

# ASD Groupe 2 - Devoir Surveillé 1

Clément Charpentier

20 Octobre 2012

## 1 Listes Simplement Chaînées

**Exercice 1.** *Ecrire la primitive suivant(L) pour le type abstrait des listes simplement chaînées, dans le cas d'une implémentation par tableau et dans le cas d'une implémentation par allocation dynamique.*

**Exercice 2.** *Dans le cas d'une implémentation par tableau d'une liste L, donner le type des variables suivantes :*

L.vListe  
L.vListe[i]  
L.premier  
L.premierLibre

**Exercice 3.** *Donner l'état de la liste L après l'exécution des commandes suivantes :*

```
creerListe(L);  
insererEnTete(L, 'a');  
insererEnTete(L, 'b');  
debutListe(L);  
insererSuivant(L, 'c');  
insererSuivant(L, 'd');
```

**Exercice 4.** *Ecrire les fonctions suivantes :*

nombre\_d\_occurrences (ref L : listeSC\_car; val X : car) : entier  
supprimer\_toutes\_les\_occurrences\_de\_la\_valeur\_minimum (ref L : listeSC\_entier) : vide

## 2 Liste Doublement Chaînées

**Exercice 5.** *Ecrire la primitive supprimerActuel(L), sans utiliser les autres primitives, dans le cas d'une implémentation par tableau et d'une implémentation par allocation dynamique.*

**Exercice 6.** *Ecrire la fonction parentheses(ref L listeDC\_car, var X car) : vide, qui mets entre une parenthèse ouvrante '(' et une parathèse fermante ')' chaque occurrence du caractère X.*