

Lista zadań nr 12

Zadania dotyczące dowodów indukcyjnych można prezentować na tablicy (dowody tradycyjne), lub przy użyciu asystenta dowodzenia Coq/Rocq. W tychże zadaniach możesz założyć, że wszystkie funkcje się zawsze zatrzymują. Zadania programistyczne można rozwiązywać w OCamlu, lub w Coq/Rocqu.

Zadanie 1. (2 pkt)

Przypomnij sobie definicję funkcji `map`. Następnie pokaż, że dla dowolnych funkcji f i g oraz listy xs zachodzi $\text{map } f (\text{map } g \ xs) = \text{map } (\lambda x. f (g \ x)) \ xs$.

Zadanie 2. (2 pkt)

Pokaż, że dla dowolnej listy xs zachodzi $\text{rev } (\text{rev } xs) = xs$.

Zadanie 3. (3 pkt)

Pokaż, że dla dowolnej listy xs zachodzi $\text{rev}' (\text{rev}' \ xs) = xs$, gdzie rev' , to funkcja odwracająca listę zdefiniowana przy pomocy funkcji `rev_append`. Nie korzystaj z równoważności obu implementacji odwracania listy, tylko sformułuj i udowodnij odpowiedni lemat o funkcji `rev_append`.

Zadanie 4. (2 pkt)

Pokaż, że (w logice intuicjonistycznej) z prawa podwójnej negacji ($\neg\neg p \rightarrow p$) wynika prawo Pierce'a ($(\neg p \rightarrow p) \rightarrow p$). Napisz dowód w formie programu funkcyjnego. Dla przypomnienia, negacja $\neg p$ odpowiada przestrzeni funkcji w typ pusty ($p \rightarrow \perp$).

Zadanie 5. (2 pkt)

Formuły w *negacyjnej postaci normalnej* to takie formuły rachunku zdań, w których wszystkie negacje znajdują się przy zmiennych zdaniowych. Dokładniej,

formuły w negacyjnej postaci normalnej składają się z koniunkcji, alternatywy i literałów, gdzie literały to zanegowane lub niezanegowane zmienne zdaniowe. Takie formuły można opisać następującym typem danych, sparametryzowanym typem opisującym zmienne.

```
type 'v nnf =  
  | NNFLit   of bool * 'v  
  | NNFConj  of 'v nnf * 'v nnf  
  | NNFDisj  of 'v nnf * 'v nnf
```

W Coqu/Rocqu taka definicja wyglądałaby następująco.

```
Inductive nnf (v : Set) : Set :=  
  | NNFLit  (b : bool) (x : v)  
  | NNFConj (f1 f2 : nnf v)  
  | NNFDisj (f1 f2 : nnf v)  
  .
```

Flaga boolowska w konstruktorze literału oznacza, czy zmienna jest zanegowana (wartość `true`), czy nie (wartość `false`).

Zdefiniuj typ wszystkich formuł rachunku zdań (niekoniecznie w negacyjnej postaci normalnej), a następnie napisz funkcję tłumaczącą formuły do negacyjnej postaci normalnej. Zadbaj o to, by używać tylko rekursji strukturalnej!

Zadanie 6. (1 pkt)

Sformułuj zasadę indukcji dla drzew binarnych (etykietowanych w wierzchołkach).