

Obliczenia i wnioskowanie w systemie Coq

Lista 4, termin: 30.04.2013

Zadanie 1(3p.)

Zapoznaj się z indukcyjnym typem `N` służącym do binarnej reprezentacji liczb naturalnych w Coqu. Zdefiniuj predykat `sorted : list N -> Prop` określający, kiedy lista jest posortowana rosnąco. Następnie napisz funkcję sortowania przez wstawianie o następującej specyfikacji:

```
Definition insertsort : forall l : list N,  
  {l' : list N | Permutation l l' /\ sorted l'}.
```

Predykat `Permutation` określa, kiedy jedna lista jest permutacją drugiej i jest zdefiniowany w module `Permutation`.

Zadanie 2(3+6p.)

Zdefiniuj typ `form` do reprezentacji formuł logicznych w logice zdań, zawierających spójniki koniunkcji, alternatywy, negacji i implikacji. Do reprezentowania zmiennych zdaniowych możesz użyć typu `string` z biblioteki `String`.

- Napisz funkcję `eval`, która dla zadanej formuły i waluacji zmiennych zdaniowych w tej formule, zwraca `True` lub `False` (w `Prop`) w zależności od prawdziwości formuły.
- Udowodnij, że prawdziwość formuły przy zadanej waluacji jest rozstrzygalna:
`forall (f:form) (v:valuation), {eval f v} + {~ eval f v}`
- (Dla chętnych) Zastanów się, jak udowodnić twierdzenie:
`forall (f:form), {tauto f} + {~ tauto f}`
gdzie `tauto : form -> Prop` jest predykatem określającym bycie tautologią.