

TD n°8

A) Rappels :

Affectation

Quand on utilise $\mathbf{a:=b}$, on affecte à la variable \mathbf{a} la valeur de \mathbf{b} . Si \mathbf{a} et \mathbf{b} sont des entiers, valant 3 pour \mathbf{a} et 10 pour \mathbf{b} , \mathbf{a} prend la valeur 13. L'affectation modifie la valeur de \mathbf{a} , mais pas celle de \mathbf{b} .

Si l'on écrit $\mathbf{a:=a+b+1}$, alors, on affecte à \mathbf{a} la valeur de l'expression $\mathbf{a+b+1}$ (qui est $10+3+1=14$).

Comparaison

=, <, >, <= (inférieur ou égal), >= (supérieur ou égal), <> (différent).

Opérations booléennes

And, or, not.

Structures

if...then...(else) (attention, jamais de ; avant le else)

for...to...do

for...downto...do

while...do...

repeat...until

Ne pas oublier d'encadrer les groupes d'instructions par **begin** et **end**;

Fonctions prédéfinies

cos, sin, ln, exp, sqrt (racine carrée)

et des moins connues : **trunc** (partie entière), **abs** (valeur absolue), **div** (division entière d'un entier par un entier)

Exemple : **trunc(5.34)=5**

34 div 5 =6

Les variables

Les variables déclarées tout au début du programme (avant les procédures et les fonctions) sont des variables globales. Elles peuvent être utilisées n'importe où, y compris dans les procédures et les fonctions.

Les variables déclarées dans une procédure ou une fonction sont des variables locales. Elles n'existent qu'à l'intérieur de la procédure ou de la fonction.

Dans une procédure, on peut entrer deux types de paramètres (ce sont les variables entre parenthèses) :

- les paramètres appelés valeurs, qui ne seront pas modifiées, mais seront utilisées.
- Ceux appelés variables, précédées de **var** dans la déclaration de la procédure, qui peuvent être modifiées lors de la procédure.

Le stockage des variables

Ne pas oublier qu'il y a une place limitée pour une variable.

Pour des réels, les problèmes rencontrés sont des problèmes d'arrondis : le réel $1+10^{-30}$ sera arrondi à 1.

Pour des entiers, on ne peut aller que jusqu'à plus ou moins 32767. Donc, attention lors de l'utilisation de grandes factorielles !

B) Exercices :

Exercice 1 :

Calculer $\sum_{k=1}^n k^4$ pour $n=100$.

Exercice 2 (ESC Maths 2007 S)

Soit la suite : $I_{n+1} = \frac{3n-1}{3n} I_n$, avec $I_1 = \frac{2\pi}{3\sqrt{3}}$.

Ecrire un programme qui demande un entier n supérieur à 2, calcule puis affiche la valeur de I_n .

Exercice 3 :

Calculer $\sum_{k=1}^n \frac{x^k}{k!}$ pour $n=100$.

Exercice 4 (ERICOME Maths 2005 S)(modifié)

a) Faire une fonction qui demande à l'utilisateur un réel $u_0 \geq 0$ jusqu'à ce que l'utilisateur lui en donne un, puis qui renvoie ce nombre.

b) Faire une fonction suite(v,n) qui renvoie le n ème terme de la suite réelle définie par

$$u_0 = v$$
$$u_n = \sqrt{n + u_{n-1}}$$

c) Faire un programme qui demande le premier terme de la suite, un entier n et qui affiche le n ème terme de la suite.

Exercice 5 :

Calculer $\sum_{1 \leq i < j \leq n=1} \frac{1}{i+j}$ pour $n=100$.

Exercice 6 :

Calculer $\prod_{k=1}^n \left(1 - \frac{k}{356}\right)$ pour $n=100$.

Exercice 7 (EDHEC 2007)

On dispose de deux urnes U et V. L'urne U contient une boule blanche et $(n-1)$ boules noires. L'urne V contient une boule noire et $(n-1)$ boules blanches.

Un joueur choisit au hasard l'urne dans laquelle il effectue le premier tirage. Le choix de l'urne des tirages suivants dépend du résultat du tirage précédent. Tous les tirages sont avec remise dans l'urne d'origine.

On note U l'événement le premier tirage a lieu dans l'urne U. On note X le numéro du tirage où l'on obtient pour la première fois une boule blanche.

a) Ecrire le programme qui calcule et affiche X lors de l'expérience décrite ci-dessus, dans le cas où tous les tirages ont lieu dans l'urne choisie au premier tirage.

b) Même question dans le cas où l'on choisit l'urne V après avoir tiré une boule noire, U après avoir tiré une boule blanche.

c) Même question dans le cas où l'on change d'urne après avoir tiré une boule noire, et où l'on garde la même après avoir tiré une boule blanche.

Exercice 8 :

Calculer u_{88} et v_{88} sachant :

$$u_0 = 1/3$$

$$v_0 = 1$$

$$u_{n+1} = u_n - u_v + 1$$

$$v_{n+1} = 3v_n + u_n - 2$$

Exercice 9 :

Calculer le n ème ($n \geq 2$ entré au clavier) terme de la suite u définie par

$$u_0 = \frac{1}{2}, \quad u_1 = \frac{3}{2} \quad \text{et} \quad u_{n+2} = \frac{2}{3}u_{n+1} + \frac{1}{3}\min(u_n, 1)$$

Exercice 10 (ESC Maths 2006 S)

On considère un réel $\alpha > 0$ et la suite $(u_n)_{n \geq 2}$ définie par : $u_n = \frac{1}{n \ln^{\alpha+1}(n)}$.

On note, pour tout entier n supérieur ou égal à 2, $S_n = \sum_{k=2}^n u_k$.

Au cours de l'exercice, on a montré que : $0 \leq L - S_n \leq \frac{1}{\alpha \ln^\alpha(n)}$

a) Ecrire un programme Turbo-Pascal qui demande deux réels strictement positifs α et ε et affiche un entier naturel n et une somme partielle S_n tels que l'écart entre S_n et L soit inférieur à ε : (on rappelle que **trunc** est la fonction partie entière).

b) On suppose que les valeurs $\alpha = 10$ et $\varepsilon = 10^{-6}$ ont été entrées. Y aura-t-il une erreur due à un débordement à l'exécution de ce programme ? (on prendra 2^{15} comme plus grand entier possible pour le type **integer** et on donne $\ln(2) \approx 0,69$)

Exercice 11 : l'écriture en binaire (ESC Maths 2004 S)(très modifié)

Dans l'ordinateur, toutes les informations sont codées avec des suites de 0 et de 1 (ou de vrai ou faux), c'est le langage binaire. Nous allons voir comment il stocke les entiers.

a) Soit n un entier. Si l'ordinateur utilise un nombre n de 0 ou de 1 pour stocker un entier, combien d'entiers différents pourra-t-il stocker ?

b) Compléter le tableau :

binaire	décimal
0	0
1	1
10	2
11	3
100	
1000	
101001	

c) Dans nos programmes, on va utiliser un type qui correspondra aux tableaux d'entiers à 8 cases, indicées de 0 à 7. Ecrire ce type « tab_entier ».

Quel est l'entier maximal que l'on peut coder ?

d) Quelle est la formule donnant l'entier correspondant à un tableau de 0 et de 1 ?

e) Comment obtient-on le chiffre de la case 0 à partir de l'entier à coder ? (on utilisera la fonction $\text{trunc}(x)$ qui renvoie la partie entière de x).

f) Comment obtient-on le i ème chiffre à partir de l'entier à coder, et des chiffres des cases précédentes ?

g) Ecrire une fonction qui renvoie l'entier correspondant à un tableau de type tab_entier.

h) Ecrire une procédure qui code un entier n dans un tableau tn .

Exercice 12 :

Faire un programme qui simule des lancers d'un dé non truqué à 6 faces, et qui affiche la somme de n lancers, où n est choisi par l'utilisateur.

Exercice 13 :

Faire un programme qui demande un réel r entre 0 et 1, un entier n , et qui affiche les résultats de n tirages où l'on obtient A avec la probabilité r , B avec la probabilité $1-r$.