TP MAPLE des 2 et 24 mai

Bibliographie : Je me suis beaucoup servie des livres suivants, qui pourront vous être utiles pour progresser avec MAPLE :
- *Algèbre, arithmétique et maple*, Bernadette Perrin-Riou, éditions Cassini (**pour la licence, le master 1 et l'agreg**),
- *Algèbre générale et linéaire, Tome 2*, D. Krob et S. Legros, éd. Vuibert (**Pour prépa et étudiants de maths et info**)
- *Maple V en classes prépas*, P. Rambach, ellipses (**pour la prépa, ou débuter en Maple**)

Initiation à MAPLE : L'aide en ligne de Maple est un outil précieux. On l'appelle en tapant simplement `?` puis « ENTRÉE », ou encore `?nom-d'une-commande` puis « ENTRÉE » lorsqu'on cherche de l'aide sur une commande Maple particulière. Le symbole > marque le début d'une ligne de commande. Que font les commandes **CTRL+T, CTRL+ J, CTRL+K** ? Une commande Maple doit toujours finir par un « `;` » (ou bien un « `:` » lorsqu'on ne veut pas que Maple affiche le résultat).

Type d'objets : Maple reconnaît la différence entre des entiers et des réels. Essayez par exemple :
`>1;1.;1e0;whattype(1);whattype(1.0);whattype(x);`
Une variable est désignée par une lettre, ou une suite de lettres et de chiffres. (Attention, certains noms comme Pi, Digits ou I sont déjà utilisés par Maple). Pour affecter une valeur à une variable, on utilise `:=`. Pour libérer une variable de toute affectation, on utilise `x:='x'`. Essayez et commentez :

```
>> x;
>> whattype(x);
>> x=1;
>> x; whattype(x);
>> x:='x';
>> x;whattype(x);
>> whattype(x+y); whattype(x*y);
>> whattype(x^(-1));whattype(x/y);whattype(1/x);
>> whattype(x2+y2-z*y);
```

Fonctions et expressions : Consultez l'aide sur `print`, `->`, `unapply`, `eval`, et effectuez et commentez les commandes suivantes

```
>> restart;
>>f:=x2+ln(x);                          %Ceci définit une expression en MAPLE
>>a:=subs(x=2,f); evalf(a);
>>restart; f(x):=x2+ln(x);f(1);
>>f:=x->x2+ln(x);                       % Ceci définit une fonction en MAPLE
>>f(1);f(2);f;eval(f);
>>expr:=x2+ln(x);
>>g:=unapply(expr,x);
>>g(1);g(2);g;eval(g);
>>f(x);g(x);
```

Comment définit-on une fonction de plusieurs variables? Définissez par exemple la fonction $f : (x, y) \in \mathbb{R}^2 \mapsto \sin(xy)$.

Procédures et boucles : une procédure est une suite d'opérations qu'on fait exécuter à Maple dans un certain ordre. Elle utilise un certain nombre de paramètres ou variables d'entrée, et donne à l'arrivée des variables de sortie. Il s'agit en fait d'une fonction au sens mathématique du terme (à une/des variable(s) de départ, on associe une/des variable(s) de sortie), mais qui ne peut pas se définir par une simple formule. On écrit une procédure sous la forme suivante :

```
>> nom_de_ma_procédure:=proc(param_entrée_1,...,param_entrée_n)
>> (...)instructions;
>> end proc;
```

Lorsqu'on écrit une procédure, on a très souvent besoin de variables intermédiaires, qui ne servent qu'aux calculs, mais qui ne nous intéressent plus une fois la procédure effectuée. Il s'agit de variables dites *locales*. On les introduit à l'aide de la commande `local`. Elles sont vidées de leur valeur une fois la procédure terminée. Par ailleurs, lorsque la procédure a été exécutée, on souhaite qu'elle nous donne le résultat. Ceci se fait grâce à la commande `RETURN`, à l'avant dernière ligne. Une procédure ressemble donc à :

```
>> nom_de_ma_procédure:=proc(param_entrée_1,...,param_entrée_n);
>> local variable_1, ... , variable_k;
>> (...)instructions;
>> RETURN(variable_de_sortie);
>> end proc;
```

Voici quelques instructions dont vous aurez très souvent besoin pour écrire des procédures. D'abord, on a très souvent besoin de donner des instructions conditionnelles, qui ne seront exécutées que si la condition est réalisée. Ceci se fait ainsi :

```
>> if condition then (...) instructions;
>> else (...) autres_instructions;
>> fi;
```

Sur certaines versions de MAPLE, on finit l'instruction par `end if;` au lieu de `fi;` Notez que la ligne commençant par `else` est facultative. On peut vouloir que Maple ne fasse rien si la condition n'est pas réalisée. Très souvent, la condition est un test d'égalité ou d'inégalité, qui s'écrit `if a=b then ...` ou `if a>b then ...` ou `if a>b=0 then ....`

Dès qu'on écrit un programme, même élémentaire, on a besoin de *boucles*, i.e. d'instructions effectuées à la suite de manière répétitive. La première façon de faire est d'utiliser l'instruction `for`. Regardez l'exemple type suivant, et expliquez ce qu'il fait.

```
>> s:=0;
>> for i from 1 to 21 by 2 do s=s+i;
>> od;
```

Sur certaines versions de Maple, il faut conclure par `end do` au lieu de `od`. La deuxième façon de faire utilise `while`.

```
>> s:=0; i:=1;
>> while i<=21  do s:=s+i; print(s); i:=i+2;
>> od:
```

Sur certaines versions de Maple, il faut conclure par `end do` au lieu de `od`. Cette instruction est très pratique lorsqu'on ne peut pas prévoir exactement le nombre d'opérations à effectuer. Mais attention à ne demander qu'une suite **finie** d'opérations!

Exercice 1 Nous allons fabriquer une procédure `>seconddegre:=proc(a,b,c)` qui va associer à trois nombres a, b, c, l'ensemble des solutions réelles de l'équation $ax2 + bx + c = 0$.
a) Écrivez à la main, la liste des opérations à effectuer, dans l'ordre, pour cette résolution.
b) En utilisant l'instruction `if ...then ... fi;`, construire la procédure voulue, qui doit retourner un *ensemble*, éventuellement vide. (Les ensembles se notent en Maple comme en maths, par exemple `ensemble-des-solutions:=`$\{0, 2\}$.)

Exercice 2 Affichez la liste des nombres premiers inférieurs à 1000 en utilisant les commandes `isprime` et `op`.

Entiers, congruences, théorème des restes chinois : Vous pourrez avoir besoin dans ce paragraphe des instructions suivantes : `ifactor, ifactors, igcd, igcdex, iquo, irem, ilcm, mod, chrem, op` N'hésitez pas à consulter l'aide à leur sujet.

Exercice 3 Calculez le reste de la division euclidienne de 52012356 par 125. Faites-le en utilisant d'une part `iquo`, d'autre part `irem`. Trouvez des couples d'entiers (x, y) solutions de $12x + 83y = 5$, puis $12x + 93y = 5$, puis $12x + 93y = 48$. Quelles sont toutes les solutions? Prévoyez la forme du résultat avant de lancer la commande Maple.

Exercice 4 **a)** Factorisez le nombre 106875. En utilisant la commande `op`, faites la liste des exposants des facteurs premiers. Calculez le produit des nombres premiers le divisant (on appelle ce produit le *radical de* 106875.
b) Faites une procédure associant à $n \in \mathbb{Z}$ la liste des exposants de la décomposition en facteurs premiers et son radical.
c) Testez-la sur des exemples.

Exercice 5 À l'aide de la commande `isolve`, trouvez successivement des solutions entières des équations $12x + 93y = 3$, $12x + 83y = 5$, $12x + 93y = 5$, $x^2 - y^2 = 1$, $x^2 + y^2 = 1$, $x^2 - 2y^2 = 1$.

Exercice 6 Le nom de la commande `chrem` vient de l'anglais *chinese remainder theorem*. Consultez l'aide à ce propos. Résolvez les systèmes suivants, i.e. donnez toutes les solutions en entiers des systèmes suivants.

$$\begin{cases} x \equiv 20 \mod 435 \\ x \equiv 5 \mod 568 \end{cases} \quad \begin{cases} x \equiv 20 \mod 435 \\ x \equiv 6 \mod 567 \end{cases} \quad \begin{cases} x \equiv 20 \mod 435 \\ x \equiv 5 \mod 567 \end{cases}$$

Polynômes : Vous pourrez avoir besoin dans ce paragraphe des instructions suivantes : `collect, expand, sort, normal, coeff, Power, Factor, Factors, gcd, Gcd, Quo, Rem, gcdex, Gcdex, irreduc`

Exercice 7 Calculez la somme et le produit de $P = x^5 + 3x^2 + 2x + 1$ et $Q = x^7 + x + 1$, ordonnez les résultats suivant les puissances décroissantes de x, donnez la liste des coefficients de PQ. Élevez le polynôme P à la puissance 3 modulo 7.

Exercice 8 Effectuez puis commentez les commandes suivantes :

```
>> P:=(x^2-1)*(x-2)-(x-1)*(x+1)*(x-2);
>> normal(P);
>> if P=0 then true else false fi;
>> is (P=0);
>> if is(P=0) then true else false fi;
>> if normal(P)=0 then true else false fi;
>> if P=normal(P) then true else false fi;
```

Recommencez en comparant les fractions rationnelles $\dfrac{(x^2+1)(x+1)^2-(x+1)}{(x+1)(x+2)^2}$ et $\dfrac{x^3+x^2+x}{(x+2)^2}$.

Exercice 9 Expliquez les résultats de

```
>> factor(x^3+1) mod 3;
>> Factor(x^3+1) mod 3;
```

Que faut-il utiliser pour factoriser un polynôme dans $\mathbb{Z}/p\mathbb{Z}[x]$ en un produit de facteurs irréductibles? Factorisez $x^5 - 1$ modulo 5. Factorisez $3x^3 + 2x + 1$ mod 5. Comparez les résultats obtenus en utilisant `Factor` et `Factors`.

Exercice 10 Calculez le pgcd de $P = x^4 - 3x^3 - 3x^2 - 3x + 1$ et $x^3 + 1$ dans $\mathbb{Z}[x]$. Calculez le pgcd Q_1 de P et de $x^3 + 1$ dans $\mathbb{Z}/7\mathbb{Z}[X]$ puis le pgcd Q_2 de P et $x^3 - 1$ dans $\mathbb{Z}/7\mathbb{Z}[x]$. Calculez le quotient de P par Q_1Q_2 modulo 7. En déduire la factorisation de P en facteurs irréductibles modulo 7. Vérifiez avec la commande `Factor`.

Exercice 11 Si P est un polynôme, `diff(P,x)` est son polynôme dérivé. Calculez le pgcd de $P = x^3 + x + 1$ et de P', puis le pgcd de P et P' dans $\mathbb{Z}/31\mathbb{Z}$. Conclusion.

Exercice 12 Calculez le pgcd de $P = 3x^2 + 5x + 7$ et $Q = x^2 + 2x + 1$. Donnez des polynômes U et V tels que $UP + VQ = 1$. À l'aide du résultat obtenu, trouvez un entier n et des polynômes U_0 et V_0 à coefficients dans $\mathbb{Z}$ tels que $U_0P + V_0Q = n$ et tel que n, le contenu de U_0 et le contenu de V_0 soient premiers entre eux dans leur ensemble (on a même $\deg(U_0) < \deg(Q)$ et $\deg(V_0) < \deg(P)$)). On pourra regarder les commandes `content` et `primpart`.

Exercice 13 (Si vous avez le temps) Faites une procédure associant à deux polynômes P et Q à coefficients entiers et premiers entre eux un entier $n = n(P, Q)$ tel qu'il existe des polynômes U_0 et V_0 à coefficients entiers tels que $U_0P + V_0Q = n$ et pgcd(cont(U_0),cont(V_0),n))= 1. L'appliquer à $P = x^8 + x^6 - 3x^4 - 3x^3 + 8x^2 + 2x - 5$ et $Q = 3x^5 + x^4 - 4x^2 - 9x + 21$.

Exercice 14 Consultez l'aide sur `resultant` et `Rootof`. Résoudre les systèmes d'équations suivants

$$\left\{\begin{array}{lcl} xy & = & 4 \\ y2 & = & x3-1 \end{array}\right. \qquad \left\{\begin{array}{lcl} x2+2y3 & = & 3 \\ x2+xy+y2 & = & 3 \end{array}\right. \qquad \left\{\begin{array}{lcl} x2+2y2 & = & 2 \\ x2+xy+y3 & = & 2 \end{array}\right.$$

Pour cela on pourra procéder ainsi. On calcule le résultant de $P = xy - 4$ et $Q = y2 - x3 + 1$ vus comme polynômes à coefficients dans $\mathbb{C}[y]$. Ce résultant est donc un polynôme en y. On cherche les racines de ce polynôme, puis on résout complètement en x et en y.

Exercice 15 (Division euclidienne) a) Commencez par effectuer sur papier la division euclidienne de $x3 + x2 + x + 1$ par $x + 1$, en donnant des noms à chacun des objets que vous manipulez, en décortiquant chaque étape (expliquez d'où vient chacun des termes que vous manipulez) et chaque opération (addition, soustraction, division,..)
b) On se propose d'écrire une procédure `div-euclidienne:=proc(A,B)` qui associe à deux polynômes A et B des polynômes Q et R tels que $A = BQ + R$ et $\deg R < \deg B$. L'algorithme est le suivant. Appelons « reste » chacun des polynômes que nous aurons à diviser successivement par B. Au départ, le reste est égal à A, à la fin, il vaudra R. Au départ, Q vaut 0, et on lui ajoutera progressivement des termes. Et on effectue les opérations suivantes :
Tant que $\deg R \geq \deg B$ (utiliser `degree`), on
- on divise le monôme de plus haut degré de R par le monôme de plus haut degré de B. On appelle `Divise` ce coefficient. (On pourra utiliser les instructions `coeff` et `degree`).
- on remplace Q par Q+`Divise` ci-dessus.
- on multiplie B par `Divise`, et on retranche le résultat à R.
- on remplace R par le résultat de la soustraction, qu'on aura développé.
- on recommence.

Lorsque la boucle est terminée, on renvoie Q et R. Écrivez la procédure correspondante, et testez la sur des exemples simples que vous savez faire à la main.