



scilab

I - Calculs numériques non programmés

Lycée

Auteur : Raymond Moché

Mots-clefs : Calculs sans programmation, constantes pré-enregistrées, utilisation de la bibliothèque de fonctions, aide en ligne, console « scilab », éditeur de texte « SciNotes », formats d'affichage, saisie d'un tableau à la main, « sin », « cos », « tan ».

Référence :

- ✓ Livret de présentation de « scilab pour les lycées » (2010) et Mise à jour et compléments (mars 2011) :

<http://www.scilab.org/education/lycee/docs>

- ✓ Aide scilab 5.3.2

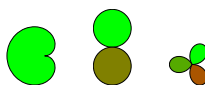
http://help.scilab.org/docs/5.3.2/fr_FR/index.html

Liste des exercices :

Énoncé n° 1 [*] : « scilab » utilisé comme une calculatrice.

Énoncé n° 2 [*] : Lignes trigonométriques remarquables.

Énoncé n° 3 [**] : Conversion de la base 2 à la base 10.



Énoncé 1 [*] « scilab » utilisé comme une calculette.

Quelle est la valeur de π ? de e ? de $\pi \cdot e$? Calculer $\frac{\sin 3 \cdot e^2}{1 + \tan \frac{\pi}{8}}$.

Mots-clefs : constantes pré-enregistrées, calcul numérique.

Commentaires : On utilise ici « scilab » comme une calculatrice ordinaire. En particulier, les règles de priorité des différentes opérations sont les règles habituelles. Bien sûr, on peut ajouter des parenthèses pour rendre le calcul plus clair. Quand les commandes à taper sont vraiment simples, on peut utiliser directement la feuille de travail « scilab » (appelée « console »), voir la figure 1 ci-dessous.

Pour trouver la valeur approchée que « scilab » donne à π avec les réglages de format d'affichage par défaut, il suffit de taper `%pi` juste après l'invite (`- - >`), puis d'utiliser la touche « Entrée » de votre ordinateur. Nous avons tapé `a=%pi` parce que nous avons désiré appeler a la variable qui prendra la valeur recherchée. Idem pour la valeur « scilab » par défaut de e , ici nommée « viventles-maths ». Nous avons oublié de nommer la variable qui prendra la valeur de $e * \pi$. Ce n'est pas grave. « scilab » lui donne un nom par défaut qui est « ans » (pour answer). Attention : si on fait une deuxième oubli de ce genre, `ans` prendra la valeur nouvellement calculée. L'ancienne valeur sera perdue. Nous avons appelée x la variable qui va

porter la valeur de « ans ». L'exécution de cette commande n'apparaît pas sur la console parce que la ligne de commande a été terminée par un « ; ». Puis, nous avons défini un tableau y à une ligne et 3 colonnes (un vecteur-ligne ou mieux une matrice (1,3) ; il est inutile de parler de ça en classe). y a été affiché par défaut dans le système décimal (16 signes dont + omis et le point). Ensuite, avec la commande « format », nous avons affiché y de diverses façons (affichage décimal à 25 signes puis affichage à virgule flottante à 10 signes).

La seule difficulté de ce calcul est donc de savoir que π (par exemple) s'écrit `%pi`, voir le livret de présentation de « scilab ».

```

Console Scilab
Initialisation :
  Chargement de l'environnement de travail

Chargement du module lycée version 1.3 (octobre 2010)
Ce module change le comportement par défaut de Scilab. Faire : "help lycée"

-->a=%pi
a =

    3.1415926535898

-->viventlesmaths=%e
viventlesmaths =

    2.718281828459

-->ans=%e*%pi
ans =

    8.5397342226736

-->x=ans;

-->y=[a,viventlesmaths,x]
y =

    3.1415926535898    2.718281828459    8.5397342226736

-->format('v',25);y
y =

    column 1 to 2
    3.1415926535897931159980    2.7182818284590450907956
    column 3
    8.5397342226735659664882

-->format('e',10);y
y =

    3.142D+00    2.718D+00    8.540D+00

-->

```

Figure 1

Commentaires :

- ✓ Lorsque l'on risque d'avoir à faire des corrections ou des modifications (c'est presque toujours le cas), il vaut mieux *éviter d'utiliser directement la console* de « scilab » au bénéfice de son *éditeur de texte* appelé « SciNotes » (cliquer sur le logo situé à l'extrême gauche de la barre de menus de la console, figure 1). Quand les commandes sont tapées dans « SciNotes »

```

A_jeter.sce (/Users/raymondmoche/Desktop/48_Calculs scilab/A_jeter.sce) - SciNotes
27_1_Ajeter.sce X A_jeter.sce X
1 clear;
2 x=sin(3)*%e/(1+tan(%pi/8));
3 format('v',25);x;
4 afficher(x)

```

Figure 2

et qu'on veut les exécuter, il suffit de les enregistrer (4ème logo de la barre de menus de « Sci-

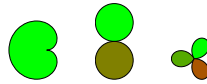
Notes », figure 2) et de donner un nom à l'algorithme, puis de l'exécuter « Exécuter>Fichier sans écho » de la barre des menus de la console). Pour les exécutions ultérieures du même algorithme, on peut utiliser « Exécuter>Enregistrer et lancer ». Il est nécessaire, quand on utilise « SciNotes », de demander l'affichage du résultat ;

- ✓ il faut savoir comment s'appellent les fonctions classiques enregistrées dans la bibliothèque de fonctions de « scilab » : *sin* désigne ici la fonction *sinus*, les angles étant exprimés en radians ; *tan* désigne la fonction *tangente* ; voir les références et notamment l'aide en ligne. Par exemple, l'aide pour « format » s'obtient par la commande :

```
-->help format
```

```
-->
```

- ✓ Il ne faut pas confondre précision de l'affichage et précision des calculs ;
- ✓ L'invite à la dernière ligne de la figure 2 montre que la console est prête pour de nouvelles commandes.



Énoncé 2 [*] Lignes trigonométriques remarquables

- 1 - Saisir le tableau « Arcs » des angles remarquables entre 0 et π .
- 2 - Calculer et afficher les sinus, cosinus et tangentes de ces angles.

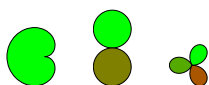
Mots-clefs : saisie d'un tableau à la main, « sin », « cos », « tan ».

Listing 1 – Lignes trigonométriques remarquables

```
// Sinus, cosinus et tangentes remarquables
Arcs=[0,%pi/6,%pi/4,%pi/3,%pi/2,2*%pi/3,3*%pi/4,5*%pi/6,%pi];
Sin=sin(Arcs);
Cos=cos(Arcs);
Tan=tan(Arcs);
afficher('Les valeurs remarquables du sinus entre 0 et pi sont')
afficher(Sin);
afficher('Les valeurs remarquables du cosinus entre 0 et pi sont')
afficher(Cos);
afficher('Les valeurs remarquables de la tangente entre 0 et pi sont')
afficher(Tan)
```

Commentaires sur les résultats (image de la console ci-après)

- ✓ Le tableau « Arcs » est ici un tableau à une ligne et 9 colonnes.
- ✓ Ce tableau a été saisi à la main. Ses éléments sont séparés par des « , ». Si on avait voulu l'écrire en colonne, on aurait utilisé des « ; ». « ; » signifie un changement de ligne.
- ✓ On aurait aimé trouver $\cos \frac{\pi}{3} = \frac{\sqrt{3}}{2}$. Sauf erreur, « scilab » ne sait pas faire cela. Sauf erreur, c'est du domaine des logiciels de calcul formel comme « Xcas ».
- ✓ On remarque aussi que « scilab » ne donne pas la valeur 0 à $\sin 0$ et à $\cos \frac{\pi}{2}$ et donne une valeur à $\tan \frac{\pi}{2}$.
- ✓ « scilab » ne donne que des résultats approchés. La problème de la précision des calculs est un réel problème pour l'utilisateur ordinaire qui ne sait pas comment « scilab » fait ses calculs et qui considère que c'est une boîte noire : on ne sait pas ce qui se passe dedans.
- ✓ Enfin, pour calculer les sinus des éléments d'un tableau « A », on voit qu'il suffit de la commande « sin(A) ». « scilab » retourne le tableau de mêmes dimensions dont les éléments sont les sinus des éléments correspondants de « A ».
- ✓ Cette faculté de calculer les valeurs d'une fonction de la bibliothèque « scilab » par tableau, composante par composante, est extrêmement pratique.



Énoncé 3 ^[**] Conversion de la base 2 à la base 10

Un nombre entier $n \geq 0$ est donné sous la forme $N = [a_0, a_1, \dots, a_p]$ dans le système à base 2. Nous conviendrons que cela signifie que a_0, \dots, a_p étant chacun égaux soit à 0, soit à 1 et p désignant un entier ≥ 0 ,

$$n = a_0 + a_1 \cdot 2 + \dots + a_p \cdot 2^p$$

Problème : écrire n dans le système décimal.

Mots-clefs : commandes « input », « size », « ones », « sum », opérations élément par élément, fonction d'un tableau.

L'algorithme qui suit est un peu violent. Il montre comment « scilab » traite les tableaux de nombres et pourquoi il permet d'écrire des algorithmes très courts.

```
N=input('N=');
n=sum(N.*2.^(0:size(N,'c')-1));
afficher(n);
```

Voici le même algorithme un peu détaillé et commenté.

```
N=input('N=');// Saisir N (liste des coefficients dyadiques).
P=0:(size(N,'c')-1);// P=[0,1,...,p].
Q=2*ones(P);// Q=[2^0,2^1,...,2^p]
R=N.*Q.^P;//R=[a0*2^0,a1*2^1,...,ap*2^p]
n=sum(R);// n=a0*2^0+a1*2^1+...+ap*2^p
afficher(P);
afficher(Q);
afficher(R);
afficher(n);
```

Cet algorithme a été tapé ou ouvert dans l'éditeur de texte « SciNotes » de « scilab ». On l'exécute dans la console (« Exécuter>Fichier sans écho »). « scilab » demande de saisir N (input). Il y a $p+1$ coefficients dyadiques, donc le nombre de colonnes de N est $p+1$ (`size(N,'c')` est le nombre de colonnes de N). P est la liste des puissances de 2 qui interviennent (on va de 0 à p avec le pas 1 qui est la valeur du pas par défaut sinon, on précise). `ones(P)` est le tableau de même dimension que P qui ne contient que des 1. Tous ces éléments sont multipliés par 2 pour obtenir Q . À la ligne suivante, il y a une fonction dont l'argument est le tableau P (on applique la fonction $x \mapsto 2^x$ à tous les éléments de P et on obtient une liste de même longueur) puis il y a une opération sur des tableaux (ici des listes) élément par élément : on multiplie chaque coefficient dyadique par la puissance de 2 correspondante. Ces produits constituent la ligne R . Enfin, on somme R pour calculer n .

Cela peut être visualisé sur la capture d'écran qui suit : c'est le cas où $N = [1, 1, 1, 1, 1, 1, 1]$. On obtient (heureusement $n = 2^7 - 1$).

$N = [1, 1, 1, 1, 1, 1, 1, 1]$

0. 1. 2. 3. 4. 5. 6.

2. 2. 2. 2. 2. 2. 2.

1. 2. 4. 8. 16. 32. 64.

127.

