

Algorithmique et Structures de Données

Master Biologie structurale, bioinformatique et biotechnologies

Contôle continu

Année 2015 - 2016

Jean-Michel Dischler

1. Recherche d'un élément

1.1 Concevez un algorithme de recherche d'un élément x dans un ensemble A à n éléments (vous disposez d'une fonction de comparaison). L'ensemble A n'est pas ordonné. On utilisera un tableau.

1.2 Quelle est la complexité au pire de votre algorithme en nombre de comparaisons ?

1.3 Proposer un algorithme *dichotomique*, dans le cas où A est ordonné (ordre croissant). L'algorithme doit être *récuratif*.

Principe : on recherche le milieu du tableau, m . Si l'élément x est inférieur à $A[m]$, alors x se trouve forcément dans la partie inférieure du tableau, si x est supérieur alors il se trouve dans la partie supérieure, enfin, si $x = A[m]$ alors l'élément a été trouvé, et on s'arrête. L'algorithme s'arrête également lorsque la tranche considérée est inférieure ou égale à 1 élément.

1.4 Quelle est la complexité au pire de votre algorithme en nombre de comparaisons ?

2. Recherche du maximum et du minimum

Nous supposons ici que l'ensemble considéré ne contient pas deux fois la même valeur.

2.1 Proposez un algorithme de recherche du maximum et du minimum d'un ensemble *ordonné* de n éléments rangés dans un tableau. Quelle est la complexité ?

2.2 Proposez un algorithme naïf de recherche du maximum et du minimum d'un ensemble de n éléments *non ordonnés*. Quelle est sa complexité en nombre de comparaisons ?

2.3 Proposez un algorithme plus efficace (non récuratif), avec lequel, dans une première phase, les éléments sont comparés par paire. Quelle est sa complexité en nombre de comparaisons ?