

Compte rendu 1 : Fonctionnement de notre calculatrice

Dans cette première séance dédiée aux mathématiques appliquées à l'informatique, on nous a demandé de calculer des valeurs, qu'on aurait très bien pu calculer à l'aide de notre calculatrice, à l'aide d'un logiciel de programmation « Scilab ».

On nous a alors demandé de calculer : $-\exp(1.37)$
 $-\sin(0.21466)$
 $-\cos(0.21466)$
 $-\ln(1.338)$

Pour cela, nous avons utilisé la théorie des séries entières et en particulier les séries de Taylor.

Avant toute chose, nous avons donc commencé par chercher les développements en entière au voisinage de 0 de ces différentes fonctions.

e^x	$\mathbb{R}$	$\sum_{n=0}^{\infty} \frac{x^n}{n!}$	$+\infty$	$\mathbb{R}$
-------	--------------	--------------------------------------	-----------	--------------

Pour cette fonction, rien de compliqué, il nous suffisait de poser $x=1.37$ dans notre programme qu'on devra modéliser pour calculer sa valeur.

$\ln(1+x)$	$] -1, +\infty[$	$\sum_{n=1}^{\infty} (-1)^{n+1} \frac{x^n}{n}$	1	$] -1, 1]$
------------	------------------	--	---	------------

Pour calculer $\ln(1.338)$, il nous suffisait de prendre $x=0.338$, donc nous aurons alors $\ln(1+0.338)$ ce qui nous permettrait d'utiliser notre formule.

$\cos x$	$\mathbb{R}$	$\sum_{n=0}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!}$	$+\infty$	$\mathbb{R}$
$\sin x$	$\mathbb{R}$	$\sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!}$	$+\infty$	$\mathbb{R}$

Pour calculer $\sin(0.21466)$ et $\cos(0.21466)$, il suffisait tout simplement d'utiliser le même principe que celui de la fonction exponentielle tout en remplaçant x par 0.21466.

Après avoir remarquer ses différents développements en séries entières, nous avons pu commencer à coder !!

Dans un premier temps, afin de nous aider par la suite, nous avons tout d'abord créé une fonction que nous avons nommé « puissance » (elle aura pour variable la valeur x et une valeur n qui déterminera la puissance affecté à notre valeur). Cette fonction nous sera très utile notamment pour calculer les puissances d'une valeur à une puissance donnée. (Sachant que toute nos développement en séries entières utilisent des puissances).

Dans le même but, nous avons aussi mis en place une fonction pour calculer les factorielles, car comme pour la puissance, tous nos développements utilisent des factorielles.

Pour mettre en place la fonction puissance, il nous a suffit de mettre une boucle « while » pour que la fonction calcule $x*x$. Et à chaque sorti de la boucle, on décrémente la valeur n de 1 jusqu'à arriver à la valeur nulle, tant que n sera différent de null, il continuera sa manœuvre. Pour la fonction factorielle, on utilise aussi une boucle « while », tant que n sera différent de null, il va continuer de faire $n*(n-1)$. Et à chaque sortie il va décrémenter n de 1.

Après avoir mis en place ces différentes fonctions, on a commençait par coder la fonction

$\exp(x)$ à l'aide de $\sum_{n=0}^{\infty} \frac{x^n}{n!}$.

Nous avons commençait par coder la fonction $\frac{x^n}{n!}$. A l'aide des fonctions « puissance » et « factorielle », il nous a suffit d'appeler simultanément ces deux fonctions pour créer cette fonction. Ensuite, nous avons créé une fonction « exponentielle » qui prend comme variable une valeur x, une valeur n et une valeur epsilon qui sera la pour déterminer la précision de notre calcul. En effet, plus l'epsilon sera grand et plus la valeur émise aura une grosse marge d'erreurs. Et plus la valeur d'epsilon sera faible, et plus le résultat trouvé sera proche du résultat réel. Dans notre cas, on nous a demandé de tester avec les epsilon suivants : 0.1 ; 0.001 ; 0.0001. Pour les trois autres fonctions, nous avons utilisé la même démarche pour les calculer. Je vais mettre tous les résultats obtenus dans le tableau ci-dessus :

Fonction exponentielle	Fonction logarithme	Fonction sinus	Fonction cosinus
exp1=3.8837902	log1=0.338	Sinus1=0.21466	Cos1=1
exp2=3.9349887	log2=0.2904866	Sinus2=0.2130114	Cos2=0.9769605
exp3=0.39352965	log3=0.2911203	Sinus3=0.2130114	Cos3=0.9769605
Dexp1=0.0515605	Dlog1=0.0468240	Dsinus1=0.0016448	Dcos1=0.0229511
Dexp2=0.0003620	Dlog2=0.0006894	Dsinus2=0.0000038	Dcos2=0.0000883
Dexp3=0.0000542	Dlog3= 0.0000556	Dsinus3=0.0000038	Dcos3=0.0000883

On remarque d'après ce tableau en effet que plus l'epsilon se rapproche de 0 et plus la valeur se rapprochait de la valeur réelle (en effet les trois dernières lignes du tableau représente respectivement la différence entre la valeur réelle et la valeur approché avec les différents epsilon 0.1, 0.001,0.0001).

D'autre part, on remarque qu'avec certaines fonctions comme par exemple, la fonction cosinus ou bien sinus, qu'à partir d'un certain epsilon ici en l'occurrence 0.001, la valeur affiché n'est plus modifiée alors que les autres fonctions telles que exponentielles et logarithme, elle continue de diminuer et de se rapprocher de la valeur réelle.

Cela est sûrement du à la particularité de la courbe. Les fonctions cosinus et sinus ont le même point commun, elles sont $2-\pi$ périodiques, et n'admettent pas de limite en infini (donc elle se répétera au cours du temps) qui n'est pas le cas des fonctions exponentielles ou logarithmes.

Conclusion : Grâce à notre logiciel, nous avons pu modéliser ces 4 fonctions et nous a permis de calculer des valeurs rapprocher sans utiliser la calculatrice. On aurait pu donner des epsilons bien plus petits afin de se rapprocher indéfiniment de la valeur exacte. Et on aurait peut être pu atteindre une valeur bien plus précise que la calculatrice.