TP1 : prise en main des logiciels `Maple` et `MATLAB`

Cette première séance de TP (deux séances d'1h20) a essentiellement pour objectif de vous familiariser avec le logiciel de calcul symbolique `Maple12`. En fin de séance, vous aurez un premier contact avec le logiciel (de calcul scientifique cette fois) `MATLAB`, ce qui vous permettra au moins juste d'appréhender la signification de l'acronyme : `MATrixLABoratory`.

Exercice 0.1. Pour commencer le travail :

(1) Connectez vous sur une des machines, *via* votre `login` et votre mot de passe, exactement comme vous vous connecteriez à votre ENT (par exemple pour aller sur le serveur `Ulysse`).
(2) Une fois que vous avez accès à votre répertoire, mettez vous dans `Documents` et créez là un dossier `N1MA3003`. Placez vous ensuite dans ce nouveau dossier pour y créer deux sous-dossiers `TPMaple12` et `TPMATLAB`.
(3) Utilisez le navigateur web pour vous placer sur le site
`http://www.math.u-bordeaux1.fr/~yger/initiationMAPLE`
et téléchargez (dans le répertoire `TPMaple12` que vous venez de créer) le fichier `premierspasMAPLE.mw` ; n'essayez pas pour l'instant d'ouvrir ce fichier (vous l'ouvrirez ultérieurement sous l'environnement `Maple12`).
(4) Toujours avec le navigateur web, placez vous sur le site
`http://www.math.u-bordeaux1.fr/~yger/initiationMATLAB`
et téléchargez tous les fichiers (`.m`) y figurant dans votre dossier `TPMATLAB` ; n'essayez pas pour l'instant d'ouvrir ces fichiers (vous les ouvrirez ultérieurement sous l'environnement `MATLAB`).

Exercice 0.2 (préparation de l'environnement `Maple12`). En utilisant votre fenêtre de terminal, lancez `Maple12` en tapant la commande `xmaple12`. Une fois le logiciel ouvert, deux opérations préliminaires sont à effectuer pour préparer votre environnement une fois pour toutes.

– Vous allez être amenés à travailler sous `Maple` toujours en mode *Worksheet*, donnant accès, contrairement au mode *Document*, à toute la puissance du logiciel. Pour que `Maple` s'ouvre tout le temps dans votre environnement sous ce mode *Worksheet*, suivez l'arbre :

$$Tools \to Options \to Interface \to Default\ format... \to Worksheet$$

Cliquez sur `Apply globally` pour valider ces instructions.

– Il est de loin préférable (pour limiter les erreurs) de travailler en mode `Text [C-Maple Input]` plutôt qu'en mode `Math`. Pour faire en sorte que ceci soit automatique dès que vous ouvrez le logiciel, suivez l'arbre :

$$Tools \to Options \to Display \to Input\ display \to Maple\ notation$$

Gardez *2-D Math notation* en revanche pour l'affichage en sortie (*Output display*). Validez encore par `Apply globally`.

Les instructions que vous allez effectuer après le prompt seront maintenant (comme le prompt) en rouge. Vérifiez le. Ouvrez (en vous mettant sous l'onglet `file`) le fichier `premierspasMAPLE.mw` que vous avez mis dans votre répertoire `TPMaple12`. Ce fichier a été rédigé sous `Maple10`, mais vous devez pouvoir l'ouvrir en mode *Worksheet* sous `Maple12` (ce mode reprenant la syntaxe des versions antérieures).

Dans une des instructions (1) à (4), remplacer le [`;`] final par [`:`] . Que constatez vous ? Ligne (2), remplacez 2 par un entier `N` que vous choisirez au hasard. Constat ? Refaites l'opération sur la même ligne en remplaçant `N` par `N.0`. Comment expliquez vous ce qui se passe ? Fermez le fichier `premierspasMAPLE.mw` sans enregistrer les modifications que vous avez été amenés à faire dans cet exercice.

Exercice 0.3 (le rôle de quelques commandes clef).

(1) Ouvrez sous `Maple12` une feuille de travail vierge (onglet *File* $\to$ *New*). Tapez les instructions suivantes et dégager le rôle des commandes `;`, `:`, `%`, `:=` (comparez avec `=`), `?`, `restart` (derrière `#`, on a fait figurer des commentaires) :

```
> P:= 1*2*3*4*5 ;
> S:= 1+2+3+4+5 ;
> P+S ;
> S ;
> A := % ; b := %%% /15 ;
> (P+S)/15 ; #verifier que b=(P+S)/15
> c = 2+3 ;
> c; #Pourquoi n'obtient-t-on pas 5 comme resultat ici ?
> ?%
> p; P;
> restart ;
> p; P #Quel est le role de la commande restart?
```

(2) Tapez les commandes suivantes et expliquez ce ce qui se passe :

```
> u := < 1, 2, 3 > ;
> v := < 4 | 5 | 6 > ;
> whattype (u) ;
> whattype (v) ;
> A := << 1,0,-1> | < 2,3,0 > | < -1,5,3 >> ;
> whattype (A) ;
> A.u ; u.A; A.v; v.A;
```

Déclarez sous `Maple` la matrice

$$A = \begin{pmatrix} 2 & -4 & -1 & -3 \\ 1 & 0 & -5 & 7 \\ -6 & -8 & 0 & 7 \end{pmatrix}$$

(3) Tapez l'instruction

```
> plot (sin(x)/x, x=-12..12);
```

et expliquez ce qui se passe.

Exercice 0.4 (le changement de base dans les systèmes de numération). Ouvrez une nouvelle feuille de travail vierge sous `Maple`. Apprenez à utiliser les onglets `T` et `[>` pour donner un titre (en texte « interte ») à votre feuille de travail (par exemple : `TP1-exo4`). Utilisez (après avoir étudié dans le *Help* le sens de leur fonction) les routines

```
> Digits := m ;
> convert (N,binary) ;
> convert (evalf(r),binary) ;
> convert (f,binary) ;
```

lorsque `N`, `r`, `f` désignent respectivement un entier naturel, un rationnel strictement positif, un nombre réel flottant strictement positif, pour calculer l'exposant et la mantisse (voir le cours, section 1.2.3) des nombres suivants : $N = 78679$, $r = 89765432/456843$, $f = \pi$, lorsque le nombre `Digits` est fixé d'abord égal à 10, puis à 20. Sauvez votre session (en lui donnant comme nom `TP1-exo4`) dans votre répertoire `TPMaple12`. Ce sera un fichier `.mw`.

Exercice 0.5 (la génération de fonctions). Ouvrez une nouvelle feuille de travail vierge sous `Maple`. Donnez lui un titre (par exemple `TP1-exo5`). Une fonction d'une variable réelle (par exemple `g`) se déclare sur le modèle par exemple de la fonction $x \mapsto x^2$:

```
> g := x -> x^2 ;
```

Vérifiez le avec cette fonction, puis déclarez les fonctions $h : x \mapsto \sin x/(1+x^2)$, $h \circ g$ et $h \circ h$ (exploiter pour cela les commandes `@` et `subs`). Tracez les graphes de ces fonctions sur $[-1,1]$, $[-10,10]$, puis sur $\mathbb{R}$. Sauvez votre session (en lui donnant comme nom `TP1-exo5`) dans votre répertoire `TPMaple12`.

Exercice 0.6 (l'utilisation de la palette de gauche « expressions »). Ouvrez une nouvelle feuille de travail vierge sous `MAPLE`. Donnez lui un titre (par exemple `TP1-exo6`).

(1) Calculez la limite en $\pi/4$ de la fonction cosinus, puis une approximation du résultat (avec `Digits:=10`, puis `Digits:=20`).

(2) Calculez la limite en 0 de $x \mapsto (1-\cos x)/x^2$.

(3) Tester ce que répond le logiciel lorsqu'on lui demande de calculer une primitive (intégrale sans bornes précisées) pour

$$x \mapsto \arctan(x),\ x \mapsto \frac{\log x}{1+x},\ x \mapsto e^{-x^2},\ x \mapsto (x^2+1)\log x.$$

Dans quel cas (parmi ces quatre exemples) le résultat est-il réellement concluant et exploitable du point de vue pratique ? Quel est dans ce cas l'outil que le logiciel est parvenu à mettre en œuvre ?

(4) Calculez la somme des entiers de 1 à n et exploiter la commande `factor` pour factoriser cette somme. Evaluer cette somme lorsque $n = 1234$ en utilisant `eval` et `subs`.

(5) Calculez, pour $n \in \mathbb{N}$,

$$\Pi(n) = \frac{\prod_{0<k \text{ pair } \leq 2n} k}{\prod_{0<k \text{ impair } <2n} k},$$

(faites une recherche avec l'aide pour comprendre ce qu'est la fonction Gamma, Γ, que le logiciel introduit ici dans ses réponses) puis évaluez $\Pi(n)/\sqrt{n}$ pour $n = 1000$, $n = 5000$, $n = 10000$. Que semble-t-il se dessiner ? En utilisant toujours la palette de gauche « Expressions » , calculez

$$\lim_{n\to+\infty} \Pi(n)/\sqrt{n}.$$

Sauvez votre session (en lui donnant comme nom `TP1-exo6`) dans votre répertoire `TPMaple12`.

Exercice 0.7 (manipulation de nombres complexes, formule de Machin). Cet exercice est la mise en pratique d'un exemple qui sera exploité en cours (section 1.4.1 du polycopié). Mais il s'agit ici que vous conduisiez vous même les calculs sous `Maple`. Ouvrez une nouvelle feuille de travail vierge sous `Maple`. Donnez lui un titre (par exemple `TP1-exo7`).

(1) Calculez (avec `Maple`) module et argument des nombres complexes $1+i$ et $2+3i$.

(2) Vérifiez (toujours sous `Maple`) la formule de John Machin (mathématicien anglais, 1680-1751) :
$$(5+i)^4 = 2(239+i)(1+i). \qquad (*)$$
Peut-on se fier par contre à `Maple` pour valider la formule
$$\arctan(1) = 4\arctan(1/5) - \arctan(1/239) \qquad (**)$$
ou faut-il raisonner mathématiquement à partir de la formule (*) validée, elle, grâce au logiciel ? Faites le si nécessaire dans ce cas.

(3) On admet ici que la formule (**) se lit aussi :
$$\frac{\pi}{4} = \arctan(1) = \lim_{n\to+\infty} \sum_{k=0}^{n} \frac{(-1)^k}{2k+1}\Big(4\,(1/5)^{2k+1} - (1/239)^{2k+1}\Big) \qquad (***)$$
car
$$\arctan(x) = \int_0^x \frac{dt}{1+t^2} \quad \forall\, x\in\mathbb{R}$$
(en particulier pour $x = 1/5$ et $x = 1/239$) et que
$$\frac{1}{1+t^2} = \lim_{n\to+\infty}\Big(\sum_{k=0}^{n}(-1)^k t^{2k}\Big) \quad \forall\, t\in]-1,1[$$
(en particulier pour tout $t \in [0,1/5] \supset [0,1/239]$). On pose, pour tout $n\in\mathbb{N}$,
$$u_n := 4\sum_{k=0}^{n} \frac{(-1)^k}{2k+1}\Big(4\,(1/5)^{2k+1} - (1/239)^{2k+1}\Big).$$
Montrez que les deux suites $(u_{2n})_{n\geq 1}$ et $(u_{2n-1})_{n\geq 1}$ sont adjacentes.

(4) Calculez, en utilisant la palette de gauche « Expressions » sous `Maple`, les deux fractions u_{199} et u_{200}, puis leurs écritures avec `Digits=500`. Combien de décimales exactes de π peut-on déduire des résultats affichés ? Justifiez clairement votre réponse en vous appuyant sur un raisonnement mathématique (reposant sur la formule $(***)$ partiellement admise).

Sauvez votre session (en lui donnant comme nom `TP1-exo7`) dans votre répertoire `TPMaple12`.

Exercice 0.8 (familiarisation avec les routines `solve` et `fsolve`). Ouvrez une nouvelle feuille de travail vierge sous `Maple`. Donnez lui un titre (par exemple `TP1-exo8`).

(1) Résolvez les équations :

$$x^2 + x + 1 = 0,\ x^3 + x + 1 = 0,\ x^2 + 3x + 1 > 0,\ \sin x = 0.$$

Expliquez pourquoi dans les deux premiers cas, on obtient bien toutes les racines complexes, alors que ce n'est manifestement pas le cas dans le dernier cas.

(2) Résolvez les inéquations (cette fois dans $\mathbb{R}$) :

$$x^2 + 3x + 1 > 0,\ x^3 + x + 1 > 0.$$

(3) Dites que répond le logiciel lorsqu'on lui demande de résoudre

$$x^5 - 2x^4 + 3x^2 - 3/2 = 0, \quad x \in \mathbb{C},$$

soit l'instruction :

```
> solve (x^5 - 2*x^4 + 3*x^2 - 3/2 =0,x) ;
```

Est-ce vraiment une surprise ? Dites ce qui se passe par contre avec la commande :

```
> solve(x^5 - 2*x^4 + 3*x^2 - 1.5 = 0,x);
```

Expliquez pourquoi la réaction du logiciel est dans ce cas différente. Que se passe t-il sur cet exemple si l'on remplace la routine `solve` par `fsolve` ?

Sauvez votre session (en lui donnant comme nom `TP1-exo8`) dans votre répertoire `TPMaple12`.

Exercice 0.9 (un (tout) premier contact avec `MATLAB`). Fermez l'environnement `Maple` et lancez `MATLAB` avec la commande `matlab` depuis la fenêtre de votre Terminal. Une fois `MATLAB` ouvert, placez vous, en utilisant l'onglet dans le bandeau supérieur, dans votre répertoire `TPMATLAB`.

(1) Ouvrez (en utilisant l'onglet `file`) le fichier `test1.m` et analysez-en la syntaxe. Que devrait être le résultat final ?

(2) Lancez `test1` derrière le prompt de `MATLAB`. Qu'observez vous ? Sachant que le logiciel code les réels en double précision (`binary64`), peut-on prédire à partir de quel seuil d'itérations N le résultat de `test1` n'est plus fiable ?

(3) Ouvrez (toujours en utilisant l'onglet `file`) le fichier `PGCD.m`. Essayez d'analyser comment l'algorithme présenté ici traduit la démarche mathématique conduisant au calcul du PGCD (souvenez vous de l'algorithme d'Euclide présenté en MISMI). Quel sous-programme est appelé dans cette procédure ? Testez la commande `PGCD` en le lançant derrière le prompt :

```
>> d=PGCD (a,b);
```

(avec des valeurs des entiers naturels a et $b > 0$ fixées). Testez aussi le sous-programme que cette procédure appelle.

(4) Vérifiez que la matrice

$$A = \begin{pmatrix} 2 & -4 & -1 & -3 \\ 1 & 0 & -5 & 7 \\ -6 & -8 & 0 & 7 \end{pmatrix}$$

se déclare sous `MATLAB` ainsi :

```
>> A = [ 2 - 4  - 1  -3 ; 1  0 - 5  7 ; -6 - 8  0  7] ;
>> A
```

Déclarez, si vous avez compris, une matrice B cette fois à 4 lignes et 3 colonnes (mettez des entrées réelles arbitraires). Que se passe -il lorsque vous tentez les commandes :

```
>> C = A*B;
>> C = B*A;
>> BB = B';
>> C = BB*A;
>> C = A*BB;
>> C = A.*BB;
>> C = BB.*A;
```

Dans chaque cas où vous n'avez pas reçu un message d'erreur, affichez `BB` ou `C` en faisant

```
>> BB
>> C
```

(sans point virgule cette fois) et tentez d'analyser ce qui se passe. Les bases de calcul matriciel acquises en S2 vous seront ici utiles.

Sauvez votre session (elle le sera automatiquement dans le répertoire `TPMATLAB` sous lequel vous travaillez sous forme d'un fichier `.mat`) ainsi (vérifiez bien après que ce fichier a été créé) :

```
>> save TPMATLAB-1
```

Exercice 0.10 (préparation au TP suivant : Euclide étendu, boucles et récursivité).

(1) Révisez dans votre cours de MISMI la démarche conduisant à la recherche d'une solution de l'identité de Bézout : étant donnés deux entiers relatifs a et b avec $b \neq 0$, trouver deux nombres entiers u et v tels que

$$au + bv = \mathrm{PGCD}\,(|a|, |b|).$$

(2) Ouvrez sous `MATLAB` la routine `bezout.m` (en vous mettant dans le répertoire `TPMATLAB`).

(3) Réfléchissez sur le synopsis de la procédure (ici récursive, on notera qu'elle s'auto-appelle) écrite dans ce programme. Essayez de la relier à la démarche mathématique vue en MISMI pour trouver une solution (u, v) à l'identité de Bézout (et en même temps d'ailleurs le PGCD de $|a|$ et $|b|$).

(4) Faites des tests pour valider cette routine sur des exemples.