

INFO0902-1 - Structures de données et Algorithmes
Projet 2: Résolution de problèmes

Sébastien Collette (sebastien.collette@ulb.ac.be)

Introduction et modalités

L'objectif de ce projet est d'implémenter et de comparer différentes méthodes algorithmiques de résolution de problèmes en langage C. Nous vous demandons de remettre le(s) code(s) source correspondant aux desideratas repris ci-dessous, ainsi qu'un bref rapport expliquant ce que vous avez fait et les problèmes rencontrés. Le tout doit être envoyé par e-mail à l'adresse `sebastien.collette@ulb.ac.be` **avant le cours théorique du 05 mai 2011**.

Comme précédemment, les fichiers en C que vous allez remettre devront être compilables sur les ordinateurs du réseau "candi". Finalement, il s'agit d'un travail **personnel** !

1 Problèmes

Etant donnée une chaîne de nombres entiers positifs, nous désirons trouver et afficher une sous-chaîne respectant les critères ci-dessous. Les éléments composant la sous-chaîne ne doivent pas nécessairement être contigus. Notez que la chaîne sera initialement placée dans un vecteur. Vous avez le droit de créer toutes les structures de données temporaires que vous jugerez nécessaires, l'objectif étant d'obtenir un algorithme dont le pire cas du temps d'exécution soit le meilleur possible.

1.1 Palindrome (10 points)

On recherche la plus longue sous-chaîne qui soit un palindrome, c'est-à-dire qui soit identique qu'on la lise de gauche à droite ou de droite à gauche. Par exemple, si la chaîne est 1, 2, 1, 3, 4, 1, 2, l'une des plus longues sous-chaînes est 2, 1, 3, 1, 2. Proposez une solution :

1. par recherche exhaustive, en énumérant toutes les sous-chaînes, puis en vérifiant, pour chaque chaîne, si elle a la propriété requise.
2. par programmation dynamique.

Comparez la complexité au pire cas de chacun de ces algorithmes.

1.2 Caractères identiques (5 points)

On recherche la plus longue sous-chaîne composée d'éléments identiques. Par exemple, si la chaîne est 1, 2, 1, 3, 1, 2, la plus longue sous-chaîne est 1, 1, 1. Réfléchissez à la meilleure méthode pour trouver la solution, et indiquez la complexité de votre algorithme.

1.3 Unimodale (5 points)

On recherche la plus longue sous-chaîne qui soit unimodale, c'est-à-dire qui soit d'abord croissante, puis décroissante. Par exemple $1, 3, 6, 2, 1$ est unimodale, alors que $1, 5, 2, 6, 2, 5, 1$ ne l'est pas.

2 Questions Bonus - + 1 point à l'examen final

Imaginez que l'on ajoute la contrainte que les éléments sont des nombres entre 1 et W . Vous obtiendrez un point complémentaire lors de l'examen de juin en proposant deux solutions pour les caractères identiques : une solution efficace dans le cas général (quand vous ne connaissez pas W) et une solution efficace quand vous savez que W est plus petit que n .