

TD-1

Exercice 1 : passages de paramètres

Simuler l'algorithme suivant :

Algorithme Alias

Var I: entier

Procédure IncrémenterDeI (var K:entier ; var L:entier)

debut

K <- K+I

L <- L+1

fin

debut

I<-1

IncrémenterDeI(I,I)

writeln(I)

fin

Exercice 2 passages de paramètres (encore).

Le phénomène d'interférence (modification de la valeur d'une variable qui change également la valeur des ses 'alias') peut conduire à des erreurs de programmation.. Soit l'algorithme :

Algorithme Factorielle?

Var X,Y : entier

Procédure Fact (var N : entier ; var F : entier)

Var K : entier

debut

K <- 0

F <- 1

TantQue K≠N faire

K <- K+1

F <- K*F

Fintq

fin

debut

readln(X)

Fact(X,Y)

writeln(Y)

Fact(Y,Y)

writeln(Y)

fin

1) Quelles sont les valeurs affichées à l'écran quand la valeur de X lue est 3 ?

2) Comment modifier l'algorithme ci-dessus pour que les valeurs affichées à l'écran soient successivement 6 et 720 (=6!).

Exercice -3

Nous disposons d'un tableau T de N éléments. Un élément est constitué d'un champ Nom (une chaîne) et d'un champ Couleur (un caractère 'B' ou 'R'). Le problème est de réarranger les éléments de T de manière à ce que les éléments de couleur 'B' précèdent les éléments de couleur 'R'. Exemple:

(Toto,B)(Titi,R),(Tutu,B)(Tata,R),(Tyty,R),(Tete,B)

est réarrangé en:

(Toto,B),(Tete,B),(Tutu,B) , (Tata,R), (Tyty,R),(Titi,R)

Il faut de plus respecter les contraintes suivantes: 1) On n'utilisera pas de tableau auxiliaire et on procédera uniquement par échanges de deux éléments, 2) on ne regardera pas deux fois la couleur d'un élément.

Il est suggéré d'utiliser la méthode suivante: On considère trois zones successives dans le tableau: la zone des 'B', la zone des éléments de couleur non identifiée, la zone des 'R'. Au début les zones 'B' et 'R' sont vides. A la fin la zone des éléments non identifiés est vide, et le problème est résolu. Pour cela on notera b, la position dans le tableau du premier élément suivant la zone des 'B', et r la position dans le tableau du premier élément précédant la zone des 'R'.

On fera ainsi pas à pas augmenter b ou décroître r, et éventuellement échanger les éléments d'indice b et r jusqu'à ce que le problème soit résolu. Exemple: au début

(Toto,B)	(Titi,R),	(Tutu,B)	(Tata,R),	(Tyty,R),	(Tete,B)
b=1					r=6

à la fin

(Toto,B),	(Tete,B),	(Tutu,B) ,	(Tata,R),	(Tyty,R),	(Titi,R)
		r=3	b=4		

1) Ecrire une procédure Ranger(T) réarrangeant le tableau T.

2) Ecrire une procédure Ranger3(T) traitant le même problème dans le cas où il y a 3 couleurs 'B', 'R', 'V'. Il faudra distinguer 4 zones et utiliser un indice supplémentaire v représentant la position du premier élément précédant la zone des 'V'. Par exemple en cours d'exécution on peut rencontrer la configuration suivante:

< <u>B</u> >	< <u>?</u> <u>?</u> >	< <u>R</u> <u>R</u> >	< <u>V</u> >		
(Toto,B)	(Titi,R),	(Tutu,B)	(Tata,R),	(Tyty,R),	(Tete,V)
	b=2	r=3		v=5	

TP1

1) Programmez les exercices 1 et 2 et vérifiez que les simulations faites en TD sont correctes. Pour cela, on affichera les valeurs intermédiaires des variables dans un format le plus proche possible de l'aspect d'une simulation « manuelle », c'est à dire en décalant lorsqu'on entre dans une boucle, en utilisant les accolades « { } », etc ...

2) Programmez l'exercice 3, en affichant les états successifs du tableau T, si possible avec la même présentation que dans l'énoncé. Structurez le programme : il faut au minimum des procédures ou fonctions de lecture (saisie du tableau) d'écriture (affichage d'un tableau) et de rangement, et d'échange.