the set of group elements $\{a \in G \mid a \circ x = y\}$ that enables x to reach y will be a coset of $stabilizer\ x$. In HOL, this is expressed as:

Theorem 15. *The set of group elements enabling x to reach point $y \in orbit\ x$ is a coset of $stabilizer\ x$.*

```
⊢ Group g ∧ action (∘) g X ∧ x ∈ X ∧ y ∈ orbit x ⇒
    {a | a ∈ G ∧ a ∘ x = y} =
      coset g (stabilizer x) (actionElement (∘) g x y)
```

where (`actionElement (∘) g x y`) *is a group element that acts on x, generating y.*

Such an element exists when x and y are in the same orbit, as assumed. Since the choice made by `actionElement` may not be drawing on a singleton set, this association of a point $y \in orbit\ x$ with a coset is only meaningful if the coset is independent of this choice. This follows from a standard result about subgroup cosets:

Theorem 16. *Two cosets of a subgroup are equal when it has the quotient of their generating elements.*

```
⊢ Group g ∧ h ≤ g ⇒
    ∀b a::(G). coset g H b = coset g H a ⟺ a⁻¹ × b ∈ H
```

Proof. For the if-part ($\Rightarrow$), since $b \in bH$, $bH = aH$ implies there is a $c \in H$ such that $b = a \times c$. Solving for c in a group: $c = a^{-1} \times b \in H$. For the only-if part ($\Leftarrow$), since $(a^{-1} \times b) \in H$, so for any $c \in H$, $(a^{-1} \times b) \times c$ equals to some $d \in H$, by closure property of a subgroup. Now $(a^{-1} \times b) \times c = d$ implies $b \times c = a \times d$, for any $c \in H$. This shows $bH \subseteq aH$. Repeating the same argument with $b^{-1} \times a = (a^{-1} \times b)^{-1} \in H$, as a subgroup includes all inverses, gives $aH \subseteq bH$. Thus $bH = aH$. □

This matching condition is used to prove the association of stabilizer cosets to orbit points (Theorem 15). Together with the matching condition of reachable points (Theorem 14), both are crucial in establishing:

Theorem 17. *The points of x's orbit are in bijection with the cosets of x's stabilizer.*

```
⊢ Group g ∧ action (∘) g X ∧ x ∈ X ⇒
    BIJ (λz. coset g (stabilizer x) (actionElement (∘) g x z))
      (orbit x) {coset g (stabilizer x) a | a | a ∈ G}
```

The last set comprehension is a special form marking a as the only variable that varies in the leftmost expression, `coset g (stabilizer x) a`. This bijection provides the key for:

Theorem 18. *Orbit-Stabilizer Theorem.*

```
⊢ FiniteGroup g ∧ action (∘) g X ∧ x ∈ X ∧ FINITE X ⇒
    |G| = |orbit x| × |stabilizer x|
```

Proof. There are $|orbit\ x|$ points in x's orbit. Each point is associated with a coset of $stabilizer\ x$. Since $stabilizer\ x$ is a subgroup (Theorem 13), each coset is the same size as $stabilizer\ x$ (Theorem 11). The cosets form a partition of the carrier set G (Theorem 9), which is counted by the bijection of Theorem 17:

$$|G| = \sum_{a \in G} |a(stabilizer\ x)| = |orbit\ x|\,|stabilizer\ x|$$

□

5.4 Applying Action to Necklaces

The Orbit-Stabilizer theorem is the key to classifying necklace orbits, especially when the necklace length is prime. First we identify the action:

Theorem 19. `cycle` *is an action from* $\mathbb{Z}_n^+$ *to the set of necklaces:*

$\vdash$ `0 < n ∧ 0 < a ⇒ action cycle` $\mathbb{Z}_n^+$ `(necklace n a)`

Proof. For necklace $\ell \in$ `necklace n a`, $|\ell| =$ `n`. Each element $k \in \mathbb{Z}_n^+$, *i.e.* $0 \leq k < n$, maps a necklace ℓ to the cycle result: `cycle k ℓ`, *i.e.* cycling of the necklace by k beads. Recall these earlier results about `cycle` (Section 2.2):

$\vdash \ell \in$ `necklace n a ⇒ ∀k. cycle k ℓ ∈ necklace n a`
$\vdash$ `cycle 0 ℓ = ℓ`
$\vdash \ell \neq$ `[] ⇒ cycle x (cycle y ℓ) = cycle ((x + y) mod |ℓ|) ℓ`

The first shows `cycle` is closed for necklaces. The second shows `cycle` has an identity. The third shows `cycle` composes under modulo n addition. Hence `cycle` is an action from the group $\mathbb{Z}_n^+$. □

Since length and colours are invariants for `cycle` (Section 2.2), a multicoloured necklace cannot be cycled to a monocoloured necklace. This shows `cycle` is also closed for those sets respectively:

$\vdash$ `0 < n ∧ 0 < a ⇒ action cycle` $\mathbb{Z}_n^+$ `(monocoloured n a)`
$\vdash$ `0 < n ∧ 0 < a ⇒ action cycle` $\mathbb{Z}_n^+$ `(multicoloured n a)`

The classification of orbits for necklaces is simple:

Theorem 20. *Only monocoloured necklaces have orbit size equal to* 1.

$\vdash$ `0 < n ∧ 0 < a ∧ ℓ ∈ monocoloured n a ⇒`
`|orbit cycle` $\mathbb{Z}_n^+$ `(monocoloured n a) ℓ| = 1`
$\vdash$ `0 < n ∧ 0 < a ∧ ℓ ∈ multicoloured n a ⇒`
`|orbit cycle` $\mathbb{Z}_n^+$ `(multicoloured n a) ℓ| ≠ 1`

Proof. Only a monocoloured necklace ℓ has `cycle 1 ℓ = ℓ` (Theorem 2), *i.e.* for all multiples k, `cycle k ℓ = ℓ` by `CYCLE_ADD`. Hence only such orbit collapses to a singleton, with cardinality 1. □

Theorem 21. *For multicoloured necklaces of length* p*, a prime, the orbit size of each necklace equals* p*.*

$\vdash$ `prime p ∧ 0 < a ∧ ℓ ∈ multicoloured p a ⇒`
`|orbit cycle` $\mathbb{Z}_p^+$ `(multicoloured p a) ℓ| = p`

Proof. When the necklace length is prime p, the action group is $\mathbb{Z}_p^+$. By the Orbit-Stabilizer theorem (Theorem 17): $|orbit\ \ell| \times |stabilizer\ \ell| = |\mathbb{Z}_p^+| = p$ for any necklace ℓ. A prime p has only trivial factorization: $p = 1 \times p = p \times 1$. By Theorem 20, the orbit of a multicoloured necklace is not a singleton, so its size must be p. □

Recall that `reach` is an equivalence relation (Theorem 12). Orbits are the equivalence classes of `reach`, so they form a partition of the target set. From Theorem 21, the target is (`multicoloured p a`) with size $(a^p - a)$ (Section 2.1). Theorem 21 also specifies an equal-size partition, giving the visual image of division (Figure 5). Hence, p `divides` $a^p - a$, which in modulo p is Fermat's Little Theorem:

$\vdash$ `prime` $p \Rightarrow a^p \equiv a \pmod{p}$

6 Group Theory applied to the Number-Theoretic Proof

Having applied group theory to the necklace proof, it is interesting to try the "same trick" with the number-theoretic proof. The subsequent results are not novel, but allow a fuller comparison of approaches when we conclude.

It is straightforward to recast the number-theoretic proof of Fermat's Little Theorem (Section 3) in the context of finite Abelian groups, the structure that naturally mimics prime modulo multiplication. The factor rearrangement and cancellation are direct consequences of commutativity and cancellation laws in Abelian groups. However, this is not very illuminating, and unnecessarily restrictive, as the group-theoretic version of Fermat's Little Theorem holds for all finite groups (not just the Abelian ones):

$$\vdash \texttt{FiniteGroup}\ g \wedge a \in \texttt{G} \Rightarrow a^{|\texttt{G}|} = e$$

Assuming this result (which will be proved later, see Theorem 22), to derive Fermat's Little Theorem it is sufficient to show that prime modulo multiplication, *i.e.* $\mathbb{Z}_p^*$ for prime p, does indeed form a group — with $|\mathbb{Z}_p^*| = |\{1 \ldots p{-}1\}| = p{-}1$, and the multiplicative identity is 1. Critically, we need to show that any $x \in \{1 \ldots p{-}1\}$ has a multiplicative inverse in $\mathbb{Z}_p^*$. This can be done by appeal to Bézout's identity, a property of `gcd` already used in Theorem 1:

$$\vdash x \neq 0 \Rightarrow \exists k\ q.\ k \times x = q \times p + \texttt{gcd}\ p\ x$$

With p a prime and $0 < x < p$, $\texttt{gcd}\ p\ x = 1$. Taking modulo p on both sides of the equation, the right-hand side becomes 1, and the k on the left gives $k \pmod p$ as the multiplicative inverse of x in $\mathbb{Z}_p^*$.

6.1 Euler's Generalization

When the modulo n is not a prime, the non-zeroes of $\mathbb{Z}_n$ do not form a multiplicative group; *e.g.* to find the multiplicative inverse for 2 in $\mathbb{Z}_6$ would require solving $2x = 6y + 1$, which is impossible by parity. However, Euler observed that the Bézout's identity of the preceding section actually guarantees a multiplicative inverse for $x < n$ whenever $\texttt{gcd}\ n\ x = 1$, *i.e.* x is co-prime to n:

$$\vdash 1 < n \wedge 0 < x \wedge x < n \wedge \texttt{coprime}\ n\ x \Rightarrow \exists y.\ 0 < y \wedge y < n \wedge \texttt{coprime}\ n\ y \wedge y \times x \bmod n = 1$$

This is then the basis for a group, whose carrier is $\mathbb{Z}_n^*$ — the set of elements of $\mathbb{Z}_n$ with multiplicative inverses. The cardinality of this set is known as its *totient*, denoted by $\varphi(n)$:

$$\vdash \mathbb{Z}_n^* = \{x \mid 0 < x \wedge x < n \wedge \texttt{coprime}\ n\ x\}$$
$$\vdash \varphi(n) = |\mathbb{Z}_n^*|$$

Euler's generalisation of Fermat's Little Theorem follows:

$$\vdash 1 < n \wedge 0 < a \wedge a < n \wedge \texttt{coprime}\ n\ a \Rightarrow a^{\varphi(n)} \equiv 1 \pmod n$$

We shall now prove the group-theoretic version of Fermat's Little Theorem for any finite group, *via* the generated subgroup of its elements.

6.2 Generated subgroups

Let a be a group element. Consider the sequence $a, a^2, a^3, \ldots$. If the group is finite, there must eventually be a repetition in this sequence. Assume $m < n$ and $a^m = a^n$, then we can use left-cancellation to remove the common factor a^m, giving us

$$\vdash \texttt{Group}\ g \wedge a \in \texttt{G} \wedge m < n \wedge a^m = a^n \Rightarrow a^{n-m} = e$$

Definition 6. *Call the least non-zero exponent that maps an element back to the identity, its* ***order****:*

$\vdash$ `order` g a = `LEAST` k. `0` $<$ k $\wedge$ a^k = e

The preceding argument shows that `order` exists for finite group elements, and it satisfies:

$\vdash$ `FiniteGroup` g $\wedge$ a $\in$ `G` $\Rightarrow$ `0` $<$ `order` g a $\wedge$ $a^{\texttt{order } g\ a}$ = e
$\vdash$ `FiniteGroup` g $\wedge$ a $\in$ `G` $\Rightarrow$ a^{-1} = $a^{\texttt{order } g\ a-1}$

By properties of group exponentiation (Section 4), the powers of an element $a \in G$ form a subgroup: `Generated` g a, also written as $\langle a \rangle$. This subgroup is related to `order` by:

$\vdash$ `FiniteGroup` g $\wedge$ a $\in$ `G` $\Rightarrow$ |`(Generated` g a`).carrier`| = `order` g a

This result can be deduced by the `LEAST` property of `order`, and provides the key for:

Theorem 22. *Fermat's Little Theorem for finite groups.*

$\vdash$ `FiniteGroup` g $\wedge$ a $\in$ `G` $\Rightarrow$ $a^{|\texttt{G}|}$ = e

Proof. Consider $\langle a \rangle$, the generated subgroup of a $\in$ `G`, with $|\langle a \rangle| =$ `order` g a, and $a^{\texttt{order } g\ a}$ = e. Since $\langle a \rangle$ is a subgroup of g, there is a k such that $|\texttt{G}|$ = `order` g $a \times k$ by Lagrange's Theorem. Hence,

$$a^{|\texttt{G}|} = a^{\texttt{order } g\ a \times k} = (a^{\texttt{order } g\ a})^k = e^k = e$$

as required. □

7 Conclusion

Fermat's Little Theorem is a very basic and well-known result in number theory. Having attempted its proof in a slew of different styles, we now attempt to draw some lessons.

For each proof discussed, we have linearised our various script files into one script containing just the lemmas required for that particular effort. Table 1 includes total line counts for each file. The verdict is clear: the basic number-theoretic approach is much better in terms of overall lines-of-code. However, these results can only be suggestive: our HOL scripts may not be the optimal expression of the various proof strategies, and our own skills may not be uniform across the numerical and algebraic domains.

Type of Proof	Approach (Section ref.)		Filename	Total
Combinatorial	Direct *via* cycles	(2)	`AllFLTnecklaceScript.sml`	824
	Group *via* action	(5)	`AllFLTactionScript.sml`	1387
Number-theoretic	Direct *via* modulo arithmetic	(3)	`AllFLTnumberScript.sml`	473
	Group *via* generated subgroups	(6)	`AllFLTgroupScript.sml`	839
	Euler *via* generated subgroups	(6)	`AllFLTeulerScript.sml`	871

Table 1. Line counts for theories developing each approach. The filename is that of the linearised script in the `bitbucket.org` repository.

Additionally, HOL4's features make some proofs easier to automate, and some goals easier to express. Certainly, we believe that our naïve approach to group theory makes the numbers for the group-theoretic

proofs worse than they might be, and the Orbit-Stabiliser theorem is arguably a steeper requirement than Theorem 22. Sub-types, perhaps best exemplified by their use in Coq (though also approximated and used for group theory in HOL4 by Hurd [8]), would be an obvious way to approach this issue. Isabelle's axiomatic type-classes and locales have also been used to provide appealing mechanisations of abstract algebra. It would be interesting to see what these other systems made of the directly combinatorial necklace proof, and of the group-theoretic version of the same.

Future Work The HOL4 source code provides an example of proving Fermat's Little Theorem using the Binomial Theorem.[5] The proof is by induction, and thus rather different from the proofs in this paper. Indeed, Fermat's Little Theorem and the Binomial Theorem are crucial concepts in Agrawal *et al.'s* famous result [1] that primality testing can be done in polynomial time. A mechanisation of the AKS algorithm is certainly an appealing prospect.

Final Verdict Our results show that we have yet to find the sweet spot when it comes to performing combinatorial proofs in HOL. Our consolation is to have found that attacking the result *via* explicit appeals to group theory gives us two distinct mechanised proofs that are arguably more elegant than their direct analogues. Our mechanisation of the necklace proofs may be the first; we hope it is not the last, and that still more beautiful pearls may be found in this vein.

References

1. Manindra Agrawal, Neeraj Kayal, and Nitin Saxena. PRIMES is in P. *Annals of Mathematics*, 160(2):781–793, 2004.
2. Keith Conrad. Group actions. `http://www.math.uconn.edu/~kconrad/blurbs/grouptheory/gpaction.pdf`, 2008.
3. Leonard Eugene Dickson. *History of the Theory of Numbers: Volume 1: Divisibility and Primality*. Carnegie Institution of Washington, 1919.
4. Solomon W. Golomb. Combinatorial proof of Fermat's "Little" Theorem. *The American Mathematical Monthly*, 63(10):718, 1956.
5. Elsa. L. Gunter. Doing algebra in simple type theory. Technical Report MS-CIS-89-38, Department of Computer and Information Science, Moore School of Engineering,University of Pennsylvania, June 1989.
6. John Harrison. *HOL Light Tutorial (for version 2.20)*. Intel JF1-13, 2011. Section 18.2: Fermat's Little Theorem.
7. Benjamin V. Holt and Tyler J. Evans. A group action proof of Fermat's Little Theorem. `http://arxiv.org/abs/math/0508396`.
8. Joe Hurd. Predicate subtyping with predicate sets. In Richard J. Boulton and Paul B. Jackson, editors, *14th International Conference on Theorem Proving in Higher Order Logics: TPHOLs 2001*, volume 2152 of *Lecture Notes in Computer Science*, pages 265–280. Springer, September 2001.
9. Joe Hurd, Mike Gordon, and Anthony Fox. Formalized elliptic curve cryptography. In *High Confidence Software and Systems: HCSS 2006*, April 2006.
10. Martijn Oostdijk. Library pocklington.fermat. `http://coq.inria.fr/pylons/contribs/files/Pocklington.fermat.html`.
11. Jeremy Rouse. Combinatorial proofs of congruences. Master's thesis, Harvey Mudd College, 2003.
12. David Russinoff. ACL2 Version 3.2 source: `books/quadratic-reciprocity/fermat.lisp`, 2007.
13. Konrad Slind and Michael Norrish. A brief overview of HOL4. In Otmane Ait Mohamed, César Muñoz, and Sofiène Tahar, editors, *Theorem Proving in Higher Order Logics, 21st International Conference*, volume 5170 of *Lecture Notes in Computer Science*, pages 28–32. Springer, 2008.
14. C. J. Smyth. A coloring proof of a generalisation of Fermat's Little Theorem. *The American Mathematical Monthly*, 93(6):469–471, 1986.
15. Wikipedia: Proofs of Fermat's Little Theorem. `http://en.wikipedia.org/wiki/Proofs_of_Fermat's_little_theorem`.

[5] This proof is by Laurent Théry, and is apparently itself a translation of a Coq proof by J. C. Almeida.