Refleksja, Specyfikacje

12 maja 2009

Plan

- Technika dowodów przez „Refleksję”
- Specyfikacje funkcji
- Ekstrakcja programów z (dowodów spełnialności) specyfikacji

Dowodzenie przez „Refleksję”

Refleksja — od *reflexion* — odbicie (np. w lustrze)

$\Longrightarrow$ Dowody bezpośrednio przez obliczenie:

- Aby dowodzić `C t` dla `C : T -> Prop`
- robimy funkcję `decide_C : T -> bool`
- dowodzimy o niej

```
Lemma decide_ok : forall t, decide_C t = true -> C t
```

- Aby udowodnić `C t` robimy
  `apply decide_ok`,
- dostajemy goal
  `decide_C t = true`
- robimy
  `reflexivity`
- i już.

Dowodzenie przez refleksje (c.d.)

⟹ Dowody przez obliczenia algebraiczne:

- Aby dowodzić `t=t'`
- robimy „sztuczną dziedzinę” `A` i funkcję `interp : A -> T`
- oraz funkcję `simplify : A -> A`
- dowodzimy o niej

```
Lemma simplify_ok : forall t,
  interp t = interp (simplify t)
```

- Aby udowodnić `t=t'`:
- Zgadujemy odpowiednie `a` i `a'`, robimy
  `change (interp a)=(interp a')`,
- potem
  `rewrite simplify_ok`
- dostajemy goal `(interp (simplify a))=(interp (simplify a))`
- robimy
  `reflexivity`
- i już :)