Programmation et Structures de Données 1995-96
TD 6
Références

Frédéric Vivien, bureau 306, e-mail: Frederic.Vivien

6 novembre 1995

1 Liste chaînée

On définit un type liste chaînée d'entiers comme suit:

```
#type mlist = Empty | cellule of int * mlist ref;;
Type mlist defined.
```

Pour cette implémentation des listes, écrire les fonction usuelles: *empty*, *cons*, *car*, *cdr*, *is_empty*, *equal* et *print*.

Écrire la fonctions *rplacd* qui remplace la queue d'une liste (*cdr*) par une nouvelle valeur et la fonction *append* qui concatène deux listes.

Définir deux listes l et m et évaluer:

```
#print l;;

#print m;;

#print (append l m);;

#print l;;

#print m;;

#append l l;;

#print l;;
```

2 Liste chaînée circulaire

Pour implanter une liste chaînée circulaire on définit le type suivant:

```
#  type clist = cellule of int * clist ref;;
Type clist defined.
```

Bien que la liste soit circulaire, un de ses éléments sera identifié comme étant sa tête. Pour pouvoir travailler en tête de liste, on accédera toujours à une liste par son dernier élément. Autrement dit, si l est une *clist*, l'évaluation de l renvoie le dernier élément de la liste...

Définir pour cette implémentation les fonctions:

- *init* qui prend un entier et renvoie la liste chaînée circulaire qui ne contient que cet entier.
- *first* qui renvoie l'entier contenu dans le premier élément de la liste.

- *last* qui renvoie l'entier contenu dans le dernier élément de la liste.
- *insert_first* qui insère un nouvel entier dans la liste devant le premier élément.
- *insert_last* qui insère un nouvel entier dans la liste derrière le dernier élément.
- *remove_first* enlève le premier élément de la liste.
- *print* affiche les entiers contenus dans la liste. Quand vous aurez implanté correctement cette fonction, appelez-moi, que vous m'expliquiez pourquoi elle fonctionne... Inutile de séchez une heure sur la question avant de m'appeler au secours.

Définir un type queue et un type pile à partir des *clist* avec toutes les fonctions que cela implique... Remarque: cela ne doit pas vous prendre plus de 5 minutes.

3 Liste circulaire doublement chaînée

Chaque élément contient une référence sur son prédécesseur et sur son successeur. Même travail.