

Examen – Informatique

Documents autorisés, pas de livre, pas de calculatrice

Lundi 18 juin 2018

Durée : 2h

Exercice 1. [? points] Une chaîne de caractères est naturellement représentée par un tableau de type `char`. Par exemple la chaîne "Bonjour !" est représentée par

['B', 'o', 'n', 'j', 'o', 'u', 'r', ' ', '!', '']

1. Proposer une structure `Chaine` en C++ permettant de représenter une chaîne de caractères de longueur au plus 100. La longueur de la chaîne sera stockée dans cette structure. On rappelle qu'en C++ un caractère est codé par le type `char`.
2. Ecrire une fonction `Chaine concatenation(Chaine c1, Chaine c2)` retournant si possible la chaîne obtenue en concaténant les chaînes `c1` et `c2`. La fonction affichera un message d'erreur et retournera une chaîne de longueur 0 si la concaténation n'est pas possible. Par exemple, si `c1` représente "Bonjour " et `c2` représente "Ada !", l'appel `concatenation(c1, c2)` retournera "Bonjour Ada !".
3. Une chaîne est un palindrome si elle peut se lire indifféremment gauche à droite ou de droite à gauche. Les chaînes "laval" et "1221" sont des palindromes tandis que "algo" n'en est pas un. Ecrire une fonction `bool est_palindrome(Chaine c)` testant si la chaîne `c` est un palindrome.

On suppose l'existence d'une fonction `bool est_chiffre(char c)` permettant de décider si le `char c` représente un chiffre ou pas. Exemple :

- `est_chiffre('7')` retourne `true`;
- `est_chiffre('a')` retourne `false`.

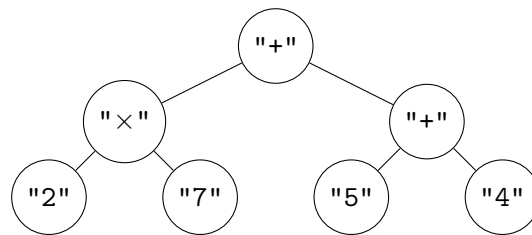
4. Ecrire une fonction `bool est_nombre(Chaine c)` qui teste si la chaîne `c` représente un nombre entier positif.

On suppose qu'il existe une fonction `int evaluer_chiffre(char c)` qui retourne la valeur entière du chiffre représenté par le caractère `c`. Par exemple `evaluer_chiffre('4')` retourne l'entier 4.

5. Ecrire une fonction `int evaluer_nombre(Chaine c)` qui étant donné une chaîne `c` représentant un nombre entier positif retourne la valeur de ce nombre.

Exercice 2.

[? points]
Un arbre arithmétique est un arbre binaire de chaînes de caractères permettant de représenter des expressions arithmétiques. Il est constitué soit de feuilles représentant des nombres entiers positifs, soit d'un noeud qui contient un opérateur arithmétique ("+", "×"), un sous arbre gauche et un sous arbre droite. L'expression $((2 \times 7) + (5 + 4))$ est ainsi représentée par l'arbre T donné par



1. Donner l'ordre de visite des sommets de T lors des parcours préfixé, infixé et postfixé.

On représente les arbres de chaînes à l'aide de la structure suivante

```
struct Arbre{
    Chaine valeur;
    Arbre* droite;
    Arbre* gauche;
};
```

où `Chaine` est introduit à l'exercice 1

Pour la suite on pourra utiliser toutes les fonctions introduites à l'exercice 1.

2. Ecrire une fonction `bool est_arithmetique(Arbre* A)` qui teste si un arbre de chaînes est arithmétique ou pas. On pourra supposer qu'on dispose des fonctions `bool est_addition(Chaine c)` et `bool est_multiplication(Chaine c)` qui teste si la chaîne `c` représente "+" ou "×".
3. Ecrire une fonction `int evalue (Arbre* A)` qui étant donné un arbre arithmétique `A` retourne la valeur qu'il représente. Par exemple, l'évaluation de l'arbre `T` retournera 23.