EPREUVE D'ALGORITHMIQUE - MATHEMATIQUES APPLIQUEES

(Durée : 3 heures)

L'objet de ce problème est d'étudier trois méthodes utilisées par les ordinateurs pour calculer la fonction exponentielle.
Au niveau "bas", un ordinateur ne sait calculer qu'un tout petit nombre de fonctions : addition, soustraction, multiplication, division, comparaisons. Les autres fonctions sont en général calculées par programmes, grâce aux fonctions du niveau "bas", selon des méthodes semblables à celles qui font l'objet de ce problème.
Dans tout ce qui suit, "Log" désigne le logarithme népérien, et "exp" l'exponentielle en base *e*.
N.B. : les trois parties de l'énoncé sont indépendantes et peuvent être traitées dans un ordre arbitraire.

I. Calcul de la fonction exponentielle à l'aide d'une approximation polynomiale.

Dans cette partie, on cherche à approcher le mieux possible la fonction exponentielle dans [-1,1] par un polynôme. Soit N un entier supérieur ou égal à 2. Notons $\mathcal{P}_N$ l'ensemble constitué des polynômes à coefficients réels de degré inférieur ou égal à N et du polynôme nul. On cherche (s'il existe) $P^* \in \mathcal{P}_N$ tel que :

$$(1) \qquad \forall Q \in \mathcal{P}_N,\ \int_{-1}^{+1}\left(P^*(x)-e^x\right)^2 dx \le \int_{-1}^{+1}\left(Q(x)-e^x\right)^2 dx$$

Par la suite, on définira dans C [-1,1] le produit scalaire $\langle f,g\rangle = \int_{-1}^{+1} f(x)g(x)dx$, et on admettra que la famille de polynômes $(W_i)_{0\le i\le N}$ définie par :

$$\begin{cases} W_0 = 1 \\ W_1(x) = x \\ W_i(x) = \dfrac{2i-1}{i}.W_{i-1}(x).x - \dfrac{i-1}{i}.W_{i-2}(x) \text{ pour } 2 \le i \le N \end{cases}$$

est orthogonale pour le produit scalaire $\langle f,g\rangle$, et vérifie plus précisément :

$$\begin{cases} \text{si } i \ge 0, \text{ alors } W_i \text{ est de degré } i \\ \langle W_i, W_j\rangle = \begin{cases} 0 \text{ si } i \ne j \\ \dfrac{2}{2i+1} \text{ si } i = j \end{cases} \end{cases}$$

I.1) Soit W un tableau PASCAL déclaré comme suit :

```
Const N = 20; (* par exemple *)
Var W : array [0..N,0..N] of real
```

Écrire une procédure PASCAL de calcul des polynômes W_i (pour $0 \le i \le N$) qui placera dans W[i,j] le terme de degré j de W_i.

I.2) Montrer que si on définit des termes a_i ($0 \le i \le N$), par $a_i = \frac{\langle \exp, W_i \rangle}{\langle W_i, W_i \rangle}$, où exp désigne la fonction exponentielle, alors le polynôme $P^* = \sum_{i=0}^{N} a_i W_i$ satisfait la relation (1).

I.3) Écrire une procédure PASCAL qui calcule les coefficients du polynôme P* défini à la question précédente.

I.4) On suppose que le polynôme P* est mémorisé dans un tableau T de type :

```
Type poly = array [0..N] of real
```

T[i] contiendra le coefficient de degré i de P*. Dans [-1,1], on utilisera P*(x) comme approximation de e^x. Écrire une fonction PASCAL qui calcule P*(x) pour tout réel compris entre -1 et 1 (on essaiera de faire le moins possible d'opérations arithmétiques).

I.5) Écrire une fonction PASCAL qui calcule une approximation de e^x pour tout réel x. On utilisera au mieux la propriété $e^{x+1} = e.e^x$, et on supposera déclarée une constante PASCAL : Const E = 2.718281828 (* valeur approchée *).

II. Une autre méthode.

On va maintenant mettre au point une deuxième méthode de calcul de l'exponentielle, basée sur la relation suivante :

$$\left(1+\frac{x}{p}\right)^p \xrightarrow[p\to\infty]{} e^x$$

II.1) Écrire la fonction PASCAL : `function approx (x: real; n : integer) : real`

qui retourne, en fonction de x et n, la valeur $\left(1+\frac{x}{2^n}\right)^{2^n}$, en effectuant un nombre de multiplications qui est une fonction affine de n.

II.2) Définissons une fonction r_n, qui mesurera l'erreur relative commise en approchant e^x par $\left(1+\frac{x}{2^n}\right)^{2^n}$, par :

$$e^x = \left(1+\frac{x}{2^n}\right)^{2^n} (1+r_n(x))$$

Donner, pour $x \in [0,1]$ et $n \ge 1$, un majorant de $r_n(x)$. En déduire une procédure calculant l'exponentielle d'un réel x appartenant à [0,1] avec une erreur $r_n(x)$ majorée par ε, où ε est un réel strictement positif quelconque.

III. Une méthode n'utilisant pas de multiplications.

On va maintenant étudier une méthode, très utilisée par les circuits intégrés de calcul des fonctions mathématiques, qui permet de calculer l'exponentielle sans effectuer une seule multiplication (en fait, on se permettra uniquement les multiplications par une puissance de 2, qui, sur un ordinateur qui représente les nombres en base 2, se réduisent à un simple décalage des chiffres vers la droite ou vers la gauche).

On pose, pour $i \in \mathbb{N}^*$, $l_i = \text{Log}(1+2^{-i})$. On suppose que l'on dispose d'une fonction PASCAL :

```
function décale (x: real; n: integer) : real
```

telle que décale(x,n) = $x.2^n$. Soit N un entier supérieur ou égal à 2, on suppose également que les N premiers termes l_i sont mémorisés dans un tableau de réels L (on a L[i] = l_i pour $0 \le i \le N-1$).

III.1) Montrer que la série $\sum_{i=0}^{\infty} l_i$ est convergente, et montrer que pour tout i, $l_{i+1} \ge \frac{1}{2} l_i$.

En déduire que pour tout i, $l_i \le \sum_{k=i+1}^{\infty} l_k$.

III.2) En déduire que pour tout u_0 compris entre 0 et $\sum_{i=0}^{\infty} l_i$, la suite u_i définie par

$u_{i+1} = \begin{cases} u_i \text{ si } u_i < l_i \\ u_i - l_i \text{ si } u_i \ge l_i \end{cases}$ converge vers zéro.

III.3) Soit u_0 compris entre 0 et $\sum_{i=0}^{\infty} l_i$, trouver la limite de la suite de terme général v_i, définie par :

$$\begin{cases} v_0 = 1 \\ \text{si } u_i < l_i \text{ alors } \begin{cases} u_{i+1} = u_i \\ v_{i+1} = v_i \end{cases} \\ \text{si } u_i \ge l_i \text{ alors } \begin{cases} u_{i+1} = u_i - l_i \\ v_{i+1} = v_i + 2^{-i} v_i \end{cases} \end{cases}$$

III.4) En déduire une fonction PASCAL calculant une approximation A de l'exponentielle de tout réel u_0 compris entre 0 et $\sum_{i=0}^{\infty} l_i$. Donner un encadrement simple de $\frac{e^{u_0}}{A}$.