

Final Automne - 16 Janvier 2006  
Durée: 2 heures (08h00 - 10h00)  
Aucun document autorisé

## Algorithmique et Programmation : Niveau 1 (LO11)

---

### 1 Questions de Cours

#### 1.1 Passage de paramètres dans les procédures et les fonctions

Quelle est la méthode de passage de paramètres retenue en C? Donnez son principe en utilisant par exemple un schéma mémoire. (Vous pouvez vous aider d'un exemple concret)

#### 1.2 Les Types complexes en C

Quels sont les trois grandes familles de types complexes utilisables en C? A quoi servent-ils?

### 2 Exercices

#### 2.1 Qu'affiche le programme?

Vous donnerez l'affichage complet de l'exécution du programme suivant:

```
#include <stdio.h>
void f(int a, int b, int c) {
a = 5*b+c;
c = a*12+10*(b*c);
printf("Dans f : a = %d, b = %d, c = %d",a,b,c);
return; }
```

```

int A(int b, int c) {
int a = 3 ;
c = a;
b = 2*c*(a-b);
printf("Dans A : a = %d, b = %d, c = %d",a,b,c);
return b; }

int F(int b, int a, int c) {
int A;
b = a-c;
A = a*(b-c);
printf("Dans F : a = %d, b = %d, c = %d",a,b,c);
return A; }

int main() {
int a,b,c;
a = 1;
b = 1;
c = 2;

f(b,c,a);
printf("Après f : a = %d, b = %d, c = %d",a,b,c);

a=F(a-b,b,c);
printf("Après F : a = %d, b = %d, c = %d",a,b,c);

b=A(a,b);
printf("Après A : a = %d, b = %d, c = %d",a,b,c); }

```

## 2.2 Tableau et tri particulier

L'objectif de cet exercice est d'effectuer le tri d'un tableau d'entiers en mettant en premier les entiers positifs (ou nuls) dans l'ordre décroissant puis ensuite les entiers négatifs mais dans l'ordre croissant (voir figures 1 et 2).

-4	3	-6	1	5	0	-2	6
----	---	----	---	---	---	----	---

Figure 1: Tableau initial

6	5	3	1	0	-6	-4	-2
---	---	---	---	---	----	----	----

Figure 2: Tableau Final

### 2.2.1 Avec un tableau intermédiaire

Ecrivez l'algorithme permettant d'effectuer ce tri en utilisant un tableau intermédiaire pour mettre le résultat. Vous pouvez vous inspirer des différentes méthodes de tri vues en cours et en TD.

### 2.2.2 Sans un tableau intermédiaire

Ecrivez le même algorithme mais cette fois-ci sans utiliser de tableau intermédiaire. Le résultat sera donc dans le tableau de départ.

## 3 Problème : Calcul de fonctions

Certaines fonctions mathématiques sont calculées à partir de leur développement limité pour obtenir une valeur approchée la plus précise possible. Par exemple :

$$\exp(x) = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots$$

Il faut alors que  $n$  soit suffisamment grand pour que l'approximation soit la plus proche possible de la fonction à évaluer:

$$\lim_{n \rightarrow \infty} \left( \exp(x) - \left( 1 + x + \frac{x^2}{2!} + \dots + \frac{x^n}{n!} \right) \right) = 0$$

### 3.1 Calcul de la fonction exponentielle ( $\exp(x)$ )

Ecrivez l'algorithme de la fonction permettant de calculer  $\exp(x)$ . ( $x$  est un réel). On supposera que l'erreur d'évaluation  $\epsilon$  est un paramètre de la fonction.

### 3.2 Calcul d'autres fonctions

Voici les développements limités de sinus et de cosinus:

$$\sin(x) = x - \frac{x^3}{3!} + \dots + \frac{(-1)^n \cdot x^{2n+1}}{(2n+1)!}$$

$$\cos(x) = 1 - \frac{x^2}{2!} + \dots + \frac{(-1)^n \cdot x^{2n}}{(2n)!}$$

Indiquez les modifications à faire sur l'algorithme précédent pour calculer ces fonctions (Ne réécrivez pas l'intégralité de l'algorithme à chaque fois !)

### 3.3 Programme de calcul de fonctions mathématiques

On veut écrire un programme qui permet à un utilisateur de faire des calculs avec les fonctions mathématiques que l'on vient de définir. Le déroulement du programme donne le résultat suivant (les zones en gras et italique correspondent à ce que saisi l'utilisateur) :

Bienvenue dans le programme de calcul de fonctions

Voici la liste des fonctions:

1. `exp(x)`
2. `sin(x)`
3. `cos(x)`

Donnez la fonction desirée : *4*

Ce choix n'est pas valide recommencez : *2*

Fonction sinus (`sin(x)`)

Donnez x : *2.0*

Donnez la précision (>0.000001): *0.0000001*

La précision n'est pas valide. La précision Maximale (0.000001) est retenue.

Le resultat est : 0.0348994

Voulez vous faire un autre calcul (oui/non)? : *non*

Au revoir et merci

Ecrivez l'algorithme permettant d'obtenir ce résultat. Vous ferez bien attention aux boucles utilisées ainsi qu'aux éventuels messages d'erreur.

### 3.4 En C

Ecrivez le programme C correspondant. (La fonction `strcmp` de la bibliothèque `string.h` permet de faire la comparaison entre deux chaînes de caractères. Son résultat est 0 si les deux chaînes sont identiques et différent de 0 sinon)