A Case Study in Basic Algebra

Clemens Ballarin

Abstract

The focus of this case study is re-use in abstract algebra. It contains locale-based formalisations of selected parts of set, group and ring theory from Jacobson's *Basic Algebra* leading to the respective fundamental homomorphism theorems. The study is not intended as a library base for abstract algebra. It rather explores an approach towards abstract algebra in Isabelle.

hide-const *map*
hide-const *partition*

no-notation *divide* (**infixl** ‹′/› *70*)
no-notation *inverse-divide* (**infixl** ‹′/› *70*)

Each statement in the formal text is annotated with the location of the originating statement in Jacobson's text [1]. Each fact that Jacobson states explicitly is marked as **theorem** unless it is translated to a **sublocale** declaration. Literal quotations from Jacobson's text are reproduced in double quotes.

Auxiliary results needed for the formalisation that cannot be found in Jacobson's text are marked as **lemma** or are **interpretation**s. Such results are annotated with the location of a related statement. For example, the introduction rule of a constant is annotated with the location of the definition of the corresponding operation.

1 Concepts from Set Theory. The Integers

1.1 The Cartesian Product Set. Maps

Maps as extensional HOL functions

p 5, ll 21–25

locale *map* =
  fixes α **and** S **and** T
  assumes *graph* [*intro*, *simp*]: $\alpha \in S \rightarrow_E T$
begin

p 5, ll 21–25

lemma *map-closed* [*intro*, *simp*]:
  $a \in S \Longrightarrow \alpha\ a \in T$
⟨*proof*⟩

p 5, ll 21–25

lemma *map-undefined* [*intro*]:
  $a \notin S \Longrightarrow \alpha\ a = \mathit{undefined}$
⟨*proof*⟩

end

p 7, ll 7–8

locale *surjective-map* = *map* + **assumes** *surjective* [*intro*]: α ‘ $S = T$

p 7, ll 8–9

locale *injective-map* = *map* + **assumes** *injective* [*intro*, *simp*]: *inj-on* α S

Enables locale reasoning about the inverse *restrict* (*inv-into* S α) T of α.

p 7, ll 9–10

locale *bijective* =
  fixes α **and** S **and** T
  assumes *bijective* [*intro*, *simp*]: *bij-betw* α S T

Exploit existing knowledge about *bij-betw* rather than extending *surjective-map* and *injective-map*.

p 7, ll 9–10

locale *bijective-map* = *map* + *bijective* **begin**

p 7, ll 9–10

sublocale *surjective-map* ⟨*proof*⟩

p 7, ll 9–10

sublocale *injective-map* ⟨*proof*⟩

p 9, ll 12–13

sublocale *inverse*: *map restrict* (*inv-into* S α) T T S
  ⟨*proof*⟩

p 9, ll 12–13

sublocale *inverse*: *bijective restrict* (*inv-into* S α) T T S
  ⟨*proof*⟩

end

p 8, ll 14–15

abbreviation *identity* $S \equiv (\lambda x \in S.\ x)$

context *map* **begin**

p 8, ll 18–20; p 9, ll 1–8

theorem *bij-betw-iff-has-inverse*:
bij-betw $\alpha\ S\ T \longleftrightarrow (\exists\beta \in T \to_E S.\ \textit{compose}\ S\ \beta\ \alpha = \textit{identity}\ S \wedge \textit{compose}\ T\ \alpha\ \beta = \textit{identity}\ T)$
(**is** - $\longleftrightarrow (\exists\beta \in T \to_E S.\ ?INV\ \beta)$)
⟨*proof*⟩

end

1.2 Equivalence Relations. Factoring a Map Through an Equivalence Relation

p 11, ll 6–11

locale *equivalence* =
fixes S **and** E
assumes *closed* [*intro*, *simp*]: $E \subseteq S \times S$
and *reflexive* [*intro*, *simp*]: $a \in S \Longrightarrow (a, a) \in E$
and *symmetric* [*sym*]: $(a, b) \in E \Longrightarrow (b, a) \in E$
and *transitive* [*trans*]: $[\![(a, b) \in E; (b, c) \in E]\!] \Longrightarrow (a, c) \in E$
begin

p 11, ll 6–11

lemma *left-closed* [*intro*]:
$(a, b) \in E \Longrightarrow a \in S$
⟨*proof*⟩

p 11, ll 6–11

lemma *right-closed* [*intro*]:
$(a, b) \in E \Longrightarrow b \in S$
⟨*proof*⟩

end

p 11, ll 16–20

locale *partition* =
fixes S **and** P
assumes *subset*: $P \subseteq Pow\ S$
and *non-vacuous*: $\{\} \notin P$
and *complete*: $\bigcup P = S$
and *disjoint*: $[\![A \in P; B \in P; A \neq B]\!] \Longrightarrow A \cap B = \{\}$

context *equivalence* **begin**

p 11, ll 24–26

definition *Class* = $(\lambda a \in S.\ \{b \in S.\ (b, a) \in E\})$

p 11, ll 24–26

lemma *Class-closed* [*dest*]:
$[\![\ a \in Class\ b;\ b \in S\ ]\!] \Longrightarrow a \in S$
⟨*proof*⟩

p 11, ll 24–26

lemma *Class-closed2* [*intro*, *simp*]:
$a \in S \Longrightarrow Class\ a \subseteq S$
⟨*proof*⟩

p 11, ll 24–26

lemma *Class-undefined* [*intro*, *simp*]:
$a \notin S \Longrightarrow Class\ a = undefined$
⟨*proof*⟩

p 11, ll 24–26

lemma *ClassI* [*intro*, *simp*]:
$(a, b) \in E \Longrightarrow a \in Class\ b$
⟨*proof*⟩

p 11, ll 24–26

lemma *Class-revI* [*intro*, *simp*]:
$(a, b) \in E \Longrightarrow b \in Class\ a$
⟨*proof*⟩

p 11, ll 24–26

lemma *ClassD* [*dest*]:
$[\![\ b \in Class\ a;\ a \in S\ ]\!] \Longrightarrow (b, a) \in E$
⟨*proof*⟩

p 11, ll 30–31

theorem *Class-self* [*intro*, *simp*]:
$a \in S \Longrightarrow a \in Class\ a$
⟨*proof*⟩

p 11, l 31; p 12, l 1

theorem *Class-Union* [*simp*]:
$(\bigcup a{\in}S.\ Class\ a) = S$
⟨*proof*⟩

p 11, ll 2–3

theorem *Class-subset*:
$(a, b) \in E \Longrightarrow Class\ a \subseteq Class\ b$
⟨*proof*⟩

p 11, ll 3–4

theorem *Class-eq*:
$(a, b) \in E \Longrightarrow Class\ a = Class\ b$
⟨*proof*⟩

p 12, ll 1–5

theorem *Class-equivalence*:
$[\![a \in S;\ b \in S]\!] \Longrightarrow Class\ a = Class\ b \longleftrightarrow (a, b) \in E$
⟨*proof*⟩

p 12, ll 5–7

theorem *not-disjoint-implies-equal*:
assumes *not-disjoint*: $Class\ a \cap Class\ b \neq \{\}$
assumes *closed*: $a \in S\ b \in S$
shows $Class\ a = Class\ b$
⟨*proof*⟩

p 12, ll 7–8

definition $Partition = Class\ `\ S$

p 12, ll 7–8

lemma *Class-in-Partition* [*intro*, *simp*]:
$a \in S \Longrightarrow Class\ a \in Partition$
⟨*proof*⟩

p 12, ll 7–8

theorem *partition*:
partition S Partition
⟨*proof*⟩

end

context *partition* **begin**

p 12, ll 9–10

theorem *block-exists*:
$a \in S \Longrightarrow \exists A.\ a \in A \wedge A \in P$
⟨*proof*⟩

p 12, ll 9–10

theorem *block-unique*:
$[\![a \in A;\ A \in P;\ a \in B;\ B \in P]\!] \Longrightarrow A = B$
⟨*proof*⟩

p 12, ll 9–10

lemma *block-closed* [*intro*]:
$[\![a \in A;\ A \in P]\!] \Longrightarrow a \in S$

$\langle proof \rangle$

p 12, ll 9–10

lemma *element-exists*:
$A \in P \Longrightarrow \exists\, a \in S.\ a \in A$
$\langle proof \rangle$

p 12, ll 9–10

definition $Block = (\lambda a \in S.\ THE\ A.\ a \in A \wedge A \in P)$

p 12, ll 9–10

lemma *Block-closed* [*intro*, *simp*]:
assumes [*intro*]: $a \in S$ **shows** $Block\ a \in P$
$\langle proof \rangle$

p 12, ll 9–10

lemma *Block-undefined* [*intro*, *simp*]:
$a \notin S \Longrightarrow Block\ a = undefined$
$\langle proof \rangle$

p 12, ll 9–10

lemma *Block-self*:
$[\![\ a \in A;\ A \in P\ ]\!] \Longrightarrow Block\ a = A$
$\langle proof \rangle$

p 12, ll 10–11

definition $Equivalence = \{(a,\ b)\ .\ \exists\, A \in P.\ a \in A \wedge b \in A\}$

p 12, ll 11–12

theorem *equivalence*: *equivalence S Equivalence*
$\langle proof \rangle$

Temporarily introduce equivalence associated to partition.

p 12, ll 12–14

interpretation *equivalence S Equivalence* $\langle proof \rangle$

p 12, ll 12–14

theorem *Class-is-Block*:
assumes $a \in S$ **shows** $Class\ a = Block\ a$
$\langle proof \rangle$

p 12, l 14

lemma *Class-equals-Block*:
$Class = Block$
$\langle proof \rangle$

p 12, l 14

theorem *partition-of-equivalence*:
Partition = *P*
⟨*proof*⟩

end

context *equivalence* **begin**

p 12, ll 14–17

interpretation *partition S Partition* ⟨*proof*⟩

p 12, ll 14–17

theorem *equivalence-of-partition*:
Equivalence = *E*
⟨*proof*⟩

end

p 12, l 14

sublocale *partition* ⊆ *equivalence S Equivalence*
rewrites *equivalence.Partition S Equivalence* = *P* **and** *equivalence.Class S Equivalence* = *Block*
⟨*proof*⟩

p 12, ll 14–17

sublocale *equivalence* ⊆ *partition S Partition*
rewrites *partition.Equivalence Partition* = *E* **and** *partition.Block S Partition* = *Class*
⟨*proof*⟩

Unfortunately only effective on input

p 12, ll 18–20

notation *equivalence.Partition* (**infixl** ‹′/› *75*)

context *equivalence* **begin**

p 12, ll 18–20

lemma *representant-exists* [*dest*]: $A \in S\ /\ E \Longrightarrow \exists\, a{\in}S.\ a \in A \wedge A = \mathit{Class}\ a$
⟨*proof*⟩

p 12, ll 18–20

lemma *quotient-ClassE*: $A \in S\ /\ E \Longrightarrow (\bigwedge a.\ a \in S \Longrightarrow P\ (\mathit{Class}\ a)) \Longrightarrow P\ A$
⟨*proof*⟩

end

p 12, ll 21–23

sublocale *equivalence* ⊆ *natural*: *surjective-map Class S S / E*
⟨*proof*⟩

Technical device to achieve Jacobson's syntax; context where α is not a parameter.

p 12, ll 25–26

locale *fiber-relation-notation* = **fixes** *S* :: *'a set* **begin**

p 12, ll 25–26

definition *Fiber-Relation* (‹*E*'(-')›) **where** *Fiber-Relation* α = {(*x*, *y*). *x* ∈ *S* ∧ *y* ∈ *S* ∧ α *x* = α *y*}

end

Context where classes and the induced map are defined through the fiber relation. This will be the case for monoid homomorphisms but not group homomorphisms.

Avoids infinite interpretation chain.

p 12, ll 25–26

locale *fiber-relation* = *map* **begin**

Install syntax

p 12, ll 25–26

sublocale *fiber-relation-notation* ⟨*proof*⟩

p 12, ll 26–27

sublocale *equivalence* **where** *E* = *E*(α)
⟨*proof*⟩

"define $\bar{\alpha}$ by $\bar{\alpha}(\bar{a}) = \alpha(a)$"

p 13, ll 8–9

definition *induced* = (λ*A* ∈ *S* / *E*(α). *THE b*. ∃ *a* ∈ *A*. *b* = α *a*)

p 13, l 10

theorem *Fiber-equality*:
⟦ *a* ∈ *S*; *b* ∈ *S* ⟧ ⟹ *Class a* = *Class b* ⟷ α *a* = α *b*
⟨*proof*⟩

p 13, ll 8–9

theorem *induced-Fiber-simp* [*simp*]:
assumes [*intro*, *simp*]: *a* ∈ *S* **shows** *induced* (*Class a*) = α *a*

⟨*proof*⟩

p 13, ll 10–11

interpretation *induced*: *map induced S / E*(α) *T*
⟨*proof*⟩

p 13, ll 12–13

sublocale *induced*: *injective-map induced S / E*(α) *T*
⟨*proof*⟩

p 13, ll 16–19

theorem *factorization-lemma*:
  $a \in S \Longrightarrow$ *compose S induced Class* $a = \alpha\ a$
  ⟨*proof*⟩

p 13, ll 16–19

theorem *factorization* [*simp*]: *compose S induced Class* $= \alpha$
  ⟨*proof*⟩

p 14, ll 2–4

theorem *uniqueness*:
  assumes *map*: $\beta \in S\ /\ E(\alpha) \rightarrow_E T$
    and *factorization*: *compose S* β *Class* $= \alpha$
  shows $\beta =$ *induced*
⟨*proof*⟩

end

hide-const *monoid*
hide-const *group*
hide-const *inverse*

no-notation *quotient* (**infixl** ‹'/'/› *90*)

2 Monoids and Groups

2.1 Monoids of Transformations and Abstract Monoids

Def 1.1

p 28, ll 28–30

locale *monoid* =
  fixes *M* **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›)
  assumes *composition-closed* [*intro*, *simp*]: $[\![\ a \in M;\ b \in M\ ]\!] \Longrightarrow a \cdot b \in M$
    and *unit-closed* [*intro*, *simp*]: $\mathbf{1} \in M$
    and *associative* [*intro*]: $[\![\ a \in M;\ b \in M;\ c \in M\ ]\!] \Longrightarrow (a \cdot b) \cdot c = a \cdot (b \cdot c)$

and *left-unit* [*intro*, *simp*]: $a \in M \Longrightarrow \mathbf{1} \cdot a = a$
and *right-unit* [*intro*, *simp*]: $a \in M \Longrightarrow a \cdot \mathbf{1} = a$

p 29, ll 27–28

locale *submonoid* = *monoid M* (·) **1**
for *N* **and** *M* **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›) +
assumes *subset*: $N \subseteq M$
and *sub-composition-closed*: $[\![\, a \in N;\ b \in N \,]\!] \Longrightarrow a \cdot b \in N$
and *sub-unit-closed*: $\mathbf{1} \in N$
begin

p 29, ll 27–28

lemma *sub* [*intro*, *simp*]:
$a \in N \Longrightarrow a \in M$
⟨*proof*⟩

p 29, ll 32–33

sublocale *sub*: *monoid N* (·) **1**
⟨*proof*⟩

end

p 29, ll 33–34

theorem *submonoid-transitive*:
assumes *submonoid K N composition unit*
and *submonoid N M composition unit*
shows *submonoid K M composition unit*
⟨*proof*⟩

p 28, l 23

locale *transformations* =
fixes *S* :: *'a set*

Monoid of all transformations

p 28, ll 23–24

sublocale *transformations* ⊆ *monoid* $S \rightarrow_E S$ *compose S identity S*
⟨*proof*⟩

N is a monoid of transformations of the set S.

p 29, ll 34–36

locale *transformation-monoid* =
transformations S + *submonoid M* $S \rightarrow_E S$ *compose S identity S* **for** *M* **and** *S*
begin

p 29, ll 34–36

lemma *transformation-closed* [*intro*, *simp*]:

$[\![\alpha \in M; x \in S]\!] \Longrightarrow \alpha\ x \in S$
⟨*proof*⟩

p 29, ll 34–36

lemma *transformation-undefined* [*intro*, *simp*]:
$[\![\alpha \in M; x \notin S]\!] \Longrightarrow \alpha\ x = \mathit{undefined}$
⟨*proof*⟩

end

2.2 Groups of Transformations and Abstract Groups

context *monoid* **begin**

Invertible elements

p 31, ll 3–5

definition *invertible* **where** $u \in M \Longrightarrow \mathit{invertible}\ u \longleftrightarrow (\exists\, v \in M.\ u \cdot v = \mathbf{1} \wedge v \cdot u = \mathbf{1})$

p 31, ll 3–5

lemma *invertibleI* [*intro*]:
$[\![u \cdot v = \mathbf{1}; v \cdot u = \mathbf{1}; u \in M; v \in M]\!] \Longrightarrow \mathit{invertible}\ u$
⟨*proof*⟩

p 31, ll 3–5

lemma *invertibleE* [*elim*]:
$[\![\mathit{invertible}\ u; \bigwedge v.\ [\![u \cdot v = \mathbf{1} \wedge v \cdot u = \mathbf{1}; v \in M]\!] \Longrightarrow P; u \in M]\!] \Longrightarrow P$
⟨*proof*⟩

p 31, ll 6–7

theorem *inverse-unique*:
$[\![u \cdot v' = \mathbf{1}; v \cdot u = \mathbf{1}; u \in M;\ \ v \in M; v' \in M]\!] \Longrightarrow v = v'$
⟨*proof*⟩

p 31, l 7

definition *inverse* **where** $\mathit{inverse} = (\lambda u \in M.\ \mathit{THE}\ v.\ v \in M \wedge u \cdot v = \mathbf{1} \wedge v \cdot u = \mathbf{1})$

p 31, l 7

theorem *inverse-equality*:
$[\![u \cdot v = \mathbf{1}; v \cdot u = \mathbf{1}; u \in M; v \in M]\!] \Longrightarrow \mathit{inverse}\ u = v$
⟨*proof*⟩

p 31, l 7

lemma *invertible-inverse-closed* [*intro*, *simp*]:
$[\![\mathit{invertible}\ u; u \in M]\!] \Longrightarrow \mathit{inverse}\ u \in M$

$\langle proof \rangle$

p 31, l 7

lemma *inverse-undefined* [*intro*, *simp*]:
$u \notin M \Longrightarrow$ *inverse* $u =$ *undefined*
$\langle proof \rangle$

p 31, l 7

lemma *invertible-left-inverse* [*simp*]:
$[\![$ *invertible* u; $u \in M$ $]\!] \Longrightarrow$ *inverse* $u \cdot u = \mathbf{1}$
$\langle proof \rangle$

p 31, l 7

lemma *invertible-right-inverse* [*simp*]:
$[\![$ *invertible* u; $u \in M$ $]\!] \Longrightarrow u \cdot$ *inverse* $u = \mathbf{1}$
$\langle proof \rangle$

p 31, l 7

lemma *invertible-left-cancel* [*simp*]:
$[\![$ *invertible* x; $x \in M$; $y \in M$; $z \in M$ $]\!] \Longrightarrow x \cdot y = x \cdot z \longleftrightarrow y = z$
$\langle proof \rangle$

p 31, l 7

lemma *invertible-right-cancel* [*simp*]:
$[\![$ *invertible* x; $x \in M$; $y \in M$; $z \in M$ $]\!] \Longrightarrow y \cdot x = z \cdot x \longleftrightarrow y = z$
$\langle proof \rangle$

p 31, l 7

lemma *inverse-unit* [*simp*]: *inverse* $\mathbf{1} = \mathbf{1}$
$\langle proof \rangle$

p 31, ll 7–8

theorem *invertible-inverse-invertible* [*intro*, *simp*]:
$[\![$ *invertible* u; $u \in M$ $]\!] \Longrightarrow$ *invertible* (*inverse* u)
$\langle proof \rangle$

p 31, l 8

theorem *invertible-inverse-inverse* [*simp*]:
$[\![$ *invertible* u; $u \in M$ $]\!] \Longrightarrow$ *inverse* (*inverse* u) $= u$
$\langle proof \rangle$

end

context *submonoid* **begin**

Reasoning about *invertible* and *inverse* in submonoids.

p 31, l 7

lemma *submonoid-invertible* [*intro*, *simp*]:
 $\llbracket$ *sub.invertible u*; $u \in N$ $\rrbracket \Longrightarrow$ *invertible u*
 ⟨*proof*⟩

p 31, l 7

lemma *submonoid-inverse-closed* [*intro*, *simp*]:
 $\llbracket$ *sub.invertible u*; $u \in N$ $\rrbracket \Longrightarrow$ *inverse* $u \in N$
 ⟨*proof*⟩

end

Def 1.2

p 31, ll 9–10

locale *group* =
 monoid G (·) **1** **for** *G* **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›) +
 assumes *invertible* [*simp*, *intro*]: $u \in G \Longrightarrow$ *invertible u*

p 31, ll 11–12

locale *subgroup* = *submonoid G M* (·) **1** + *sub*: *group G* (·) **1**
 for *G* **and** *M* **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›)
begin

Reasoning about *invertible* and *inverse* in subgroups.

p 31, ll 11–12

lemma *subgroup-inverse-equality* [*simp*]:
 $u \in G \Longrightarrow$ *inverse u* = *sub.inverse u*
 ⟨*proof*⟩

p 31, ll 11–12

lemma *subgroup-inverse-iff* [*simp*]:
 $\llbracket$ *invertible x*; $x \in M$ $\rrbracket \Longrightarrow$ *inverse* $x \in G \longleftrightarrow x \in G$
 ⟨*proof*⟩

end

p 31, ll 11–12

lemma *subgroup-transitive* [*trans*]:
 assumes *subgroup K H composition unit*
 and *subgroup H G composition unit*
 shows *subgroup K G composition unit*
⟨*proof*⟩

context *monoid* **begin**

Jacobson states both directions, but the other one is trivial.

p 31, ll 12–15

theorem *subgroupI*:
  fixes *G*
  assumes *subset* [*THEN subsetD*, *intro*]: $G \subseteq M$
    and [*intro*]: $\mathbf{1} \in G$
    and [*intro*]: $\bigwedge g\ h.\ [\![\ g \in G;\ h \in G\ ]\!] \Longrightarrow g \cdot h \in G$
    and [*intro*]: $\bigwedge g.\ g \in G \Longrightarrow$ *invertible* g
    and [*intro*]: $\bigwedge g.\ g \in G \Longrightarrow$ *inverse* $g \in G$
  shows *subgroup* $G\ M\ (\cdot)\ \mathbf{1}$
⟨*proof*⟩

p 31, l 16

definition *Units* = $\{u \in M.$ *invertible* $u\}$

p 31, l 16

lemma *mem-UnitsI*:
  $[\![$ *invertible* u; $u \in M\ ]\!] \Longrightarrow u \in$ *Units*
  ⟨*proof*⟩

p 31, l 16

lemma *mem-UnitsD*:
  $[\![\ u \in$ *Units* $]\!] \Longrightarrow$ *invertible* $u \wedge u \in M$
  ⟨*proof*⟩

p 31, ll 16–21

interpretation *units*: *subgroup Units M*
⟨*proof*⟩

p 31, ll 21–22

theorem *group-of-Units* [*intro*, *simp*]:
  group Units $(\cdot)\ \mathbf{1}$
  ⟨*proof*⟩

p 31, l 19

lemma *composition-invertible* [*simp*, *intro*]:
  $[\![$ *invertible* x; *invertible* y; $x \in M$; $y \in M\ ]\!] \Longrightarrow$ *invertible* $(x \cdot y)$
  ⟨*proof*⟩

p 31, l 20

lemma *unit-invertible*:
  invertible $\mathbf{1}$
  ⟨*proof*⟩

Useful simplification rules

p 31, l 22

lemma *invertible-right-inverse2*:
  $[\![$ *invertible* u; $u \in M$; $v \in M\ ]\!] \Longrightarrow u \cdot ($*inverse* $u \cdot v) = v$

⟨*proof*⟩

p 31, l 22

lemma *invertible-left-inverse2*:
$[\![$ *invertible* u; $u \in M$; $v \in M$ $]\!] \Longrightarrow$ *inverse* $u \cdot (u \cdot v) = v$
⟨*proof*⟩

p 31, l 22

lemma *inverse-composition-commute*:
assumes [*simp*]: *invertible* x *invertible* y $x \in M$ $y \in M$
shows *inverse* $(x \cdot y)$ = *inverse* $y \cdot$ *inverse* x
⟨*proof*⟩

end

p 31, l 24

context *transformations* **begin**

p 31, ll 25–26

theorem *invertible-is-bijective*:
assumes *dom*: $\alpha \in S \rightarrow_E S$
shows *invertible* $\alpha \longleftrightarrow$ *bij-betw* α S S
⟨*proof*⟩

p 31, ll 26–27

theorem *Units-bijective*:
Units = $\{\alpha \in S \rightarrow_E S.$ *bij-betw* α S $S\}$
⟨*proof*⟩

p 31, ll 26–27

lemma *Units-bij-betwI* [*intro*, *simp*]:
$\alpha \in$ *Units* $\Longrightarrow$ *bij-betw* α S S
⟨*proof*⟩

p 31, ll 26–27

lemma *Units-bij-betwD* [*dest*, *simp*]:
$[\![$ $\alpha \in S \rightarrow_E S$; *bij-betw* α S S $]\!] \Longrightarrow \alpha \in$ *Units*
⟨*proof*⟩

p 31, ll 28–29

abbreviation *Sym* $\equiv$ *Units*

p 31, ll 26–28

sublocale *symmetric*: *group Sym compose S identity S*
⟨*proof*⟩

end

p 32, ll 18–19

locale *transformation-group* =
  transformations S + *symmetric*: *subgroup G Sym compose S identity S* **for** *G* **and** *S*
begin

p 32, ll 18–19

lemma *transformation-group-closed* [*intro*, *simp*]:
  $[\![\alpha \in G; x \in S]\!] \Longrightarrow \alpha\ x \in S$
  ⟨*proof*⟩

p 32, ll 18–19

lemma *transformation-group-undefined* [*intro*, *simp*]:
  $[\![\alpha \in G; x \notin S]\!] \Longrightarrow \alpha\ x = \mathit{undefined}$
  ⟨*proof*⟩

end

2.3 Isomorphisms. Cayley's Theorem

Def 1.3

p 37, ll 7–11

locale *monoid-isomorphism* =
  bijective-map η M M' + *source*: *monoid* M $(\cdot)$ $\mathbf{1}$ + *target*: *monoid* M' $(\cdot')$ $\mathbf{1}'$
  for η **and** M **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›)
    and M' **and** *composition'* (**infixl** ‹·''› *70*) **and** *unit'* (‹**1''**›) +
  assumes *commutes-with-composition*: $[\![x \in M; y \in M]\!] \Longrightarrow \eta\ x \cdot' \eta\ y = \eta\ (x \cdot y)$
    and *commutes-with-unit*: $\eta\ \mathbf{1} = \mathbf{1}'$

p 37, l 10

definition *isomorphic-as-monoids* (**infixl** ‹$\cong_M$› *50*)
  where $\mathcal{M} \cong_M \mathcal{M}' \longleftrightarrow$ (*let* (M, *composition*, *unit*) = $\mathcal{M}$; (M', *composition'*, *unit'*) = $\mathcal{M}'$ *in*
  ($\exists\,\eta$. *monoid-isomorphism* η M *composition unit* M' *composition' unit'*))

p 37, ll 11–12

locale *monoid-isomorphism'* =
  bijective-map η M M' + *source*: *monoid* M $(\cdot)$ $\mathbf{1}$ + *target*: *monoid* M' $(\cdot')$ $\mathbf{1}'$
  for η **and** M **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›)
    and M' **and** *composition'* (**infixl** ‹·''› *70*) **and** *unit'* (‹**1''**›) +
  assumes *commutes-with-composition*: $[\![x \in M; y \in M]\!] \Longrightarrow \eta\ x \cdot' \eta\ y = \eta\ (x \cdot y)$

p 37, ll 11–12

sublocale *monoid-isomorphism* ⊆ *monoid-isomorphism'*
  ⟨*proof*⟩

Both definitions are equivalent.

p 37, ll 12–15

sublocale *monoid-isomorphism'* $\subseteq$ *monoid-isomorphism*
⟨*proof*⟩

context *monoid-isomorphism* **begin**

p 37, ll 30–33

theorem *inverse-monoid-isomorphism*:
monoid-isomorphism (*restrict* (*inv-into* M η) M') M' $(\cdot')$ $\mathbf{1}'$ M $(\cdot)$ $\mathbf{1}$
⟨*proof*⟩

end

We only need that η is symmetric.

p 37, ll 28–29

theorem *isomorphic-as-monoids-symmetric*:
(M, *composition*, *unit*) $\cong_M$ (M', *composition'*, *unit'*) $\Longrightarrow$ (M', *composition'*, *unit'*) $\cong_M$ (M, *composition*, *unit*)
⟨*proof*⟩

p 38, l 4

locale *left-translations-of-monoid* = *monoid* **begin**

p 38, ll 5–7

definition *translation* (‹'(-')$_L$›) **where** *translation* = $(\lambda a \in M.\ \lambda x \in M.\ a \cdot x)$

p 38, ll 5–7

lemma *translation-map* [*intro*, *simp*]:
$a \in M \Longrightarrow (a)_L \in M \rightarrow_E M$
⟨*proof*⟩

p 38, ll 5–7

lemma *Translations-maps* [*intro*, *simp*]:
translation ‘ $M \subseteq M \rightarrow_E M$
⟨*proof*⟩

p 38, ll 5–7

lemma *translation-apply*:
$[\![\ a \in M;\ b \in M\ ]\!] \Longrightarrow (a)_L\ b = a \cdot b$
⟨*proof*⟩

p 38, ll 5–7

lemma *translation-exist*:
$f \in$ *translation* ‘ $M \Longrightarrow \exists\, a \in M.\ f = (a)_L$
⟨*proof*⟩

p 38, ll 5–7

lemmas *Translations-E* [*elim*] = *translation-exist* [*THEN bexE*]

p 38, l 10

theorem *translation-unit-eq* [*simp*]:
identity $M = (\mathbf{1})_L$
⟨*proof*⟩

p 38, ll 10–11

theorem *translation-composition-eq* [*simp*]:
assumes [*simp*]: $a \in M$ $b \in M$
shows *compose* M $(a)_L$ $(b)_L = (a \cdot b)_L$
⟨*proof*⟩

p 38, ll 7–9

sublocale *transformation*: *transformations M* ⟨*proof*⟩

p 38, ll 7–9

theorem *Translations-transformation-monoid*:
transformation-monoid (*translation* ‘ *M*) *M*
⟨*proof*⟩

p 38, ll 7–9

sublocale *transformation*: *transformation-monoid translation* ‘ *M M*
⟨*proof*⟩

p 38, l 12

sublocale *map translation M translation* ‘ *M*
⟨*proof*⟩

p 38, ll 12–16

theorem *translation-isomorphism* [*intro*]:
monoid-isomorphism translation M (·) **1** (*translation* ‘ *M*) (*compose M*) (*identity M*)
⟨*proof*⟩

p 38, ll 12–16

sublocale *monoid-isomorphism translation M* (·) **1** *translation* ‘ *M compose M identity M* ⟨*proof*⟩

end

context *monoid* **begin**

p 38, ll 1–2

interpretation *left-translations-of-monoid* ⟨*proof*⟩

p 38, ll 1–2

theorem *cayley-monoid*:
 $\exists\, M'$ *composition′ unit′*. *transformation-monoid* M' $M \wedge (M, (\cdot), \mathbf{1}) \cong_M (M', \mathit{composition}', \mathit{unit}')$
 ⟨*proof*⟩

end

p 38, l 17

locale *left-translations-of-group* = *group* **begin**

p 38, ll 17–18

sublocale *left-translations-of-monoid* **where** $M = G$ ⟨*proof*⟩

p 38, ll 17–18

notation *translation* (‹′(-′)$_L$›)

The group of left translations is a subgroup of the symmetric group, hence *transformation.sub.invertible*.

p 38, ll 20–22

theorem *translation-invertible* [*intro*, *simp*]:
 assumes [*simp*]: $a \in G$
 shows *transformation.sub.invertible* $(a)_L$
⟨*proof*⟩

p 38, ll 19–20

theorem *translation-bijective* [*intro*, *simp*]:
 $a \in G \Longrightarrow$ *bij-betw* $(a)_L$ G G
 ⟨*proof*⟩

p 38, ll 18–20

theorem *Translations-transformation-group*:
 transformation-group (*translation* ‘ G) G
⟨*proof*⟩

p 38, ll 18–20

sublocale *transformation*: *transformation-group translation* ‘ G G
 ⟨*proof*⟩

end

context *group* **begin**

p 38, ll 2–3

interpretation *left-translations-of-group* ⟨*proof*⟩

p 38, ll 2–3

theorem *cayley-group*:
$\exists\, G'$ *composition′ unit′*. *transformation-group* G' $G \wedge (G, (\cdot), \mathbf{1}) \cong_M (G'$, *composition′*, *unit′*)
⟨*proof*⟩

end

Exercise 3

p 39, ll 9–10

locale *right-translations-of-group* = *group* **begin**

p 39, ll 9–10

definition *translation* $(\langle'(\text{-}')_R\rangle)$ **where** *translation* $= (\lambda a \in G.\ \lambda x \in G.\ x \cdot a)$

p 39, ll 9–10

abbreviation *Translations* ≡ *translation* ‘ G

The isomorphism that will be established is a map different from *translation*.

p 39, ll 9–10

interpretation *aux*: *map translation G Translations*
⟨*proof*⟩

p 39, ll 9–10

lemma *translation-map* [*intro*, *simp*]:
$a \in G \Longrightarrow (a)_R \in G \rightarrow_E G$
⟨*proof*⟩

p 39, ll 9–10

lemma *Translation-maps* [*intro*, *simp*]:
Translations $\subseteq G \rightarrow_E G$
⟨*proof*⟩

p 39, ll 9–10

lemma *translation-apply*:
$[\![\ a \in G;\ b \in G\ ]\!] \Longrightarrow (a)_R\ b = b \cdot a$
⟨*proof*⟩

p 39, ll 9–10

lemma *translation-exist*:
$f \in$ *Translations* $\Longrightarrow \exists\, a \in G.\ f = (a)_R$
⟨*proof*⟩

p 39, ll 9–10

lemmas *Translations-E* [*elim*] = *translation-exist* [*THEN bexE*]

p 39, ll 9–10

lemma *translation-unit-eq* [*simp*]:
identity $G = (\mathbf{1})_R$
⟨*proof*⟩

p 39, ll 10–11

lemma *translation-composition-eq* [*simp*]:
assumes [*simp*]: $a \in G$ $b \in G$
shows *compose* G $(a)_R$ $(b)_R = (b \cdot a)_R$
⟨*proof*⟩

p 39, ll 10–11

sublocale *transformation*: *transformations* G ⟨*proof*⟩

p 39, ll 10–11

lemma *Translations-transformation-monoid*:
transformation-monoid Translations G
⟨*proof*⟩

p 39, ll 10–11

sublocale *transformation*: *transformation-monoid Translations* G
⟨*proof*⟩

p 39, ll 10–11

lemma *translation-invertible* [*intro*, *simp*]:
assumes [*simp*]: $a \in G$
shows *transformation.sub.invertible* $(a)_R$
⟨*proof*⟩

p 39, ll 10–11

lemma *translation-bijective* [*intro*, *simp*]:
$a \in G \Longrightarrow$ *bij-betw* $(a)_R$ G G
⟨*proof*⟩

p 39, ll 10–11

theorem *Translations-transformation-group*:
transformation-group Translations G
⟨*proof*⟩

p 39, ll 10–11

sublocale *transformation*: *transformation-group Translations* G
⟨*proof*⟩

p 39, ll 10–11

lemma *translation-inverse-eq* [*simp*]:
assumes [*simp*]: $a \in G$

shows *transformation.sub.inverse* $(a)_R = (inverse\ a)_R$
⟨*proof*⟩

p 39, ll 10–11

theorem *translation-inverse-monoid-isomorphism* [*intro*]:
monoid-isomorphism ($\lambda a{\in}G$. *transformation.symmetric.inverse* $(a)_R$) G $(\cdot)$ **1** *Translations* (*compose* G) (*identity* G)
(**is** *monoid-isomorphism ?inv* - - - - - -)
⟨*proof*⟩

p 39, ll 10–11

sublocale *monoid-isomorphism*
$\lambda a{\in}G$. *transformation.symmetric.inverse* $(a)_R$ G $(\cdot)$ **1** *Translations compose G identity G* ⟨*proof*⟩

end

2.4 Generalized Associativity. Commutativity

p 40, l 27; p 41, ll 1–2

locale *commutative-monoid* = *monoid* +
assumes *commutative*: $[\![\ x \in M;\ y \in M\ ]\!] \Longrightarrow x \cdot y = y \cdot x$

p 41, l 2

locale *abelian-group* = *group* + *commutative-monoid* G $(\cdot)$ **1**

2.5 Orbits. Cosets of a Subgroup

context *transformation-group* **begin**

p 51, ll 18–20

definition *Orbit-Relation*
where *Orbit-Relation* $= \{(x, y).\ x \in S \wedge y \in S \wedge (\exists\, \alpha \in G.\ y = \alpha\ x)\}$

p 51, ll 18–20

lemma *Orbit-Relation-memI* [*intro*]:
$[\![\ \exists\, \alpha \in G.\ y = \alpha\ x;\ x \in S\ ]\!] \Longrightarrow (x, y) \in$ *Orbit-Relation*
⟨*proof*⟩

p 51, ll 18–20

lemma *Orbit-Relation-memE* [*elim*]:
$[\![\ (x, y) \in$ *Orbit-Relation*; $\bigwedge\alpha.\ [\![\ \alpha \in G;\ x \in S;\ y = \alpha\ x\ ]\!] \Longrightarrow Q\ ]\!] \Longrightarrow Q$
⟨*proof*⟩

p 51, ll 20–23, 26–27

sublocale *orbit*: *equivalence S Orbit-Relation*
⟨*proof*⟩

p 51, ll 23–24

theorem *orbit-equality*:
$x \in S \Longrightarrow orbit.Class\ x = \{\alpha\ x \mid \alpha.\ \alpha \in G\}$
⟨*proof*⟩

end

context *monoid-isomorphism* **begin**

p 52, ll 16–17

theorem *image-subgroup*:
assumes *subgroup G M* (·) **1**
shows *subgroup* $(\eta\ `\ G)\ M'\ (\cdot')\ \mathbf{1}'$
⟨*proof*⟩

end

Technical device to achieve Jacobson's notation for *Right-Coset* and *Left-Coset*. The definitions are pulled out of *subgroup-of-group* to a context where H is not a parameter.

p 52, l 20

locale *coset-notation* = **fixes** *composition* (**infixl** ‹·› *70*) **begin**

Equation 23

p 52, l 20

definition *Right-Coset* (**infixl** ‹|·› *70*) **where** $H \mid\cdot\ x = \{h \cdot x \mid h.\ h \in H\}$

p 53, ll 8–9

definition *Left-Coset* (**infixl** ‹·|› *70*) **where** $x \cdot\mid H = \{x \cdot h \mid h.\ h \in H\}$

p 52, l 20

lemma *Right-Coset-memI* [*intro*]:
$h \in H \Longrightarrow h \cdot x \in H \mid\cdot\ x$
⟨*proof*⟩

p 52, l 20

lemma *Right-Coset-memE* [*elim*]:
$[\![\ a \in H \mid\cdot\ x;\ \bigwedge h.\ [\![\ h \in H;\ a = h \cdot x\ ]\!] \Longrightarrow P\ ]\!] \Longrightarrow P$
⟨*proof*⟩

p 53, ll 8–9

lemma *Left-Coset-memI* [*intro*]:
$h \in H \Longrightarrow x \cdot h \in x \cdot\mid H$
⟨*proof*⟩

p 53, ll 8–9

lemma *Left-Coset-memE* [*elim*]:
  $[\![\ a \in x \cdot|\ H;\ \bigwedge h.\ [\![\ h \in H;\ a = x \cdot h\ ]\!] \Longrightarrow P\ ]\!] \Longrightarrow P$
  ⟨*proof*⟩

end

p 52, l 12

locale *subgroup-of-group* = *subgroup H G* (·) **1** + *coset-notation* (·) + *group G* (·) **1**
  for *H* **and** *G* **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›)
begin

p 52, ll 12–14

interpretation *left*: *left-translations-of-group* ⟨*proof*⟩
interpretation *right*: *right-translations-of-group* ⟨*proof*⟩

left.translation ‘ *H* denotes Jacobson's $H_L(G)$ and *left.translation* ‘ *G* denotes Jacobson's G_L.

p 52, ll 16–18

theorem *left-translations-of-subgroup-are-transformation-group* [*intro*]:
  transformation-group (*left.translation* ‘ *H*) *G*
⟨*proof*⟩

p 52, l 18

interpretation *transformation-group left.translation* ‘ *H G* ⟨*proof*⟩

p 52, ll 19–20

theorem *Right-Coset-is-orbit*:
  $x \in G \Longrightarrow H\ |\cdot\ x = orbit.Class\ x$
  ⟨*proof*⟩

p 52, ll 24–25

theorem *Right-Coset-Union*:
  $(\bigcup x{\in}G.\ H\ |\cdot\ x) = G$
  ⟨*proof*⟩

p 52, l 26

theorem *Right-Coset-bij*:
  assumes *G* [*simp*]: $x \in G\ y \in G$
  shows *bij-betw* $(inverse\ x \cdot y)_R\ (H\ |\cdot\ x)\ (H\ |\cdot\ y)$
⟨*proof*⟩

p 52, ll 25–26

theorem *Right-Cosets-cardinality*:
  $[\![\ x \in G;\ y \in G\ ]\!] \Longrightarrow card\ (H\ |\cdot\ x) = card\ (H\ |\cdot\ y)$
  ⟨*proof*⟩

p 52, l 27

theorem *Right-Coset-unit*:
  $H \mid\cdot \mathbf{1} = H$
  ⟨*proof*⟩

p 52, l 27

theorem *Right-Coset-cardinality*:
  $x \in G \Longrightarrow card\ (H \mid\cdot x) = card\ H$
  ⟨*proof*⟩

p 52, ll 31–32

definition $index = card\ orbit.Partition$

Theorem 1.5

p 52, ll 33–35; p 53, ll 1–2

theorem *lagrange*:
  $finite\ G \Longrightarrow card\ G = card\ H * index$
  ⟨*proof*⟩

end

Left cosets

context *subgroup* **begin**

p 31, ll 11–12

lemma *image-of-inverse* [*intro*, *simp*]:
  $x \in G \Longrightarrow x \in inverse\ `\ G$
  ⟨*proof*⟩

end

context *group* **begin**

p 53, ll 6–7

lemma *inverse-subgroupI*:
  assumes *sub*: $subgroup\ H\ G\ (\cdot)\ \mathbf{1}$
  shows $subgroup\ (inverse\ `\ H)\ G\ (\cdot)\ \mathbf{1}$
⟨*proof*⟩

p 53, ll 6–7

lemma *inverse-subgroupD*:
  assumes *sub*: $subgroup\ (inverse\ `\ H)\ G\ (\cdot)\ \mathbf{1}$
    and *inv*: $H \subseteq Units$
  shows $subgroup\ H\ G\ (\cdot)\ \mathbf{1}$
⟨*proof*⟩

end

context *subgroup-of-group* **begin**

p 53, l 6

interpretation *right-translations-of-group* ⟨*proof*⟩

translation ‘ H denotes Jacobson's $H_R(G)$ and *Translations* denotes Jacobson's G_R.

p 53, ll 6–7

theorem *right-translations-of-subgroup-are-transformation-group* [*intro*]:
  transformation-group (*translation* ‘ H) G
⟨*proof*⟩

p 53, ll 6–7

interpretation *transformation-group translation* ‘ H G ⟨*proof*⟩

Equation 23 for left cosets

p 53, ll 7–8

theorem *Left-Coset-is-orbit*:
  $x \in G \Longrightarrow x \cdot| H = orbit.Class\ x$
  ⟨*proof*⟩

end

2.6 Congruences. Quotient Monoids and Groups

Def 1.4

p 54, ll 19–22

locale *monoid-congruence* = *monoid* + *equivalence* **where** $S = M$ +
  assumes *cong*: $[\![(a, a') \in E; (b, b') \in E]\!] \Longrightarrow (a \cdot b, a' \cdot b') \in E$
begin

p 54, ll 26–28

theorem *Class-cong*:
  $[\![Class\ a = Class\ a'; Class\ b = Class\ b'; a \in M; a' \in M; b \in M; b' \in M]\!] \Longrightarrow Class\ (a \cdot b) = Class\ (a' \cdot b')$
  ⟨*proof*⟩

p 54, ll 28–30

definition *quotient-composition* (**infixl** ‹[·]› *70*)
  where $quotient\text{-}composition = (\lambda A \in M\ /\ E.\ \lambda B \in M\ /\ E.\ THE\ C.\ \exists\, a \in A.\ \exists\, b \in B.\ C = Class\ (a \cdot b))$

p 54, ll 28–30

theorem *Class-commutes-with-composition*:

$\llbracket a \in M;\ b \in M \rrbracket \Longrightarrow$ *Class* a [·] *Class* b = *Class* $(a \cdot b)$
⟨*proof*⟩

p 54, ll 30–31

theorem *quotient-composition-closed* [*intro*, *simp*]:
$\llbracket A \in M\ /\ E;\ B \in M\ /\ E \rrbracket \Longrightarrow A$ [·] $B \in M\ /\ E$
⟨*proof*⟩

p 54, l 32; p 55, ll 1–3

sublocale *quotient*: *monoid* M / E ([·]) *Class* **1**
⟨*proof*⟩

end

p 55, ll 16–17

locale *group-congruence* = *group* + *monoid-congruence* **where** $M = G$ **begin**

p 55, ll 16–17

notation *quotient-composition* (**infixl** ‹[·]› *70*)

p 55, l 18

theorem *Class-right-inverse*:
$a \in G \Longrightarrow$ *Class* a [·] *Class* (*inverse* a) = *Class* **1**
⟨*proof*⟩

p 55, l 18

theorem *Class-left-inverse*:
$a \in G \Longrightarrow$ *Class* (*inverse* a) [·] *Class* a = *Class* **1**
⟨*proof*⟩

p 55, l 18

theorem *Class-invertible*:
$a \in G \Longrightarrow$ *quotient.invertible* (*Class* a)
⟨*proof*⟩

p 55, l 18

theorem *Class-commutes-with-inverse*:
$a \in G \Longrightarrow$ *quotient.inverse* (*Class* a) = *Class* (*inverse* a)
⟨*proof*⟩

p 55, l 17

sublocale *quotient*: *group* G / E ([·]) *Class* **1**
⟨*proof*⟩

end

Def 1.5

p 55, ll 22–25

locale *normal-subgroup* =
subgroup-of-group K G $(\cdot)$ **1** **for** K **and** G **and** *composition* (**infixl** ‹·› *70*) **and** *unit*
(‹**1**›) +
assumes *normal*: $[\![\ g \in G;\ k \in K\ ]\!] \Longrightarrow$ *inverse* $g \cdot k \cdot g \in K$

Lemmas from the proof of Thm 1.6

context *subgroup-of-group* **begin**

We use H for K.

p 56, ll 14–16

theorem *Left-equals-Right-coset-implies-normality*:
assumes [*simp*]: $\bigwedge g.\ g \in G \Longrightarrow g \cdot| H = H |\cdot g$
shows *normal-subgroup* H G $(\cdot)$ **1**
⟨*proof*⟩

end

Thm 1.6, first part

context *group-congruence* **begin**

Jacobson's K

p 56, l 29

definition *Normal* = *Class* **1**

p 56, ll 3–6

interpretation *subgroup Normal* G $(\cdot)$ **1**
⟨*proof*⟩

Coset notation

p 56, ll 5–6

interpretation *subgroup-of-group Normal* G $(\cdot)$ **1** ⟨*proof*⟩

Equation 25 for right cosets

p 55, ll 29–30; p 56, ll 6–11

theorem *Right-Coset-Class-unit*:
assumes *g*: $g \in G$ **shows** *Normal* $|\cdot\ g$ = *Class* g
⟨*proof*⟩

Equation 25 for left cosets

p 55, ll 29–30; p 56, ll 6–11

theorem *Left-Coset-Class-unit*:
assumes *g*: $g \in G$ **shows** $g\ \cdot|$ *Normal* = *Class* g

⟨*proof*⟩

Thm 1.6, statement of first part

p 55, ll 28–29; p 56, ll 12–16

theorem *Class-unit-is-normal*:
normal-subgroup Normal G (·) **1**
⟨*proof*⟩

sublocale *normal*: *normal-subgroup Normal G* (·) **1**
⟨*proof*⟩

end

context *normal-subgroup* **begin**

p 56, ll 16–19

theorem *Left-equals-Right-coset*:
$g \in G \Longrightarrow g \cdot| K = K |\cdot g$
⟨*proof*⟩

Thm 1.6, second part

p 55, ll 31–32; p 56, ll 20–21

definition $Congruence = \{(a, b).\ a \in G \wedge b \in G \wedge inverse\ a \cdot b \in K\}$

p 56, ll 21–22

interpretation *right-translations-of-group* ⟨*proof*⟩

p 56, ll 21–22

interpretation *transformation-group translation* ‘ *K G* **rewrites** *Orbit-Relation* = *Congruence*
⟨*proof*⟩

p 56, ll 20–21

lemma *CongruenceI*: $[\![a = b \cdot k; a \in G; b \in G; k \in K]\!] \Longrightarrow (a, b) \in Congruence$
⟨*proof*⟩

p 56, ll 20–21

lemma *CongruenceD*: $(a, b) \in Congruence \Longrightarrow \exists k{\in}K.\ a = b \cdot k$
⟨*proof*⟩

“We showed in the last section that the relation we are considering is an equivalence relation in G for any subgroup K of G. We now proceed to show that normality of K ensures that [...] $a \equiv b \pmod K$ is a congruence.”

p 55, ll 30–32; p 56, ll 1, 22–28

sublocale *group-congruence* **where** $E = Congruence$ **rewrites** $Normal = K$

⟨*proof*⟩

end

context *group* **begin**

Pulled out of *normal-subgroup* to achieve standard notation.

p 56, ll 31–32

abbreviation *Factor-Group* (**infixl** ‹'/'/› *75*)
  where S // $K \equiv S$ / (*normal-subgroup.Congruence* K G $(\cdot)$ $\mathbf{1}$)

end

context *normal-subgroup* **begin**

p 56, ll 28–29

theorem *Class-unit-normal-subgroup*: *Class* $\mathbf{1} = K$
  ⟨*proof*⟩

p 56, ll 1–2; p 56, l 29

theorem *Class-is-Left-Coset*:
  $g \in G \Longrightarrow$ *Class* $g = g \cdot| K$
  ⟨*proof*⟩

p 56, l 29

lemma *Left-CosetE*: $[\![A \in G$ // K; $\bigwedge a.\ a \in G \Longrightarrow P\ (a \cdot| K)]\!] \Longrightarrow P\ A$
  ⟨*proof*⟩

Equation 26

p 56, ll 32–34

theorem *factor-composition* [*simp*]:
  $[\![g \in G; h \in G]\!] \Longrightarrow (g \cdot| K)\ [\cdot]\ (h \cdot| K) = g \cdot h \cdot| K$
  ⟨*proof*⟩

p 56, l 35

theorem *factor-unit*:
  $K = \mathbf{1} \cdot| K$
  ⟨*proof*⟩

p 56, l 35

theorem *factor-inverse* [*simp*]:
  $g \in G \Longrightarrow$ *quotient.inverse* $(g \cdot| K) =$ (*inverse* $g \cdot| K$)
  ⟨*proof*⟩

end

p 57, ll 4–5

locale *subgroup-of-abelian-group* = *subgroup-of-group* H G (·) **1** + *abelian-group* G (·) **1**
for H **and** G **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›)

p 57, ll 4–5

sublocale *subgroup-of-abelian-group* ⊆ *normal-subgroup* H G (·) **1**
⟨*proof*⟩

2.7 Homomorphims

Def 1.6

p 58, l 33; p 59, ll 1–2

locale *monoid-homomorphism* =
map η M M'+ *source*: *monoid* M (·) **1** + *target*: *monoid* M' (·′) **1**′
for η **and** M **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›)
and M' **and** *composition′* (**infixl** ‹·′′› *70*) **and** *unit′* (‹**1**′′›) +
assumes *commutes-with-composition*: ⟦ $x \in M$; $y \in M$ ⟧ $\Longrightarrow$ η $(x \cdot y) = \eta\ x \cdot' \eta\ y$
and *commutes-with-unit*: η **1** = **1**′
begin

Jacobson notes that *commutes-with-unit* is not necessary for groups, but doesn't make use of that later.

p 58, l 33; p 59, ll 1–2

notation *source.invertible* (‹*invertible* -› [*100*] *100*)
notation *source.inverse* (‹*inverse* -› [*100*] *100*)
notation *target.invertible* (‹*invertible′′* -› [*100*] *100*)
notation *target.inverse* (‹*inverse′′* -› [*100*] *100*)

end

p 59, ll 29–30

locale *monoid-epimorphism* = *monoid-homomorphism* + *surjective-map* η M M'

p 59, l 30

locale *monoid-monomorphism* = *monoid-homomorphism* + *injective-map* η M M'

p 59, ll 30–31

sublocale *monoid-isomorphism* ⊆ *monoid-epimorphism*
⟨*proof*⟩

p 59, ll 30–31

sublocale *monoid-isomorphism* ⊆ *monoid-monomorphism*
⟨*proof*⟩

context *monoid-homomorphism* **begin**

p 59, ll 33–34

theorem *invertible-image-lemma*:
  assumes *invertible a a* $\in$ *M*
  shows η *a* $\cdot'$ η (*inverse a*) = $\mathbf{1}'$ **and** η (*inverse a*) $\cdot'$ η *a* = $\mathbf{1}'$
  ⟨*proof*⟩

p 59, l 34; p 60, l 1

theorem *invertible-target-invertible* [*intro*, *simp*]:
  ⟦ *invertible a*; *a* $\in$ *M* ⟧ $\Longrightarrow$ *invertible'* (η *a*)
  ⟨*proof*⟩

p 60, l 1

theorem *invertible-commutes-with-inverse*:
  ⟦ *invertible a*; *a* $\in$ *M* ⟧ $\Longrightarrow$ η (*inverse a*) = *inverse'* (η *a*)
  ⟨*proof*⟩

end

p 60, ll 32–34; p 61, l 1

sublocale *monoid-congruence* $\subseteq$ *natural*: *monoid-homomorphism Class M* ($\cdot$) **1** *M* / *E* ([$\cdot$]) *Class* **1**
  ⟨*proof*⟩

Fundamental Theorem of Homomorphisms of Monoids

p 61, ll 5, 14–16

sublocale *monoid-homomorphism* $\subseteq$ *image*: *submonoid* η ‘ *M* *M'* ($\cdot'$) $\mathbf{1}'$
  ⟨*proof*⟩

p 61, l 4

locale *monoid-homomorphism-fundamental* = *monoid-homomorphism* **begin**

p 61, ll 17–18

sublocale *fiber-relation* η *M M'* ⟨*proof*⟩
notation *Fiber-Relation* (‹*E'*(-')›)

p 61, ll 6–7, 18–20

sublocale *monoid-congruence* **where** *E* = *E*(η)
  ⟨*proof*⟩

p 61, ll 7–9

induced denotes Jacobson's $\bar{\eta}$. We have the commutativity of the diagram, where *induced* is unique:

compose M induced Class = η

$[\![?\beta \in Partition \rightarrow_E M'; compose\ M\ ?\beta\ Class = \eta]\!] \Longrightarrow ?\beta = induced$

.

p 61, l 20

notation *quotient-composition* (**infixl** ‹[·]› *70*)

p 61, ll 7–8, 22–25

sublocale *induced*: *monoid-homomorphism induced M / E*(η) ([·]) *Class* **1** *M′* (·′) **1′**
⟨*proof*⟩

p 61, ll 9, 26

sublocale *natural*: *monoid-epimorphism Class M* (·) **1** *M / E*(η) ([·]) *Class* **1** ⟨*proof*⟩

p 61, ll 9, 26–27

sublocale *induced*: *monoid-monomorphism induced M / E*(η) ([·]) *Class* **1** *M′* (·′) **1′**
⟨*proof*⟩

end

p 62, ll 12–13

locale *group-homomorphism* =
monoid-homomorphism η *G* (·) **1** *G′* (·′) **1′** +
source: *group G* (·) **1** + *target*: *group G′* (·′) **1′**
for η **and** *G* **and** *composition* (**infixl** ‹·› *70*) **and** *unit* (‹**1**›)
and *G′* **and** *composition′* (**infixl** ‹·′′› *70*) **and** *unit′* (‹**1′′**›)
begin

p 62, l 13

sublocale *image*: *subgroup* η ‘ *G G′* (·′) **1′**
⟨*proof*⟩

p 62, ll 13–14

definition *Ker* = η −‘ {**1′**} ∩ *G*

p 62, ll 13–14

lemma *Ker-equality*:
Ker = {*a* | *a*. *a* ∈ *G* ∧ η *a* = **1′**}
⟨*proof*⟩

p 62, ll 13–14

lemma *Ker-closed* [*intro*, *simp*]:
a ∈ *Ker* ⟹ *a* ∈ *G*
⟨*proof*⟩

p 62, ll 13–14

lemma *Ker-image* [*intro*]:

$a \in Ker \Longrightarrow \eta\ a = \mathbf{1}'$
⟨*proof*⟩

p 62, ll 13–14

lemma *Ker-memI* [*intro*]:
$[\![\ \eta\ a = \mathbf{1}';\ a \in G\ ]\!] \Longrightarrow a \in Ker$
⟨*proof*⟩

p 62, ll 15–16

sublocale *kernel*: *normal-subgroup Ker G*
⟨*proof*⟩

p 62, ll 17–20

theorem *injective-iff-kernel-unit*:
inj-on $\eta\ G \longleftrightarrow Ker = \{\mathbf{1}\}$
⟨*proof*⟩

end

p 62, l 24

locale *group-epimorphism* = *group-homomorphism* + *monoid-epimorphism* η G $(\cdot)$ **1** G' $(\cdot')$ $\mathbf{1}'$

p 62, l 21

locale *normal-subgroup-in-kernel* =
group-homomorphism + *contained*: *normal-subgroup* L G $(\cdot)$ **1** **for** L +
assumes *subset*: $L \subseteq Ker$
begin

p 62, l 21

notation *contained.quotient-composition* (**infixl** ‹[·]› *70*)

"homomorphism onto *contained.Partition*"

p 62, ll 23–24

sublocale *natural*: *group-epimorphism contained.Class* G $(\cdot)$ **1** G // L $([\cdot])$ *contained.Class* **1** ⟨*proof*⟩

p 62, ll 25–26

theorem *left-coset-equality*:
assumes *eq*: $a \cdot| L = b \cdot| L$ **and** [*simp*]: $a \in G$ **and** *b*: $b \in G$
shows $\eta\ a = \eta\ b$
⟨*proof*⟩

$\bar{\eta}$

p 62, ll 26–27

definition *induced* = ($\lambda A \in G$ // *L*. *THE b*. $\exists\, a \in G$. $a \cdot| L = A \wedge b = \eta\ a$)

p 62, ll 26–27

lemma *induced-closed* [*intro*, *simp*]:
assumes [*simp*]: $A \in G$ // *L* **shows** *induced* $A \in G'$
⟨*proof*⟩

p 62, ll 26–27

lemma *induced-undefined* [*intro*, *simp*]:
$A \notin G$ // $L \Longrightarrow$ *induced* A = *undefined*
⟨*proof*⟩

p 62, ll 26–27

theorem *induced-left-coset-closed* [*intro*, *simp*]:
$a \in G \Longrightarrow$ *induced* $(a \cdot| L) \in G'$
⟨*proof*⟩

p 62, ll 26–27

theorem *induced-left-coset-equality* [*simp*]:
assumes [*simp*]: $a \in G$ **shows** *induced* $(a \cdot| L) = \eta\ a$
⟨*proof*⟩

p 62, l 27

theorem *induced-Left-Coset-commutes-with-composition* [*simp*]:
⟦ $a \in G$; $b \in G$ ⟧ $\Longrightarrow$ *induced* $((a \cdot| L)\ [\cdot]\ (b \cdot| L))$ = *induced* $(a \cdot| L) \cdot'$ *induced* $(b$
$\cdot| L)$
⟨*proof*⟩

p 62, ll 27–28

theorem *induced-group-homomorphism*:
group-homomorphism induced (*G* // *L*) ([·]) (*contained.Class* **1**) G' $(\cdot')$ $\mathbf{1}'$
⟨*proof*⟩

p 62, l 28

sublocale *induced*: *group-homomorphism induced G* // *L* ([·]) *contained.Class* **1** G'
$(\cdot')$ $\mathbf{1}'$
⟨*proof*⟩

p 62, ll 28–29

theorem *factorization-lemma*: $a \in G \Longrightarrow$ *compose G induced contained.Class* $a = \eta$
a
⟨*proof*⟩

p 62, ll 29–30

theorem *factorization* [*simp*]: *compose G induced contained.Class* $= \eta$
⟨*proof*⟩

Jacobson does not state the uniqueness of *induced* explicitly but he uses it later, for rings, on p 107.

p 62, l 30

theorem *uniqueness*:
  assumes *map*: $\beta \in G \mathbin{//} L \to_E G'$
    and *factorization*: $compose\ G\ \beta\ contained.Class = \eta$
  shows $\beta = induced$
⟨*proof*⟩

p 62, l 31

theorem *induced-image*:
  $induced\ `\ (G \mathbin{//} L) = \eta\ `\ G$
  ⟨*proof*⟩

p 62, l 33

interpretation *L*: *normal-subgroup L Ker*
  ⟨*proof*⟩

p 62, ll 31–33

theorem *induced-kernel*:
  $induced.Ker = Ker\ /\ L.Congruence$
⟨*proof*⟩

p 62, ll 34–35

theorem *induced-inj-on*:
  $inj\text{-}on\ induced\ (G \mathbin{//} L) \longleftrightarrow L = Ker$
  ⟨*proof*⟩

end

Fundamental Theorem of Homomorphisms of Groups

p 63, l 1

locale *group-homomorphism-fundamental* = *group-homomorphism* **begin**

p 63, l 1

notation *kernel.quotient-composition* (**infixl** ‹[·]› *70*)

p 63, l 1

sublocale *normal-subgroup-in-kernel* **where** $L = Ker$ ⟨*proof*⟩

p 62, ll 36–37; p 63, l 1

induced denotes Jacobson's $\bar{\eta}$. We have the commutativity of the diagram, where *induced* is unique:

$compose\ G\ induced\ kernel.Class = \eta$

$[\![?\beta \in kernel.Partition \rightarrow_E G'; compose\ G\ ?\beta\ kernel.Class = \eta]\!] \Longrightarrow ?\beta = induced$

end

p 63, l 5

locale *group-isomorphism* = *group-homomorphism* + *bijective-map* η G G' **begin**

p 63, l 5

sublocale *monoid-isomorphism* η G $(\cdot)$ $\mathbf{1}$ G' $(\cdot')$ $\mathbf{1}'$
⟨*proof*⟩

p 63, l 6

lemma *inverse-group-isomorphism*:
group-isomorphism (*restrict* (*inv-into* G η) G') G' $(\cdot')$ $\mathbf{1}'$ G $(\cdot)$ $\mathbf{1}$
⟨*proof*⟩

end

p 63, l 6

definition *isomorphic-as-groups* (**infixl** ‹$\cong_G$› *50*)
where $\mathcal{G} \cong_G \mathcal{G}' \longleftrightarrow$ (*let* (G, *composition*, *unit*) = $\mathcal{G}$; (G', *composition'*, *unit'*) = $\mathcal{G}'$ *in*
($\exists\,\eta$. *group-isomorphism* η G *composition unit* G' *composition' unit'*))

p 63, l 6

lemma *isomorphic-as-groups-symmetric*:
(G, *composition*, *unit*) $\cong_G$ (G', *composition'*, *unit'*) $\Longrightarrow$ (G', *composition'*, *unit'*) $\cong_G$ (G, *composition*, *unit*)
⟨*proof*⟩

p 63, l 1

sublocale *group-isomorphism* $\subseteq$ *group-epimorphism* ⟨*proof*⟩

p 63, l 1

locale *group-epimorphism-fundamental* = *group-homomorphism-fundamental* + *group-epimorphism*
begin

p 63, ll 1–2

interpretation *image*: *group-homomorphism induced* G // *Ker* ([·]) *kernel.Class* $\mathbf{1}$ (η ‘ G) $(\cdot')$ $\mathbf{1}'$
⟨*proof*⟩

p 63, ll 1–2

sublocale *image*: *group-isomorphism induced* G // *Ker* ([·]) *kernel.Class* $\mathbf{1}$ (η ‘ G) $(\cdot')$ $\mathbf{1}'$
⟨*proof*⟩

end

context *group-homomorphism* **begin**

p 63, ll 5–7

theorem *image-isomorphic-to-factor-group*:
$\exists K$ *composition unit*. *normal-subgroup* K G $(\cdot)$ $\mathbf{1}$ $\wedge$ $(\eta$ ' G, $(\cdot')$, $\mathbf{1}')$ $\cong_G$ $(G$ // K, *composition*, *unit*)
⟨*proof*⟩

end

no-notation *plus* (**infixl** ‹+› *65*)
no-notation *minus* (**infixl** ‹−› *65*)
unbundle *no uminus-syntax*
no-notation *quotient* (**infixl** ‹′/′/› *90*)

3 Rings

3.1 Definition and Elementary Properties

Def 2.1

p 86, ll 20–28

locale *ring* = *additive*: *abelian-group* R (+) $\mathbf{0}$ + *multiplicative*: *monoid* R $(\cdot)$ $\mathbf{1}$
for R **and** *addition* (**infixl** ‹+› *65*) **and** *multiplication* (**infixl** ‹·› *70*) **and** *zero* (‹**0**›) **and** *unit* (‹**1**›) +
assumes *distributive*: $[\![a \in R; b \in R; c \in R]\!] \Longrightarrow a \cdot (b + c) = a \cdot b + a \cdot c$
$[\![a \in R; b \in R; c \in R]\!] \Longrightarrow (b + c) \cdot a = b \cdot a + c \cdot a$
begin

p 86, ll 20–28

notation *additive.inverse* (‹− -› [*66*] *65*)
abbreviation *subtraction* (**infixl** ‹−› *65*) **where** $a - b \equiv a + (- b)$

end

p 87, ll 10–12

locale *subring* =
additive: *subgroup* S R (+) $\mathbf{0}$ + *multiplicative*: *submonoid* S R $(\cdot)$ $\mathbf{1}$
for S **and** R **and** *addition* (**infixl** ‹+› *65*) **and** *multiplication* (**infixl** ‹·› *70*) **and** *zero* (‹**0**›) **and** *unit* (‹**1**›)

context *ring* **begin**

p 88, ll 26–28

lemma *right-zero* [*simp*]:
  assumes [*simp*]: $a \in R$ **shows** $a \cdot \mathbf{0} = \mathbf{0}$
⟨*proof*⟩

p 88, l 29

lemma *left-zero* [*simp*]:
  assumes [*simp*]: $a \in R$ **shows** $\mathbf{0} \cdot a = \mathbf{0}$
⟨*proof*⟩

p 88, ll 29–30; p 89, ll 1–2

lemma *left-minus*:
  assumes [*simp*]: $a \in R$ $b \in R$ **shows** $(- a) \cdot b = - a \cdot b$
⟨*proof*⟩

p 89, l 3

lemma *right-minus*:
  assumes [*simp*]: $a \in R$ $b \in R$ **shows** $a \cdot (- b) = - a \cdot b$
⟨*proof*⟩

end

3.2 Ideals, Quotient Rings

p 101, ll 2–5

locale *ring-congruence* = *ring* +
  additive: *group-congruence* R (+) **0** E +
  multiplicative: *monoid-congruence* R (·) **1** E
  for E
begin

p 101, ll 2–5

notation *additive.quotient-composition* (**infixl** ‹[+]› *65*)
notation *additive.quotient.inverse* (‹[−] -› [*66*] *65*)
notation *multiplicative.quotient-composition* (**infixl** ‹[·]› *70*)

p 101, ll 5–11

sublocale *quotient*: *ring* R / E ([+]) ([·]) *additive.Class* **0** *additive.Class* **1**
  ⟨*proof*⟩

end

p 101, ll 12–13

locale *subgroup-of-additive-group-of-ring* =
  additive: *subgroup* I R (+) **0** + *ring* R (+) (·) **0** **1**
  for I **and** R **and** *addition* (**infixl** ‹+› *65*) **and** *multiplication* (**infixl** ‹·› *70*) **and** *zero* (‹**0**›) **and** *unit* (‹**1**›)
begin

p 101, ll 13–14

definition *Ring-Congruence* = {(*a*, *b*). *a* ∈ *R* ∧ *b* ∈ *R* ∧ *a* − *b* ∈ *I*}

p 101, ll 13–14

lemma *Ring-CongruenceI*: ⟦ *a* − *b* ∈ *I*; *a* ∈ *R*; *b* ∈ *R* ⟧ ⟹ (*a*, *b*) ∈ *Ring-Congruence*
⟨*proof*⟩

p 101, ll 13–14

lemma *Ring-CongruenceD*: (*a*, *b*) ∈ *Ring-Congruence* ⟹ *a* − *b* ∈ *I*
⟨*proof*⟩

Jacobson's definition of ring congruence deviates from that of group congruence; this complicates the proof.

p 101, ll 12–14

sublocale *additive*: *subgroup-of-abelian-group I R* (+) **0**
rewrites *additive-congruence*: *additive.Congruence* = *Ring-Congruence*
⟨*proof*⟩

p 101, l 14

notation *additive.Left-Coset* (**infixl** ‹+|› *65*)

end

Def 2.2

p 101, ll 21–22

locale *ideal* = *subgroup-of-additive-group-of-ring* +
assumes *ideal*: ⟦ *a* ∈ *R*; *b* ∈ *I* ⟧ ⟹ *a* · *b* ∈ *I* ⟦ *a* ∈ *R*; *b* ∈ *I* ⟧ ⟹ *b* · *a* ∈ *I*

context *subgroup-of-additive-group-of-ring* **begin**

p 101, ll 14–17

theorem *multiplicative-congruence-implies-ideal*:
assumes *monoid-congruence R* (·) **1** *Ring-Congruence*
shows *ideal I R* (+) (·) **0** **1**
⟨*proof*⟩

end

context *ideal* **begin**

p 101, ll 17–20

theorem *multiplicative-congruence* [*intro*]:
assumes *a*: (*a*, *a*′) ∈ *Ring-Congruence* **and** *b*: (*b*, *b*′) ∈ *Ring-Congruence*
shows (*a* · *b*, *a*′ · *b*′) ∈ *Ring-Congruence*
⟨*proof*⟩

p 101, ll 23–24

sublocale *ring-congruence* **where** $E = \mathit{Ring\text{-}Congruence}$ ⟨*proof*⟩

end

context *ring* **begin**

Pulled out of *ideal* to achieve standard notation.

p 101, ll 24–26

abbreviation *Quotient-Ring* (**infixl** ‹'/'/› *75*)
where $S \mathbin{//} I \equiv S \mathbin{/} (\mathit{subgroup\text{-}of\text{-}additive\text{-}group\text{-}of\text{-}ring.Ring\text{-}Congruence}\ I\ R\ (+)$
$\mathbf{0})$

end

p 101, ll 24–26

locale *quotient-ring* = *ideal* **begin**

p 101, ll 24–26

sublocale *quotient*: *ring* $R \mathbin{//} I$ ([+]) ([·]) *additive.Class* **0** *additive.Class* **1** ⟨*proof*⟩

p 101, l 26

lemmas *Left-Coset* = *additive.Left-CosetE*

Equation 17 (1)

p 101, l 28

lemmas *quotient-addition* = *additive.factor-composition*

Equation 17 (2)

p 101, l 29

theorem *quotient-multiplication* [*simp*]:
$[\![\ a \in R;\ b \in R\ ]\!] \Longrightarrow (a \mathbin{+|} I)\ [\cdot]\ (b \mathbin{+|} I) = a \cdot b \mathbin{+|} I$
⟨*proof*⟩

p 101, l 30

lemmas *quotient-zero* = *additive.factor-unit*
lemmas *quotient-negative* = *additive.factor-inverse*

end

3.3 Homomorphisms of Rings. Basic Theorems

Def 2.3

p 106, ll 7–9

locale *ring-homomorphism* =
map η *R R'* + *source*: *ring R* (+) (·) **0** **1** + *target*: *ring R'* (+') (·') **0'** **1'** +
additive: *group-homomorphism* η *R* (+) **0** *R'* (+') **0'** +
multiplicative: *monoid-homomorphism* η *R* (·) **1** *R'* (·') **1'**
for η
and *R* **and** *addition* (**infixl** ‹+› *65*) **and** *multiplication* (**infixl** ‹·› *70*) **and** *zero* (‹**0**›) **and** *unit* (‹**1**›)
and *R'* **and** *addition'* (**infixl** ‹+''› *65*) **and** *multiplication'* (**infixl** ‹·''› *70*) **and** *zero'* (‹**0''**›) **and** *unit'* (‹**1''**›)

p 106, l 17

locale *ring-epimorphism* = *ring-homomorphism* + *surjective-map* η *R R'*

p 106, ll 14–18

sublocale *quotient-ring* ⊆ *natural*: *ring-epimorphism*
where η = *additive.Class* **and** *R'* = *R* // *I* **and** *addition'* = ([+]) **and** *multiplication'* = ([·])
and *zero'* = *additive.Class* **0** **and** *unit'* = *additive.Class* **1**
⟨*proof*⟩

context *ring-homomorphism* **begin**

Jacobson reasons via $a - b \in$ *additive.Ker* being a congruence; we prefer the direct proof, since it is very simple.

p 106, ll 19–21

sublocale *kernel*: *ideal* **where** *I* = *additive.Ker*
⟨*proof*⟩

end

p 106, l 22

locale *ring-monomorphism* = *ring-homomorphism* + *injective-map* η *R R'*

context *ring-homomorphism* **begin**

p 106, ll 21–23

theorem *ring-monomorphism-iff-kernel-unit*:
ring-monomorphism η *R* (+) (·) **0** **1** *R'* (+') (·') **0'** **1'** ⟷ *additive.Ker* = {**0**} (**is** *?monom* ⟷ *?ker*)
⟨*proof*⟩

end

p 106, ll 23–25

sublocale *ring-homomorphism* ⊆ *image*: *subring* η ‘ *R R'* (+') (·') **0'** **1'** ⟨*proof*⟩

p 106, ll 26–27

locale *ideal-in-kernel* =
ring-homomorphism + *contained*: *ideal I R* (+) (·) **0** **1** **for** *I* +
assumes *subset*: *I* ⊆ *additive.Ker*
begin

p 106, ll 26–27

notation *contained.additive.quotient-composition* (**infixl** ‹[+]› *65*)
notation *contained.multiplicative.quotient-composition* (**infixl** ‹[·]› *70*)

Provides *additive.induced*, which Jacobson calls $\bar{\eta}$.

p 106, l 30

sublocale *additive*: *normal-subgroup-in-kernel* η *R* (+) **0** *R′* (+′) **0′** *I*
rewrites *normal-subgroup.Congruence I R addition zero* = *contained.Ring-Congruence*
⟨*proof*⟩

Only the multiplicative part needs some work.

p 106, ll 27–30

sublocale *induced*: *ring-homomorphism additive.induced R* // *I* ([+]) ([·]) *contained.additive.Class* **0** *contained.additive.Class* **1**
⟨*proof*⟩

p 106, l 30; p 107, ll 1–3

additive.induced denotes Jacobson's $\bar{\eta}$. We have the commutativity of the diagram, where *additive.induced* is unique:

compose R additive.induced contained.additive.Class = η

⟦*?β* ∈ *contained.additive.Partition* $\rightarrow_E$ *R′*;
compose R ?β contained.additive.Class = η⟧
⟹ *?β* = *additive.induced*

end

Fundamental Theorem of Homomorphisms of Rings

p 107, l 6

locale *ring-homomorphism-fundamental* = *ring-homomorphism* **begin**

p 107, l 6

notation *kernel.additive.quotient-composition* (**infixl** ‹[+]› *65*)
notation *kernel.multiplicative.quotient-composition* (**infixl** ‹[·]› *70*)

p 107, l 6

sublocale *ideal-in-kernel* **where** *I* = *additive.Ker* ⟨*proof*⟩

p 107, ll 8–9

sublocale *natural*: *ring-epimorphism*
where η = *kernel.additive.Class* **and** R' = R // *additive.Ker*
and *addition'* = *kernel.additive.quotient-composition*
and *multiplication'* = *kernel.multiplicative.quotient-composition*
and *zero'* = *kernel.additive.Class* **0** **and** *unit'* = *kernel.additive.Class* **1**
⟨*proof*⟩

p 107, l 9

sublocale *induced*: *ring-monomorphism*
where η = *additive.induced* **and** R = R // *additive.Ker*
and *addition* = *kernel.additive.quotient-composition*
and *multiplication* = *kernel.multiplicative.quotient-composition*
and *zero* = *kernel.additive.Class* **0** **and** *unit* = *kernel.additive.Class* **1**
⟨*proof*⟩

end

p 107, l 11

locale *ring-isomorphism* = *ring-homomorphism* + *bijective-map* η R R' **begin**

p 107, l 11

sublocale *ring-monomorphism* ⟨*proof*⟩
sublocale *ring-epimorphism* ⟨*proof*⟩

p 107, l 11

lemma *inverse-ring-isomorphism*:
ring-isomorphism (*restrict* (*inv-into* R η) R') R' (+') (·') **0'** **1'** R (+) (·) **0** **1**
⟨*proof*⟩

end

p 107, l 11

definition *isomorphic-as-rings* (**infixl** ‹$\cong_R$› *50*)
where $\mathcal{R} \cong_R \mathcal{R}' \longleftrightarrow$ (*let* (R, *addition*, *multiplication*, *zero*, *unit*) = $\mathcal{R}$; (R', *addition'*, *multiplication'*, *zero'*, *unit'*) = $\mathcal{R}'$ *in*
($\exists\,\eta$. *ring-isomorphism* η R *addition multiplication zero unit* R' *addition' multiplication' zero' unit'*))

p 107, l 11

lemma *isomorphic-as-rings-symmetric*:
(R, *addition*, *multiplication*, *zero*, *unit*) $\cong_R$ (R', *addition'*, *multiplication'*, *zero'*, *unit'*) $\Longrightarrow$
(R', *addition'*, *multiplication'*, *zero'*, *unit'*) $\cong_R$ (R, *addition*, *multiplication*, *zero*, *unit*)
⟨*proof*⟩

context *ring-homomorphism* **begin**

Corollary

p 107, ll 11–12

theorem *image-is-isomorphic-to-quotient-ring*:
$\exists K$ *add mult zero one*. *ideal K R* $(+)$ $(\cdot)$ $\mathbf{0}$ $\mathbf{1}$ $\wedge$ $(\eta$ ' R, $(+')$, $(\cdot')$, $\mathbf{0}'$, $\mathbf{1}') \cong_R (R$ // K, *add*, *mult*, *zero*, *one*)
⟨*proof*⟩

end

References

[1] N. Jacobson. *Basic Algebra*, volume I. Freeman, 2nd edition, 1985.