Proof Assistants – TP. 6

Bruno Barras

Oct 18, 2018

1 Even numbers

Consider the definition of even numbers

```
Inductive even : nat → Prop :=
| even0 : even 0
| evenS n : even n → even (S (S n)).
```

1- Does Coq produce the even_rect elimination scheme? Why?
2- Prove:

```
Lemma even_is_double : forall n, even n → exists m, n=m+m.
```

by induction on (even n).
We now want to prove:

```
Lemma half : forall n, even n → {m | n=m+m}.
```

(Notation { _ | _ } stands for the constant sig)
3- What is the difference in the meaning of the 2 statements even_is_double and half
4- Can we prove it in the same way? Prove it by induction on n. Beware that the direct subterm of an even number is not even!
5- Perform the extraction

```
Extraction half.
```

and compare with the definition of half
6- Is this property provable?

```
Lemma even_rect : forall P:forall n, even n → Type,
  P 0 even0 →
  (forall n (e:even n), P n e → P (S (S n) (evenS n e))) →
  forall n (e:even n), P n e.
```

Is it equivalent to even_rect that Coq might have generated automatically?

2 Division

1- Prove the specification of the euclidean division:

```
Require Import Omega.
Lemma diveucl: forall a b, b > 0 → {q:nat & {r:nat | a = b * q + r ∧ r < b}}.
```

(use the tactic omega to quickly dispatch arithmetical properties.)
2- Extract the program
3- What happens when we do the following extraction ?

```
  Extraction (diveucl 3 0)
```

Can you guess why ?