MT44 UTBM-année 2000/2001
TP numéro 1 Printemps 2001

ALGORITHME DE HÖRNER ET CHANGEMENT DE BASE

1. Introduction

Si b désigne un entier supérieur ou égal à deux, tout entier Q admet une unique décomposition en base b :

$$Q = \sum_{i=0}^{n} a_i b^i, \tag{1}$$

où, si Q est nul, $n = a_0 = 0$ et si Q est non nul, $n \in \mathbb{N}$, $a_n \neq 0$ et pour tout $i \in \{0, ..., n\}$, $a_i \in \{0, ..., b-1\}$. On notera cette écriture sous la forme

$$Q = \overline{a_n a_{n-1} ... a_1 a_0}. \tag{2}$$

En base 10, l'écriture d'un entier traduit donc l'égalité :

$$Q = \sum_{i=0}^{n} a_i 10^i. \tag{3}$$

Le but de ce TP est, pour un entier Q donné, de passer de son écriture en base b (1) à son écriture en base 10 (3) et réciproquement, en utilisant un algorithme de Hörner pour évaluer un polynôme en un réel.

2. L'algorithme de Hörner

Soit un polynôme à coefficient réels (ou complexes)

$$P = a_n X^n + a_{n-1} X^{n-1} + ... + a_1 X + a_0.$$

En mettant P sous la forme

$$P = ((...(((a_n X + a_{n-1})X + a_{n-2})X + a_{n-3})...)X + a_1)X + a_0,$$

on constate que l'on peut évaluer $P(t)$ en utilisant l'algorithme de Hörner (légèrement modifié par rapport à celui de votre cours) : l'entier n, et les nombres $a_n....a_0$ et t étant donnés, l'algorithme s'écrit

$s \leftarrow a_n$
Pour $i \leftarrow n-1$ jusqu'à 0,
$\quad s \leftarrow s \times t + a_i$

En fin d'algorithme, s contiendra la valeur $P(t)$.

Question 1

*Écrire une fonction matlab **evalhorner**(P,x) qui calcule $P(x)$; le polynôme P est donné sous la forme d'un tableau de réels (ou de complexes) $P = [a_n, ..., a_0]$ et x est un réel (ou un complexe).*

Il peut apparaître curieux de choisir l'ordre décroissant pour les coefficient du polynôme P mais cet ordre est celui choisi par matlab pour stocker les polynômes ; de surcroît, il est conforme à l'écriture (2). On s'assurera que la fonction est valable pour un polynôme constant.

Question 2

Quelle est la complexité de l'algorithme de Hörner ?

3. Passage de la base b à la base 10

Soit Q un entier dont on connaît l'écriture (1) en base b. On souhaite connaître son écriture (3) en base 10. Il suffit de remarquer que, si P est le polynôme $a_nX^n + a_{n-1}X^{n-1} + ... + a_1X + a_0$, alors $Q = P(b)$. Ainsi, il suffit d'appliquer l'algorithme de Hörner pour connaître l'écriture de Q en base 10.

Question 3

*Écrire une fonction matlab **transfob10**(b,A) qui renvoie l'écriture en base 10, sous la forme d'un entier, d'un entier Q. Le tableau $A = [a_n, ..., a_0]$ représente l'écriture en base b de l'entier Q.*

On pourra tester cette fonction pour $b = 10$ puis en base 2, ou sur d'autres bases.

4. Passage de la base 10 à la base b

Soit Q un entier dont on connaît l'écriture (3) en base 10. On souhaite connaître son écriture (1) en base b. Comme précédemment, on peut remarquer que, si P est le polynôme $a_nX^n + a_{n-1}X^{n-1} + ... + a_1X + a_0$, alors $Q = P(10)$ où l'addition et la multiplication (ainsi que les chiffres a_n,...,a_0 et la constante 10) sont écrits en base b. Ainsi, il suffit d'appliquer l'algorithme de Hörner pour connaître l'écriture de Q en base b, en prenant soin de considérer dans l'instruction $s \leftarrow s \times_b 10 +_b a_i$ les constantes 10, s et a_i écrites en base b ; l'addition $+_b$ et la multiplication $\times_b$ sont en base b (elles portent sur deux entiers écrits en base b). Il faut donc programmer l'addition et la multiplication directement en base b, ainsi que la conversion de a_i et de 10 en base b. Il faut d'abord transformer chacun des chiffres $a_i \in \{0, ..., 9\}$ ainsi que le nombre 10 en base b :

Question 4

Écrire une fonction matlab qui transforme un entier Q en base b ; le résultat sera stocké sous la forme d'un tableau représentant l'entier Q en base b, conformément à l'écriture (2)).

Question 5

Écrire une fonction matlab qui calcule la somme de deux entiers P et Q écrits en base b (et représentés par deux tableaux, conformément à l'écriture (2)*) sous la forme d'un tableau.*

Pour programmer cette fonction, on se rappellera comment on pose une addition en base 10, sans oublier la retenue.

Question 6

Écrire une fonction matlab qui calcule le produit de deux entiers P et Q écrits en base b (et représentés par deux tableaux, conformément à l'écriture (2)*) sous la forme d'un tableau.*

Pour programmer cette fonction, on se rappellera comment on pose une multiplication en base 10 et on décomposera une multiplication en q multiplications élémentaires par un nombre à un chiffre dont les résultats sont décalés puis sommés.

Grâce à toutes ces fonctions, vous pouvez programmer maintenant l'algorithme de Hörner directement en base b :

Question 7

*Écrire une fonction matlab **transfo10b**(b,a) qui renvoie l'écriture en base b, sous la forme d'un tableau, d'un entier a (connu en base 10).*

On pourra tester cette fonction pour $b = 10$ puis en base 2, ou sur d'autres bases. On pourra aussi vérifier que si b est un entier, a est un entier, alors $\mathit{transfob10}\big(b, \mathit{transfo10b}(b,a)\big)$ renvoie a et si A est un tableau, alors $\mathit{transfo10b}\big(b, \mathit{transfob10}(b,A)\big)$ renvoie A.

5. Consignes générales

Chacune des fonctions programmées doit être **commentée** (cf. help) : il faut savoir dans tous les cas :
- ce que fait la fonction ;
- quels sont ses arguments ;
- quelles sont les valeurs sorties.

Ces commentaires sont utiles pour le correcteurs, mais aussi pour vous !

Pour chaque TP, vous disposez de quinze jours (à compter de la dernière séance du TP sur le thème envisagé) pour rendre un rapport écrit (manuscrit ou non et contenant les éventuelles réponses aux différentes questions posées, les algorithmes utilisés, les simulations numériques faites ...) et les sources **commentées** de vos fonctions et scripts, sur disquette ou, de préférence, par mail (voir avec le professeur).

6. Comparaison de la fonction evalhorner et de la fonction polyval de matlab – Partie facultative

La fonction polyval de matlab permet d'évaluer un polynôme P en réel t ou sur une matrice X (voir help).

Question 8 – Facultatif

*Modifier la fonction **evalhorner**(P, x) pour qu'elle soit valable, comme la fonction polyval, si x est une matrice. Comparer, en utilisant tic, toc et flops, les deux fonctions, en terme de nombre d'opérations et de temps de calcul.*

7. Travail sur les chaînes – Partie facultative

Pour simplifier l'entrée ou la sortie des entiers en base b, on peut les rentrer sous forme de chaîne de caractères (par exemple, en base 12, l'entier $\overline{11\,11\,1\,4\,0\,10}$ est représenté par le tableau $[11\,,11\,,1\,,4\,,0\,,10]$ ou la chaîne de caractères 'bb140a').

Question 9 – Facultatif

*Modifier les fonctions **transfob10** et **transfo10b** pour entrer ou sortir le résultat sous la forme d'une chaîne de caractères.*

8. Conversion directe d'une base à une autre – Partie facultative

Pour convertir l'écriture un nombre d'une base b à une base b', on peut appeler les deux fonctions écrites précédemment en passant par la base 10. Il peut être plus rapide d'envisager un passage direct d'une base à l'autre.

Question 10 – Facultatif

*Écrire une fonction matlab **transfobb**(b,b',A) qui renvoie l'écriture en base b', sous la forme d'un tableau ou d'une chaîne, d'un entier Q. Le tableau $A = [a_n, ..., a_0]$ représente l'écriture en base b de Q.*

9. Comparaison de deux algorithme – Partie facultative

Il existe un autre moyen pour passer de la base 10 à la base b : on divise le nombre Q considéré (connu en base 10) successivement par b (on fait des «paquets» de b).

Question 11 – Facultatif

*En utilisant la fonction déjà programmée (en section 4) pour convertir les chiffres a_0, ..., a_n de l'écriture d'un nombre en base 10 en base b, comparer en terme de nombre d'opérations et de temps de calculs (utiliser les fonctions tic, toc et flops) cet algorithme et la fonction **transfo10b**. Que peut on en conclure ?*