Travaux Dirigés N^o 6 et 7

Quelques algorithmes de tri sur des tableaux

Exercice 1

Le *tri par insertion* s'inspire de la manière dont la plupart des gens trient une poignée de cartes (Bridge, Tarot, etc.). On commence avec une main gauche vide. On retire ensuite du paquet une carte à la fois, pour l'insérer à sa bonne place dans la main gauche. Pour trouver cette bonne place, on la compare avec chacune des cartes déjà présentes dans la main gauche.

1. Spécifier et réaliser l'algorithme du tri par insertion *séquentielle* : on cherche la bonne place en parcourant séquentiellement la "main gauche", de droite à gauche ou de gauche à droite.
2. Spécifier et réaliser l'algorithme du tri par insertion *dichotomique* : on cherche la bonne place avec une méthode de recherche dichotomique.

Exercice 2

Le *tri à bulles* fonctionne de la manière suivante : on parcourt le tableau (dans un sens ou dans l'autre) et, chaque fois que l'on rencontre deux éléments consécutifs qui ne sont pas dans le bon ordre, on les permute. On réitère le traitement jusqu'à ce que le tableau soit trié. Spécifier et réaliser un algorithme de tri par cette méthode.

Exercice 3

Le paradigme "diviser pour régner" est souvent utilisé dans les méthodes de tri. Ce paradigme se traduit par les 3 étapes suivantes :

Diviser : le problème en un certain nombre de sous-problèmes.

Régner : sur les sous-problèmes en les résolvant récursivement. Par ailleurs, si la taille d'un sous-problème est assez réduite, on peut le résoudre directement.

Combiner : les solutions aux sous-problèmes en une seule solution complète pour le problème initial.

1. Spécifier et réaliser l'algorithme du *tri par fusion* :

   Diviser: Diviser le tableau T[1..n] en deux sous-tableaux de $\frac{n}{2}$ éléments (à un près).

   Régner: Trier les deux sous-tableaux récursivement à l'aide du tri par fusion.

   Combiner: Fusionner les deux sous-tableaux triés pour produire la réponse triée.

2. Spécifier et réaliser l'algorithme du *tri rapide* :

   Diviser: Le tableau T[p..r] est partitionné en deux sous-tableaux non vides T[p..q] et T[q+1..r], tels que chaque élément de T[p..q] soit inférieur ou égal à chaque élément de T[q+1..r].

   Régner: Les deux sous-tableaux sont triés par des appels récursifs au tri rapide.

   Combiner: Rien.