Library reals_axioms

reals_axioms

```
Require Export mk_theorems.

Class Real_Struct := {
    R : Class;
    fp : Class;
    zeroR : Class;
    fm : Class;
    oneR : Class;
    Leq : Class; }.

Section real_numbers.


Class Real_Axioms (R_str : Real_Struct) := {
    Ensemble_R : Ensemble R;
    PlusR : (Function fp) /\ (dom(fp) = R × R) /\ (ran(fp) ⊂ R);
    zero_in_R : 0 ∈ R;
    Plus_P1 : ∀ x, x ∈ R -> x + 0 = x;
    Plus_P2 : ∀ x, x ∈ R -> ∃ y, y ∈ R /\ x + y = 0;
    Plus_P3 : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R -> x + (y + z) = (x + y) + z;
    Plus_P4 : ∀ x y, x ∈ R -> y ∈ R -> x + y = y + x;
    MultR : (Function fm) /\ (dom(fm) = R × R) /\ (ran(fm) ⊂ R);
    one_in_R : 1 ∈ (R ~ [0]);
    Mult_P1 : ∀ x, x ∈ R -> x · 1 = x;
    Mult_P2 : ∀ x, x ∈ (R ~ [0]) -> ∃ y, y ∈ (R ~ [0]) /\ x · y = 1;
    Mult_P3 : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R -> x · (y · z) = (x · y) · z;
    Mult_P4 : ∀ x y, x ∈ R -> y ∈ R -> x · y = y · x;
    Mult_P5 : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R
        -> (x + y) · z = (x · z) + (y · z);
    LeqR : Leq ⊂ R × R;
    Leq_P1 : ∀ x, x ∈ R -> x ≤ x;
    Leq_P2 : ∀ x y, x ∈ R -> y ∈ R -> x ≤ y -> y ≤ x -> x = y;
    Leq_P3 : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R -> x ≤ y -> y ≤ z -> x ≤ z;
    Leq_P4 : ∀ x y, x ∈ R -> y ∈ R -> x ≤ y \/ y ≤ x;
    Plus_Leq : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R -> x ≤ y -> x + z ≤ y + z;
    Mult_Leq : ∀ x y, x ∈ R -> y ∈ R -> 0 ≤ x -> 0 ≤ y -> 0 ≤ x · y;
    Completeness : ∀ X Y, X ⊂ R -> Y ⊂ R -> X <> Φ -> Y <> Φ
        -> (∀ x y, x ∈ X -> y ∈ Y -> x ≤ y)
        -> ∃ c, c ∈ R /\ (∀ x y, x ∈ X -> y ∈ Y -> (x ≤ c /\ c ≤ y)); }.

Variable R_str : Real_Struct.
Variable RA : Real_Axioms R_str.

Corollary one_in_R_Co : 1 ∈ R.

Local Hint Resolve zero_in_R one_in_R one_in_R_Co : real.


Corollary Plus_close : ∀ x y, x ∈ R -> y ∈ R -> (x + y) ∈ R.

Local Hint Resolve Plus_close : real.

Corollary Plus_Co1 : ∀ x, x ∈ R -> (∀ y, y ∈ R -> y + x = y) -> x = 0.

Corollary Plus_Co2 : ∀ x, x ∈ R -> (∃! x0, x0 ∈ R /\ x + x0 = 0).
```

```
Corollary Plus_neg1a : ∀ a, a ∈ R -> (-a) ∈ R.

Corollary Plus_neg1b : ∀ a, (-a) ∈ R -> a ∈ R.

Corollary Plus_neg2 : ∀ a, a ∈ R -> a + (-a) = 0.

Corollary Minus_P1 : ∀ a, a ∈ R -> a - a = 0.

Corollary Minus_P2 : ∀ a, a ∈ R -> a - 0 = a.

Local Hint Resolve Plus_neg1a Plus_neg1b Plus_neg2 Minus_P1 Minus_P2 : real.

Corollary Plus_Co3 : ∀ a x b, a ∈ R -> x ∈ R -> b ∈ R -> a + x = b
    -> x = b + (-a).


Corollary Mult_close : ∀ x y, x ∈ R -> y ∈ R -> (x · y) ∈ R.

Local Hint Resolve Mult_close : real.

Corollary Mult_Co1 : ∀ x, x ∈ (R ~ [0]) -> (∀ y, y ∈ R -> y · x = y) -> x = 1.

Corollary Mult_Co2 : ∀ x, x ∈ (R ~ [0])
    -> (∃! x0, x0 ∈ (R ~ [0]) /\ x · x0 = 1).

Corollary Mult_inv1 : ∀ a, a ∈ (R ~ [0]) -> (a⁻) ∈ (R ~ [0]).

Corollary Mult_inv2 : ∀ a, a ∈ (R ~ [0]) -> a · (a⁻) = 1.

Corollary Divide_P1 : ∀ a, a ∈ (R ~ [0]) -> a / a = 1.

Corollary Divide_P2 : ∀ a, a ∈ R -> a / 1 = a.

Local Hint Resolve Mult_inv1 Mult_inv2 Divide_P1 Divide_P2 : real.

Corollary Mult_Co3 : ∀ a x b, a ∈ (R ~ [0]) -> x ∈ R -> b ∈ R
    -> a · x = b -> x = b · (a⁻).


Corollary PlusMult_Co1 : ∀ x, x ∈ R -> x · 0 = 0.

Corollary PlusMult_Co2 : ∀ x y, x ∈ R -> y ∈ R -> x · y = 0 -> x = 0 \/ y = 0.

Corollary PlusMult_Co3 : ∀ x, x ∈ R -> -x = (-(1)) · x.

Corollary PlusMult_Co4 : ∀ x, x ∈ R -> (-(1)) · (-x) = x.

Corollary PlusMult_Co5 : ∀ x, x ∈ R -> (-x) · (-x) = x · x.

Corollary PlusMult_Co6 : ∀ x, x ∈ (R ~ [0]) <-> (x⁻) ∈ (R ~ [0]).


Corollary Order_Co1 : ∀ x y, x ∈ R -> y ∈ R -> x < y \/ y < x \/ x = y.

Corollary Order_Co2 : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R
    -> (x < y /\ y ≤ z) \/ (x ≤ y /\ y < z) -> x < z.


Corollary OrderPM_Co1 : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R
    -> x < y -> x + z < y + z.

Corollary OrderPM_Co2a : ∀ x, x ∈ R -> 0 < x -> (-x) < 0.

Corollary OrderPM_Co2b : ∀ x, x ∈ R -> 0 ≤ x -> (-x) ≤ 0.
Corollary OrderPM_Co3 : ∀ x y z w, x ∈ R -> y ∈ R -> z ∈ R
    -> w ∈ R -> x ≤ y -> z ≤ w -> x + z ≤ y + w.
```

```
Corollary OrderPM_Co4 : ∀ x y z w, x ∈ R -> y ∈ R -> z ∈ R
    -> w ∈ R -> x ≤ y -> z < w -> x + z < y + w.

Corollary OrderPM_Co5 : ∀ x y, x ∈ R -> y ∈ R
    -> (0 < x /\ 0 < y) \/ (x < 0 /\ y < 0) -> 0 < x · y.

Corollary OrderPM_Co6 : ∀ x y, x ∈ R -> y ∈ R -> x < 0 -> 0 < y -> x · y < 0.

Corollary OrderPM_Co7a : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R -> x < y
    -> 0 < z -> x · z < y · z.

Corollary OrderPM_Co7b : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R -> x ≤ y
    -> 0 ≤ z -> x · z ≤ y · z.

Corollary OrderPM_Co8a : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R -> x < y
    -> z < 0 -> y · z < x · z.

Corollary OrderPM_Co8b : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R -> x ≤ y
    -> z ≤ 0 -> y · z ≤ x · z.

Corollary OrderPM_Co9 : 0 < 1.

Local Hint Resolve OrderPM_Co9 : real.

Corollary OrderPM_Co10 : ∀ x, x ∈ R -> 0 < x -> 0 < (x⁻).

Corollary OrderPM_Co11 : ∀ x y, x ∈ R -> y ∈ R -> 0 < x -> x < y
    -> 0 < (y⁻) /\ (y⁻) < (x⁻).


Definition Upper X c := X ⊂ R /\ c ∈ R /\ (∀ x, x ∈ X -> x ≤ c).
Definition Lower X c := X ⊂ R /\ c ∈ R /\ (∀ x, x ∈ X -> c ≤ x).

Definition Bounded X := ∃ c1 c2, Upper X c1 /\ Lower X c2.

Definition Max X c := X ⊂ R /\ c ∈ X /\ (∀ x, x ∈ X -> x ≤ c).
Definition Min X c := X ⊂ R /\ c ∈ X /\ (∀ x, x ∈ X -> c ≤ x).

Property Max_Unique : ∀ X c1 c2, Max X c1 -> Max X c2 -> c1 = c2.

Property Min_Unique : ∀ X c1 c2, Min X c1 -> Min X c2 -> c1 = c2.

Definition Sup1 X s := Upper X s /\ (∀ s1, s1 ∈ R -> s1 < s
    -> (∃ x1, x1 ∈ X /\ s1 < x1)).
Definition Sup2 X s := Min (\{ λ u, Upper X u \}) s.

Property Sup1_equ_Sup2 : ∀ X s, Sup1 X s <-> Sup2 X s.

Definition Inf1 X i := Lower X i /\ (∀ i1, i1 ∈ R -> i < i1
    -> (∃ x1, x1 ∈ X /\ x1 < i1)).
Definition Inf2 X i := Max (\{ λ u, Lower X u \}) i.

Property Inf1_equ_Inf2 : ∀ X i, Inf1 X i <-> Inf2 X i.

Theorem SupT : ∀ X, X ⊂ R -> X <> Φ -> (∃ c, Upper X c) -> ∃! s, Sup1 X s.

Theorem InfT : ∀ X, X ⊂ R -> X <> Φ -> (∃ c, Lower X c) -> ∃! i, Inf1 X i.


Definition IndSet X := X ⊂ R /\ (∀ x, x ∈ X -> (x + 1) ∈ X).

Proposition IndSet_P1 : ∀ X, X <> Φ -> (∀ x, x ∈ X -> IndSet x)
    -> IndSet (∩X).
```

```
Definition N := ∩(\{ λ u, IndSet u /\ 1 ∈ u \}).

Property N_Subset_R : N ⊂ R.

Property one_in_N : 1 ∈ N.

Property zero_not_in_N : 0 ∉ N.

Property IndSet_N : IndSet N.

Local Hint Resolve N_Subset_R one_in_N : real.

Theorem MathInd : ∀ E, E ⊂ N -> 1 ∈ E -> (∀ x, x ∈ E -> (x + 1) ∈ E)
    -> E = N.

Proposition Nat_P1a : ∀ m n, m ∈ N -> n ∈ N -> (m + n) ∈ N.

Proposition Nat_P1b : ∀ m n, m ∈ N -> n ∈ N -> (m · n) ∈ N.

Local Hint Resolve Nat_P1a Nat_P1b : real.

Proposition Nat_P2 : ∀ n, n ∈ N -> n <> 1 -> (n - 1) ∈ N.

Proposition Nat_P3 : ∀ n, n ∈ N -> Min (\{ λ u, u ∈ N /\ n < u \}) (n + 1).

Proposition Nat_P4 : ∀ m n, m ∈ N -> n ∈ N -> n < m -> (n + 1) ≤ m.

Proposition Nat_P5 : ∀ n, n ∈ N -> ~ (∃ x, x ∈ N /\ n < x /\ x < (n + 1)).

Proposition Nat_P6 : ∀ n, n ∈ N -> n <> 1
    -> ~ (∃ x, x ∈ N /\ (n - 1) < x /\ x < n).

Lemma one_is_min_in_N : Min N 1.

Proposition Nat_P7 : ∀ E, E ⊂ N -> E <> Φ -> ∃ n, Min E n.


Definition Z := N ∪ \{ λ u, (-u) ∈ N \} ∪ [0].

Property N_Subset_Z : N ⊂ Z.

Property Z_Subset_R : Z ⊂ R.

Lemma Int_P1_Lemma : ∀ m n, m ∈ N -> n ∈ N -> m < n -> (n - m) ∈ N.

Proposition Int_P1a : ∀ m n, m ∈ Z -> n ∈ Z -> (m + n) ∈ Z.

Proposition Int_P1b : ∀ m n, m ∈ Z -> n ∈ Z -> (m · n) ∈ Z.

Local Hint Resolve N_Subset_Z Z_Subset_R Int_P1a Int_P1b : real.

Proposition Int_P2 : ∀ n, n ∈ Z -> n + 0 = n /\ 0 + n = n.

Proposition Int_P3 : ∀ n, n ∈ Z -> (-n) ∈ Z /\ n + (-n) = 0 /\ (-n) + n = 0.

Proposition Int_P4 : ∀ m n k, m ∈ Z -> n ∈ Z -> k ∈ Z
    -> m + (n + k) = (m + n) + k.

Proposition Int_P5 : ∀ m n, m ∈ Z -> n ∈ Z -> m + n = n + m.


Definition Q := \{ λ u, ∃ m n, m ∈ Z /\ n ∈ (Z ~ [0]) /\ u = m / n \}.

Property Z_Subset_Q : Z ⊂ Q.

Property Q_Subset_R : Q ⊂ R.
```

```
Proposition Frac_P1 : ∀ m n k, m ∈ R -> n ∈ (R ~ [0])
    -> k ∈ (R ~ [0]) -> m / n = (m · k) / (n · k).

Proposition Frac_P2 : ∀ m n k t, m ∈ R -> n ∈ (R ~ [0])
    -> k ∈ R -> t ∈ (R ~ [0]) -> (m / n) · (k / t) = (m · k) / (n · t).

Proposition Rat_P1a : ∀ x y, x ∈ Q -> y ∈ Q -> (x + y) ∈ Q.

Proposition Rat_P1b : ∀ x y, x ∈ Q -> y ∈ Q -> (x · y) ∈ Q.

Local Hint Resolve Z_Subset_Q Q_Subset_R Rat_P1a Rat_P1b : real.

Proposition Rat_P2 : ∀ x, x ∈ Q -> x + 0 = x /\ 0 + x = x.

Proposition Rat_P3 : ∀ n, n ∈ Q -> (-n) ∈ Q /\ n + (-n) = 0 /\ (-n) + n = 0.

Proposition Rat_P4 : ∀ x y z, x ∈ Q -> y ∈ Q -> z ∈ Q
    -> x + (y + z) = (x + y) + z.

Proposition Rat_P5 : ∀ x y, x ∈ Q -> y ∈ Q -> x + y = y + x.

Proposition Rat_P6 : ∀ x, x ∈ Q -> x · 1 = x /\ 1 · x = x.

Proposition Rat_P7 : ∀ x, x ∈ (Q ~ [0]) -> (x⁻) ∈ Q /\ x · (x⁻) = 1.

Proposition Rat_P8 : ∀ x y z, x ∈ Q -> y ∈ Q -> z ∈ Q
    -> x · (y · z) = (x · y) · z.

Proposition Rat_P9 : ∀ x y, x ∈ Q -> y ∈ Q -> x · y = y · x.

Proposition Rat_P10 : ∀ x y z, x ∈ Q -> y ∈ Q -> z ∈ Q
    -> (x + y) · z = (x · z) + (y · z).

Definition Even n := ∃ k, k ∈ Z /\ n = (1 + 1) · k.
Definition Odd n := ∃ k, k ∈ Z /\ n = (1 + 1) · k + 1.

Proposition Even_and_Odd_P1 : ∀ n, n ∈ N -> Even n \/ Odd n.

Lemma Even_and_Odd_P2_Lemma : ∀ m n, m ∈ Z -> n ∈ Z -> n < m -> (n + 1) ≤ m.

Proposition Even_and_Odd_P2 : ∀ n, n ∈ Z -> ~ (Even n /\ Odd n).

Proposition Even_and_Odd_P3 : ∀ r, r ∈ N -> Even (r · r) -> Even r.

Lemma Existence_of_irRational_Number_Lemma1 : ∀ a b, a ∈ R -> b ∈ R
    -> a <> b -> (((b · b) - (a · a)) < ((1 + 1) · b · (b - a))).

Lemma Existence_of_irRational_Number_Lemma2 : ∀ x y, x ∈ R -> y ∈ R -> x < y
    -> ∃ r, r ∈ R /\ x < r /\ r < y.

Lemma Existence_of_irRational_Number_Lemma3 : ∀ x y, x ∈ R -> y ∈ R -> 0 < y
    -> (y · y) < x -> ∃ r, r ∈ R /\ y < r /\ 0 < r /\ (r · r) < x.

Lemma Existence_of_irRational_Number_Lemma4 : ∀ x y, x ∈ R -> y ∈ R -> 0 < y
    -> 0 < x -> x < (y · y) -> ∃ r, r ∈ R /\ r < y /\ 0 < r /\ x < (r · r).

Lemma Existence_of_irRational_Number_Lemma5 : ∀ r, r ∈ R -> 0 < r
    -> (∃ x, x ∈ R /\ 0 < x /\ x · x = r).

Lemma Existence_of_irRational_Number_Lemma6 : ∀ r, r ∈ Q -> 0 < r
    -> ∃ a b, a ∈ N /\ b ∈ N /\ r = a / b.

Theorem Existence_of_irRational_Number : (R ~ Q) <> Φ.
```

```
Proposition Arch_P1 : ∀ E, E ⊂ N -> E <> Φ -> Bounded E -> ∃ n, Max E n.

Proposition Arch_P2 : ~ ∃ n, Upper N n.

Lemma Arch_P3_Lemma : ∀ m n, m ∈ Z -> n ∈ Z -> n < m -> (n + 1) ≤ m.

Proposition Arch_P3a : ∀ E, E ⊂ Z -> E <> Φ -> (∃ x, Upper E x)
    -> ∃ n, Max E n.

Proposition Arch_P3b : ∀ E, E ⊂ Z -> E <> Φ -> (∃ x, Lower E x)
    -> ∃ n, Min E n.

Proposition Arch_P4 : ∀ E, E ⊂ N -> E <> Φ -> Bounded E -> ∃ n, Min E n.

Proposition Arch_P5 : ~ (∃ n, Lower Z n) /\ ~ (∃ n, Upper Z n).

Proposition Arch_P6 : ∀ h x, h ∈ R -> 0 < h -> x ∈ R
    -> (∃! k, k ∈ Z /\ (k - 1) · h ≤ x /\ x < k · h).

Proposition Arch_P7 : ∀ x, x ∈ R -> 0 < x
    -> (∃ n, n ∈ N /\ 0 < 1 / n /\ 1 / n < x).

Proposition Arch_P8 : ∀ x, x ∈ R -> 0 ≤ x -> (∀ n, n ∈ N -> x < 1 / n)
    -> x = 0.

Proposition Arch_P9 : ∀ a b, a ∈ R -> b ∈ R -> a < b
    -> (∃ r, r ∈ Q /\ a < r /\ r < b).

Proposition Arch_P10 : ∀ x, x ∈ R -> (∃! k, k ∈ Z /\ k ≤ x /\ x < k + 1).


Definition Abs := \{\ λ u v, u ∈ R
    /\ ((0 ≤ u /\ v = u) \/ (u < 0 /\ v = (-u))) \}\.

Property Abs_is_Function : Function Abs /\ dom(Abs) = R
    /\ ran(Abs) = \{ λ x, x ∈ R /\ 0 ≤ x \}.

Property Abs_in_R : ∀ x, x ∈ R -> ∣x∣ ∈ R.

Local Hint Resolve Abs_in_R : real.

Property me_zero_Abs : ∀ x, x ∈ R -> 0 ≤ x -> ∣x∣ = x.

Property le_zero_Abs : ∀ x, x ∈ R -> x ≤ 0 -> ∣x∣ = -x.

Proposition Abs_P1 : ∀ x, x ∈ R -> 0 ≤ ∣x∣ /\ (∣x∣ = 0 <-> x = 0).

Proposition Abs_P2 : ∀ x, x ∈ R -> ∣x∣ = ∣(-x)∣ /\ -∣x∣ ≤ x /\ x ≤ ∣x∣.

Proposition Abs_P3 : ∀ x y, x ∈ R -> y ∈ R -> ∣(x · y)∣ = ∣x∣ · ∣y∣.

Proposition Abs_P4 : ∀ x y, x ∈ R -> y ∈ R -> 0 ≤ y
    -> ∣x∣ ≤ y <-> (-y ≤ x /\ x ≤ y).

Proposition Abs_P5 : ∀ x y, x ∈ R -> y ∈ R -> ∣(x + y)∣ ≤ ∣x∣ + ∣y∣
    /\ ∣(x - y)∣ ≤ ∣x∣ + ∣y∣ /\ ∣(∣x∣ - ∣y∣)∣ ≤ ∣(x - y)∣.

Proposition Abs_P6 : ∀ x y, x ∈ R -> y ∈ R -> ∣(x - y)∣ = 0 <-> x = y.

Proposition Abs_P7 : ∀ x y, x ∈ R -> y ∈ R -> ∣(x - y)∣ = ∣(y - x)∣.

Proposition Abs_P8 : ∀ x y z, x ∈ R -> y ∈ R -> z ∈ R
    -> ∣(x - y)∣ ≤ ∣(x - z)∣ + ∣(z - y)∣.

Definition Distance x y := ∣(x - y)∣.
```

```
Notation "] a , b [" := (\{ λ x, x ∈ R /\ a < x /\ x < b \})
    (at level 5, a at level 0, b at level 0).

Notation " [ a , b ] " := (\{ λ x, x ∈ R /\ a ⩽ x /\ x ⩽ b \})
    (at level 5, a at level 0, b at level 0).

Notation "] a , b ] " := (\{ λ x, x ∈ R /\ a < x /\ x ⩽ b \})
    (at level 5, a at level 0, b at level 0).

Notation " [ a , b [" := (\{ λ x, x ∈ R /\ a ⩽ x /\ x < b \})
    (at level 5, a at level 0, b at level 0).

Notation "] a , +∞ [" := (\{ λ x, x ∈ R /\ a < x \})
    (at level 5, a at level 0).

Notation " [ a , +∞ [" := (\{ λ x, x ∈ R /\ a ⩽ x \})
    (at level 5, a at level 0).

Notation "] -∞ , b ] " := (\{ λ x, x ∈ R /\ x ⩽ b \})
    (at level 5, b at level 0).

Notation "] -∞ , b [" := (\{ λ x, x ∈ R /\ x < b \})
    (at level 5, b at level 0).

Notation "]-∞ , +∞[" := (R) (at level 0).

Definition Neighbourhood x δ := x ∈ R /\ δ ∈ R /\ x ∈ ] (x - δ), (x + δ) [.

End real_numbers.
```


Global Index

R

Library Index

R

Global Index A B C D E F G H I J K L M N O P Q R S T U V W X Y Z _ other (1 entry)
Library Index A B C D E F G H I J K L M N O P Q R S T U V W X Y Z _ other (1 entry)