```
theory PoTD-15
  imports
    HOL−Analysis.Analysis
    HOL−Library.Quadratic-Discriminant
begin
```

Let a be a positive real number. Define a sequence x_n by

$x_0 = 0,\ x_{n+1} = a + {x_n}^2,\ n \geq 0.$

Find a necessary and sufficient condition on a for a finite limit $\lim_{n\to\infty} x_n$ existing.

```
context
  fixes a :: real
  assumes a > 0
begin
```

We will define the sequence as a function $x : \mathbb{N} \to \mathbb{R}$.

```
fun x :: nat ⇒ real where
  x 0 = 0 |
  x (Suc n) = a + (x n)^2

lemma x-nonneg: x n ≥ 0
  using ‹a > 0› by (induction n, auto)

lemma x-incseq: x n ≤ x (Suc n)
proof (induction n)
  case 0
  from ‹a > 0› show x 0 ≤ x (Suc 0) by simp
next
  case (Suc k)
  from ‹x k ≤ x (Suc k)›
  have (x k)^2 ≤ (x (Suc k))^2
    using x-nonneg by (smt (z3) power2-le-imp-le)
  hence a + (x k)^2 ≤ a + (x (Suc k))^2 by auto
  thus x (Suc k) ≤ x (Suc (Suc k)) by auto
qed

theorem convergent x ⟷ a ≤ 1/4
proof
  assume convergent x
  then obtain L where x ⟶ L
    using convergent-def by auto
  define f where f u = a + u^2 for u :: real
  have continuous (at u) f for u :: real
    unfolding f-def by auto
  with ‹x ⟶ L› have (f ∘ x) ⟶ f L
    using continuous-at-sequentially by auto
    — continuous-at-sequentially is the theorem usually known as sequential continuity.
  moreover have (f ∘ x) ⟶ L
  proof −
    have f ∘ x = (λn. x (Suc n))
      using f-def by auto
      — i.e. f ∘ x is the same sequence as x, but without the first element.
    thus (f ∘ x) ⟶ L
      using ‹x ⟶ L› and LIMSEQ-Suc by fastforce
  qed
  ultimately have f L = L
    by (rule LIMSEQ-unique)
  hence L^2 − L + a = 0
    unfolding f-def by simp
  hence discrim 1 (−1) a ≥ 0
    using discriminant-negative[of 1 −1 a] by fastforce
  thus a ≤ 1/4
    unfolding discrim-def by simp
```

```
next
  assume a ≤ 1/4
  have x n ≤ 1/2 for n
  proof (induction n)
    case 0
    then show x 0 ≤ 1/2 by simp
  next
    case (Suc k)
    from ‹x k ≤ 1/2›
    have (x k)^2 ≤ (1/2)^2
      using x-nonneg by (smt (z3) power2-le-imp-le)
    hence a + (x k)^2 ≤ 1/2
      using ‹a ≤ 1/4› by (simp add: power2-eq-square)
    then show x (Suc k) ≤ 1/2 by simp
  qed
  with x-incseq obtain L where x ⟶ L and ∀ n. x n ≤ L
    using incseq-convergent by (blast intro!: incseq-SucI)
  thus convergent x by (auto simp add: convergent-def)
qed

end
end
```