Master Parisien de Recherche en Informatique 2007–2008
Assistants de Preuve

TD/TP 5 - Programmation fonctionnelle

Mardi 8 janvier 2008

1 Programmation avec types dépendants

1. Développer un programme pour le tri par fusion (fonctions `split`, `merge` et `sort`) en utilisant des types incluant leur spécification.
   On pourra utiliser les définitions introduites en cours disponibles dans le fichier `cours5.v`.

2 Types indexés

On souhaite définir inductivement un type des listes triées sans passer par la définition des listes générales.

La spécification de la fonction `cons` prend en arument une liste triée l et un élément a. Elle doit de plus s'assurer que a est plus petit que les éléments de l. Pour éviter la circularité, on définit d'abord un type des listes tríées indexées par un élément x, intuitivement les objets de ce type sont des listes triées dont le plus petit élément est x. Comme la liste vide n'a pas d'élément, on indexe le type des listes tríées par un objet dans le type `option` A. La liste vide est indéxée par `None` et la liste $a :: l$ par `Some` a.

1. Définir inductivement le type des listes triées indexées par leur plus petit élément.
2. Définir la fonction d'insertion dans une liste triée en utilisant cette structure.
3. On définit ensuite le type `set` A, comme étant le couple fomé d'un index x et d'une liste triée d'index x.
   (a) Définir le type `set` en suivant ce schéma.
   (b) Définir le prédicat `in_set` d'appartenance d'un élément à un ensemble.
   (c) Définir l'objet correspondant à l'ensemble vide ainsi que la fonction d'ajout d'un élément à un ensemble.
4. On constate que l'index apparaît dans les termes extraits de ces fonctions. Pour éviter ce problème, on peut choisir d'indexer le type des listes triées par un prédicat $P : A \to \mathbf{Prop}$ tel que $P\,x$ si et seulement x est le plus petit élément de la liste.
   (a) Définir ce nouveau type (on pourra s'inspirer de la définition des ensembles finis).
   (b) Reprendre les questions précédentes avec ce nouveau type et comparer les fonctions extraites.