```
THEORY Ring
  IMPORT THEORY PROJECTS
    /SimpleDEq THEORIES /SimpleDEq/Group.dtf|org.eventb.theory.core.deployedTheoryRoot#Group
  TYPE PARAMETERS A
  OPERATORS
    distributive <predicate> (oplus: (A × A) → A, otimes: (A × A) → A)
      well−definedness A ≠ ∅
      direct definition
        ∀ x, y, z · x ∈ A ∧ y ∈ A ∧ z ∈ A ⇒ (
          otimes(x ↦ oplus(y ↦ z)) = oplus(otimes(x ↦ y) ↦ otimes(x ↦ z)) ∧
          otimes(oplus(y ↦ z) ↦ x) = oplus(otimes(y ↦ x) ↦ otimes(z ↦ x))
        )
    integral <predicate> (otimes: (A × A) → A, azero: A)
      well−definedness A ≠ ∅ ∧ A ≠ {azero}
      direct definition
        ∀ x, y · x ∈ A ∧ y ∈ A ⇒ (otimes(x ↦ y) = azero ⇒ (x = azero ∨ y = azero))
    Ring <predicate> (oplus: (A × A) → A, otimes: (A × A) → A, azero: A, aunit: A)
      well−definedness A ≠ ∅
      direct definition
        AbelianGroup(oplus, azero) ∧ Monoid(otimes, aunit) ∧ distributive(oplus, otimes)
    CommutativeRing <predicate> (oplus: (A × A) → A, otimes: (A × A) → A, azero: A, aunit: A)
      well−definedness A ≠ ∅
      direct definition
        Ring(oplus, otimes, azero, aunit) ∧ commutative(otimes)
    nonZeroInvertible <predicate> (op: (A × A) → A, azero: A, aunit: A)
      well−definedness A ≠ ∅ ∧ (azero ≠ aunit)
      direct definition
        ∀ x · x ∈ A ∧ x ≠ azero ⇒ (∃ y · y ∈ A ∧ y ≠ azero ⇒ (op(x ↦ y) = aunit ∧ op(y ↦ x) = aunit)
          )
    nonZeroInverses <predicate> (op: (A × A) → A, azero: A, aunit: A, x: A, y: A)
      well−definedness A ≠ ∅ ∧ (azero ≠ aunit)
      direct definition
        nonZeroInvertible(op, azero, aunit) ∧ x ≠ azero ⇒ ((op(x ↦ y) = aunit) ∧ (op(y ↦ x) = aunit))
    DivisionRing <predicate> (oplus: (A × A) → A, otimes: (A × A) → A, azero: A, aunit: A)
      well−definedness A ≠ ∅
      direct definition
        (azero ≠ aunit) ∧ Ring(oplus, otimes, azero, aunit) ∧ nonZeroInvertible(otimes, azero, aunit)
    Field <predicate> (oplus: (A × A) → A, otimes: (A × A) → A, azero: A, aunit: A)
      well−definedness A ≠ ∅
      direct definition
        (azero ≠ aunit) ∧
        Ring(oplus, otimes, azero, aunit) ∧
        nonZeroInvertible(otimes, azero, aunit) ∧
        integral(otimes, azero) ∧
        commutative(otimes)
    absorbing <predicate> (op: (A × A) → A, azero: A)
      direct definition
        ∀ x · x ∈ A ⇒ (op(x ↦ azero) = azero ∧ op(azero ↦ x) = azero)
  THEOREMS
    zeroAbsorbing:
      ∀ oplus, otimes, azero, aunit · oplus ∈ ((A × A) → A) ∧ otimes ∈ ((A × A) → A) ∧ azero ∈ A ∧ aunit
          ∈ A ∧
        Ring(oplus, otimes, azero, aunit) ⇒ absorbing(otimes, azero)
    plusInverseLeftDistribution:
      ∀ oplus, otimes, azero, aunit · oplus ∈ ((A × A) → A) ∧ otimes ∈ ((A × A) → A) ∧ azero ∈ A ∧ aunit
          ∈ A ∧ Ring(oplus, otimes, azero, aunit) ⇒ (
        ∀ a, b, a1 · a ∈ A ∧ b ∈ A ∧ a1 ∈ A ∧ inverses(oplus, azero, a, a1) ⇒ (
          inverses(oplus, azero, otimes(a ↦ b), otimes(a1 ↦ b))
        )
      )
    plusInverseRightDistribution:
      ∀ oplus, otimes, azero, aunit · oplus ∈ ((A × A) → A) ∧ otimes ∈ ((A × A) → A) ∧ azero ∈ A ∧ aunit
          ∈ A ∧ Ring(oplus, otimes, azero, aunit) ⇒ (
        ∀ a, b, b1 · a ∈ A ∧ b ∈ A ∧ b1 ∈ A ∧ inverses(oplus, azero, b, b1) ⇒ (
          inverses(oplus, azero, otimes(a ↦ b), otimes(a ↦ b1))
        )
      )
    ringLeftCancellation:
      ∀ oplus, otimes, azero, aunit · oplus ∈ ((A × A) → A) ∧ otimes ∈ ((A × A) → A) ∧ azero ∈ A ∧ aunit
          ∈ A ∧ azero ≠ aunit ∧ Ring(oplus, otimes, azero, aunit) ∧ integral(otimes, azero) ⇒ (
        ∀ a, b, c · a ∈ A ∧ b ∈ A ∧ c ∈ A ∧ a ≠ azero ⇒
          ((otimes(a ↦ b) = otimes(a ↦ c)) ⇔ (b = c))
      )
```

```
    ringRightCancellation:
      ∀ oplus, otimes, azero, aunit · oplus ∈ ((A × A) → A) ∧ otimes ∈ ((A × A) → A) ∧ azero ∈ A ∧ aunit
          ∈ A ∧ azero ≠ aunit ∧ Ring(oplus, otimes, azero, aunit) ∧ integral(otimes, azero) ⇒ (
        ∀ a, b, c · a ∈ A ∧ b ∈ A ∧ c ∈ A ∧ a ≠ azero ⇒
          ((otimes(b ↦ a) = otimes(c ↦ a)) ⇔ (b = c))
      )
    nonZeroInverseNotZero:
      ∀ oplus, otimes, azero, aunit, x, y ·
        oplus ∈ ((A × A) → A) ∧ otimes ∈ ((A × A) → A) ∧ azero ∈ A ∧ aunit ∈ A ∧
        Ring(oplus, otimes, azero, aunit) ∧ nonZeroInvertible(otimes, azero, aunit) ∧
        x ∈ A ∧ y ∈ A ∧ x ≠ azero ∧ nonZeroInverses(otimes, azero, aunit, x, y) ⇒ (
          y ≠ azero
      )
END
```