**DM n°2**  TS1 année 2011-2012

|  |  |  |
| --- | --- | --- |
| k | *ak* = 1 + 10-k | bk= ln(1+10 - k) |
| 0 | 2 | 0,693 1472 |
| 1 | 1,1 | 0,095 3102 |
| 2 | 1,01 | 0,009 9503 |
| 3 | 1,001 | 0,000 9995 |
| 4 | 1,0001 | 0,000 1 |
| 5 | 1,00001 | 0,000 01 |
| 6 | 1,000001 | 0,000 001 |
| 7 | 1,0000001 | 0,000 0001 |

**Exercice 1** : **calcul des logarithmes à l’aide de l’algorithme CORDIC utilisé dans les premières calculatrices ( COordinate Rotation for DIgital Computer ), mis au point en 1959 par l’Americain Volder.**

Principe : à l’aide d’un petit nombre de valeurs préenregistrées (en mémoire morte), calculer des valeurs approchées du logarithme de n’importe quel nombre réel strictement positif X.

1. X s’écrit en notation scientifique sous la forme X = *x* x 10p avec p∈Z et *x*∈[1,10[.

En utilisant les propriétés de la fonction ln , on obtient lnX = ln *x* + p x ln10

avec ln 10 ≈ 2,3025851

Il suffit ensuite de déterminer ln*x* à l’aide des valeurs du tableau ci-contre en utilisant une décomposition approchée de *x* sous la forme : *x* ≈ où les ci sont des constantes du type *ak* = 1 + 10-k. On obtient ainsi ln*x* à l’aide de soustractions des valeurs de la troisième colonne du tableau : ln*x* ≈ ln10 – ln c1 – ln c2 – … – ln cn

L’algorithme ci-dessous permet de déterminer une valeur approchée de ln*x*, en cherchant les ci de proche en proche en construisant une suite de nombres *xi* tels que :

*x* x se rapproche de 10 à 10-6 près.

*x1*

*x2*

… …. … … … …. ….

*xn*

|  |
| --- |
| **algorithme**  **Entrée** : x un réel compris entre 1 et 10  **Initialisation** :  Liste u ( u(0) ;u(1) ;u(2) ;u(3) ;u(4) ;u(5) ;u(6) ;u(7))←(0,6931472;0,0953102;0,0099503;0,0009995;0,0001;0,00001;0,000001;0,0000001)  k←0  y←2,3025851  **traitement**  Tant que 10-x ≥ 10-6 faire :  Si x x(1+10-k) ≤ 10 alors  *test de la multiplication de x par 2 puis 1,1 ; 1,01 ; etc … pour approcher 10 sans le dépasser*  x ← x x(1+10-k)  afficher « a » *affiche les différentes valeurs de ci sous la forme des ak ( c'est-à-dire les* (1+10-k))  afficher k *i prend n’importe quelle valeur entière et compte le nombre de ci (boucles)*  y ← y-u(k) *k prend des valeurs entières comprises entre 0 et 7.*  Sinon  k←k+1  fin du Si  fin du tant que  **sortie**  afficher « ln x =»  afficher y |

Faire fonctionner l’algorithme sans calculatrice en utilisant le papier et le crayon pour *x*= 4,455 8911

( ne pas oublier que multiplier par 1,1 revient à ajouter au nombre le même nombre décalé d’une virgule! de même pour 1,01 et deux virgules…

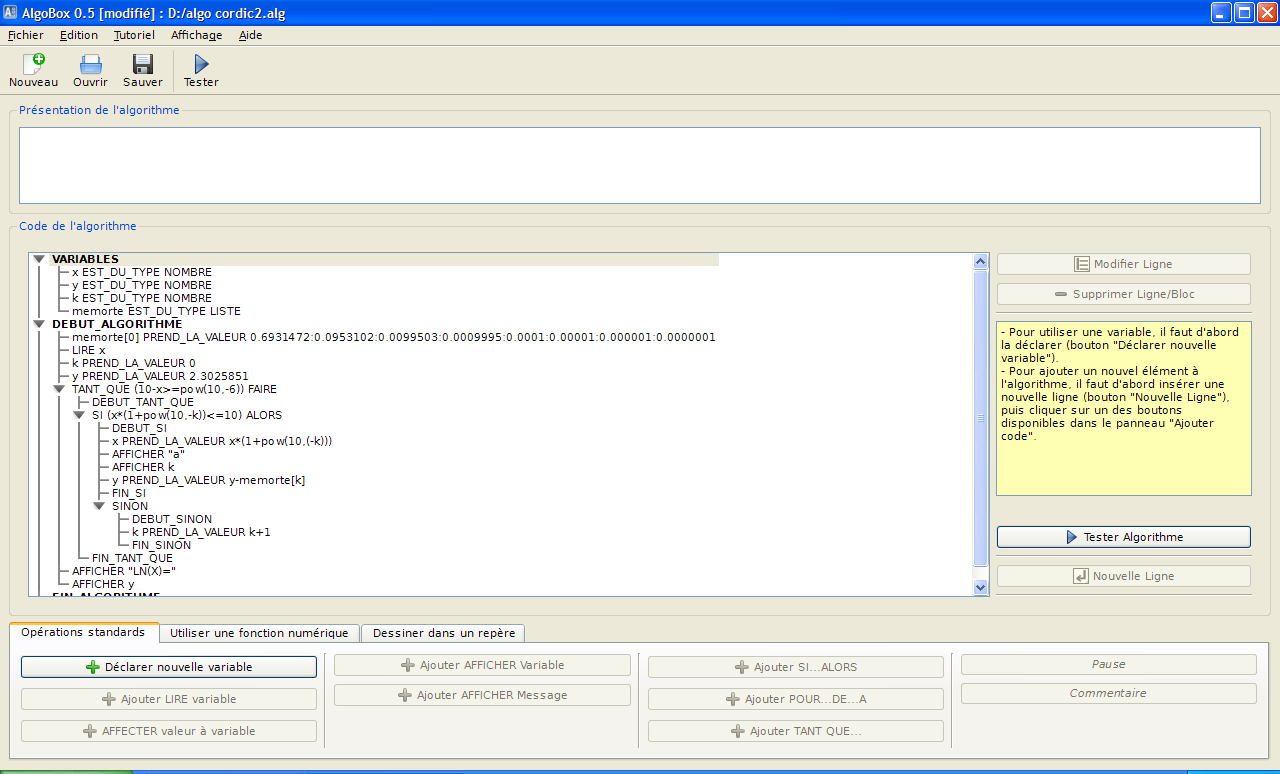
Exemple : 9,802 960 4x1,1 = 9,802 960 4 ).

+ 0,098 02 9 6

Donner le nombre d’étapes et à chaque étape les valeurs des ci et des *xi*et enfin la valeur approchée de ln 4,455 8911.

1. Utiliser le logiciel algobox pour faire fonctionner cet algorithme sur ordinateur . ( On peut le télécharger gratuitement sur internet). Vérifier les résultats obtenus au b- , puis utiliser le programme pour déterminer une valeur approchée de

ln ( 9,071847 3) , ln4 et autres valeurs … que remarque-t-on sur le nombre des ci ?



1. Comment modifier le programme pour obtenir les valeurs des logarithmes des nombres n’appartenant pas à l’intervalle

[ 1 ; 10[ ? le faire fonctionner , puis l’envoyer à Mme Boyer via les ENT …

*Remarque : Ce TD est tiré d’un exercice de mon manuel de mathématiques Terminale C , chez Didier édition 1984 … !!!!*