

Zadanie zaliczeniowe – Why3 Cięcie w tablicy i drzewa BST

Termin rozwiązania zadania: 6 czerwca 2013

1 Wstęp

To zadanie zaliczeniowe składa się z dwóch zadań do wykonania w Why3: cięcie i intbst. Zadania będą polegały na napisaniu odpowiednich funkcji i specyfikacji tych funkcji tak aby wszystkie obligacje dowodowe wygenerowane przez Why3 można było udowodnić automatycznie.

2 Cięcie

W pliku `ciecie.mlw` znajduje się nagłówek funkcji `ciecie a`. Funkcja ta ma sprawdzać czy w tablicy `a` istnieje cięcie, czyli taki indeks k , że $0 \leq k < \text{length } a$ i dla każdego j :

- jeśli $0 \leq j \leq k$ to $a[j] \leq a[k]$ oraz
- jeśli $k \leq j < \text{length } a$ to $a[k] \leq a[j]$.

Wynikiem funkcji ma być wartość cięcia (dowolnego, jeśli cięć jest więcej niż jedno) lub -1 jeśli cięcie nie istnieje.

3 Drzewa BST

W pliku `intbst.mlw` znajdują się definicje:

- typu `tree` drzew binarnych o wartościach całkowitych w węzłach,
- predykatu `contains t n` stwierdzającego, że w drzewie t występuje wartość n ,
- predykatu indukcyjnego `bst t` opisującego kiedy drzewo t jest BST,
- funkcji `size t` obliczającej rozmiar drzewa t .

Dodatkowo są tam dwa lematy `bst_Node_inv` i `size_nonneg`. W sesji znajdującej się w katalogu `intbst` oba te lematy są już udowodnione w Coqu i nie trzeba ich dowodzić.

Zadanie polega na napisaniu treści i specyfikacji funkcji:

1. `add n t` dodającej wartość n do drzewa BST t jeśli jej tam nie ma; wpp. wynik funkcji powinien być równy argumentowi t ,
2. `split n t` dzielącej drzewo BST t na dwa drzewa BST składające się odpowiednio z wierzchołków mniejszych i większych od n .