Travaux Dirigés d'algorithmique n°8

Cours d'Informatique de Deuxième Année

—DEUG Sciences M3-MI—

Révisions

Cette feuille de travaux dirigés rassemble des exercices donnés lors de précédents examens.

▸ **Exercice 1. (Juin 96)** *Les listes chaînées d'entiers peuvent être représentées par la structure :*

```
typedef struct Cellule {
   int x;
   struct Cellule * Suivant;
   } * ListeEntier;
```

1. *Donner une fonction* `PremierDernier` *qui prend une liste en argument et qui renvoie une liste (sans recopie), qui est la même que la première, sauf que le premier élément est devenu le dernier. Si la première liste est vide, renvoyer la liste vide.*
2. *En déduire une et une seule fonction récursive* `PairImpair` *qui prend une liste en argument et qui renvoie (sans recopie) une liste identique à la première, à ceci près que tous les éléments pairs sont au début et tous les éléments impairs à la fin. L'ordre des éléments n'a pas d'importance.*

▸ **Exercice 2. (Juin 97)** *Les listes chaînées d'entiers peuvent être représentées par la structure :*

```
typedef struct Cellule {
          int entier ;
          struct Cellule * Suivant ; } StrCellule ;
typedef StrCellule * ListeEntier ;
```

- *Ecrire une fonction* `PremVersDern` *qui prend en argument une liste* `lc` *et qui place la première cellule en dernière position de la liste ;*
- *écrire une fonction* `EstlePlusPetit` *prenant en argument une liste d'entiers* `lc`*, et un entier* `n`*, et qui teste si* `n` *est plus petit (au sens large) que tous les éléments de la liste* `lc`*;*
- *En déduire une fonction de tri (récursive) prenant en argument une liste* `lc` *et qui effectue un tri sur la liste* `lc`*.*