Calcul formel
et
Mathématiques
avec
la HP40G

Renée De Graeve
Maître de Conférence à Grenoble I
Version 2.0

23 janvier 2001

Remerciements

Tout le monde savait que c'etait impossible d'écrire seul, un logiciel de calcul formel performant....Seul, un illuminé, Bernard Parisse ne le savait pas...et il l'a fait!!!
Voici son logiciel de calcul formel (dit ERABLE) implanté pour la deuxième fois sur une calculatrice HP.
Cela a amené Bernard Parisse à modifier quelque peu son logiciel de façon à ce que les fonctions de calcul formel puissent être éditées et avoir leurs réponses dans l'éditeur d'équations....
A vous de découvrir toutes les performances de cette calculatrice, au fil des pages de ce livre.

Je remercie :
- Bernard Parisse pour ses précieux conseils, ses remarques sur ce texte, sa relecture, et pour sa faculté d'écrire des fonctions à la demande, avec efficacité et gentillesse,
- Jean Tavenas pour l'intérêt porté à l'achèvement de ce guide,
- Jean Yves Avenard pour avoir pris en compte nos suppliques et pour avoir, grâce à son esprit prompt, écrit la commande PROMPT de façon impromptue... (cf 6.4.2).

© 2000 Hewlett-Packard, `http ://www.hp.com/calculators`
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU Free Documentation License (chapter 8, p. 145)".

Préface

La HP40G va marquer une nouvelle étape dans la démocratisation de l'utilisation du calcul formel d'une part, par son prix très compétitif, et d'autre part, par le nombre de possibilités d'exécuter en pas-à-pas les principaux algorithmes enseignés en mathématiques au lycée et dans les premières années à l'Université.

Mais encore fallait-il lui adjoindre une documentation adéquate, de préférence écrite par un enseignant de mathématiques. C'est ce que vous trouverez dans ce guide réalisé par Renée De Graeve, maitre de conférences à l'Université de Grenoble I et animatrice à l'IREM de Grenoble. Il contient, bien sûr, une référence complète des fonctions de calcul formel, mais montre aussi sur des exemples tirés du baccalauréat et du brevet comment tirer parti intelligemment de la puissance de calcul de la HP40G et se termine par deux chapitres consacrés à la programmation : le premier pour apprendre à programmer et le second qui illustre l'algorithmique appliquée au programme d'arithmétique de spécialité des Terminales Scientifiques.

Bernard Parisse

Maître de Conférences à l'Université de Grenoble I

End-User Terms and Conditions :
Use of the CAS Software requires from the user an appropriate mathematical knowledge. There is no warranty for the CAS Software, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holder provides the CAS Software As Is without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the CAS Software is with you. Should the CAS Software prove defective, you assume the cost of all necessary servicing, repair or correction.

In no event unless required by applicable law will any copyright holder be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the CAS Software (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the CAS Software to operate with any other programs), even if such holder or other party has been advised of the possibility of such damages. If required by applicable law the maximum amount payable for damages by the copyright holder shall not exceed the royalty amount paid by Hewlett-Packard to the copyright holder for the CAS Software.

Pour commencer

0.1 Présentation générale

0.1.1 Mise en route

Appuyer sur la touche `ON`.
Vous êtes dans l'écran `HOME`.
En cours de travail, cette touche `ON` annule l'opération en cours : elle joue le rôle de `CANCEL`.
Pour éteindre la calculatrice, taper `SHIFT` puis sur `ON (OFF)`.
Si malgré plusieurs `ON` (`CANCEL`), la calculatrice ne repond pas, appuyer simultanément sur `ON` et `F3` pour la réinitialiser.

0.1.2 Que voit-on ?

De haut en bas :
1. l'écran de `HOME`
1.a l'état de la calculatrice
1.b un trait horizontal
1.c un bandeau contenant des commandes
2. le clavier

1.L'écran :
1.a L'état de la calculatrice décrit les modes mis en œuvre dans l'écran `HOME` :

- `RAD` ou `DEG` ou `GRD` selon que l'on travaille en radians ou en degrés ou en grades.
- `{FUNCTION}` pour indiquer le nom de l'`Aplet` sélectionnée ici : l'`Aplet Function`.

- ▲ pour indiquer que la flèche vers le haut vous permet de remonter dans l'historique.

1.b Un trait horizontal :
- au dessus de ce trait c'est l'historique des calculs faits dans l'écran `HOME`.

Principe : sur l'écran, le calcul demandé s'inscrit à gauche et le résultat s'inscrit à droite.
- en dessous de ce trait c'est la ligne d'édition des commandes.

On peut, grâce à la flèche vers le haut, remonter dans l'historique et recopier, avec `COPY` du bandeau, une commande ou un résultat précédent dans la ligne de commande.

1.c Le bandeau :
Les commandes du bandeau sont accessibles par les 6 touches grises sans nom que l'on nommera ici :
`F1 F2 F3 F4 F5 F6`.
Le bandeau peut contenir des répertoires contenant un ensemble de commandes, ils sont repérables par leur forme de valise.
Pour activer une commande du bandeau, il suffit de taper sur la touche `Fi` correspondante.
Dans l'écran `HOME`, le bandeau possède deux commandes :
- `STO▷` qui permet de mettre une valeur dans une variable et,
- `CAS` qui permet d'ouvrir l'éditeur d'équations pour faire du calcul formel.

2. Le clavier :
Vous avez déjà repéré :
la touche `ON` pour la mise en route ou pour arrêter un calcul en cours et `SHIFT ON` pour éteindre la calculatrice.
Il faut repérer :
- les quatres flèches (gauche, droite, haut, bas) qui permettent de déplacer le curseur lorsqu'on est dans l'éditeur d'équations, dans un menu etc...
- la touche `SHIFT` qui permet à une même touche d'avoir une autre fonction.
- la touche `ALPHA` pour taper du texte en majuscules et les touches `SHIFT` puis `ALPHA` pour taper du texte en minuscules.
  Pour rester en mode de saisie alphabétique il faut maintenir la touche `ALPHA` appuyée.
- `X T` θ permet de taper selon le contexte directement `X`, `T`, θ, `N`.
- la touche `ENTER` sert à valider une commande.

0.2 Notations

Les quatre flèches de direction du curseur sont ici représentées par les quatre triangles :

$$\triangle \ \triangleleft \ \triangleright \ \triangledown$$

Le `STO▷` du bandeau de `HOME` est représenté dans un programme par :

`STO` ▷ ou ▷

Dans l'éditeur d'équations la position du curseur est représentée par :

◀

0.3 L'aide en ligne

Cette calculatrice possède une aide en ligne en français, ou en anglais (cf 4.1.1), très pratique et performante.
On vous propose la liste, par ordre alphabétique, des fonctions de calcul formel. Comme dans chaque menu déroulant, vous pouvez, en appuyant sur une lettre, accéder aux fonctions commencant par cette lettre, sans avoir besoin de taper sur `ALPHA`.
L'aide consiste en une description succinte de la commande, d'un exemple et de sa réponse. Chaque exemple peut être testé avec `ECHO` du bandeau et être traité tel quel, ou modifié. On peut aussi aller voir l'aide des commandes proches grâce aux `SEE1 SEE2...` du bandeau.
Pour plus de détails se référer à la description des touches `SHIFT 2 (SYNTAX)` sections 2.5.4 et 2.7.3.

Chapitre 1

Les Aplets

1.1 La touche APLET

La touche APLET donne accès à la liste des Aplets utilisables.
Cette calculatrice permet en effet de travailler avec des Aplets.
Mais qu'est-ce qu'une Aplet ?
Une Aplet est un logiciel intégré à la machine qui permet facilement d'obtenir 3 vues d'un objet mathématique (une vue symbolique, une vue numérique et une vue graphique) et tout est déjà préprogrammé ! ! !
Les différentes Aplets permettent de travailler avec des objets mathématiques tels que : fonctions, suites, séries statistiques etc...
Certaines Aplets sont des logiciels illustrant des parties de cours.

1.2 Les différentes Aplets

Lorsque vous êtes dans HOME, vous pouvez savoir en regardant la ligne d'état, le nom de l'Aplet sélectionnée.
Voici quelques choix possibles de la touche APLET :

– Sequence
  Cette Aplet permet de définir des suites ayant pour noms :
  U1, U2 ..U9, U0
  On définit U1(N) :
  - soit en fonction de N,
  - soit en fonction de U1(N-1),

- soit en fonction de `U1(N-1)` et de `U1(N-2)`.
On définit par exemple :
`U1(N)=N*N+1`
et alors les valeurs de `U1(1)` et de `U1(2)` sont calculées et mises automatiquement.
En cochant `U1`, puis en appuyant sur `NUM` les valeurs de `U1(N)` s'affichent.
On trouvera d'autres exemples utilisant l'`Aplet Sequence` au paragraphe suivant comme le calcul du $PGCD$ de deux nombres (cf 1.3) et le calcul des coefficients de l'identité de Bézout (cf 1.3).

– `Function`
Cette `Aplet` permet de définir des fonctions ayant pour noms :
`F1(X), F2(X) ..F9(X), F0(X)`
On définit `F1(X)` :
- soit par une expression fonction de `X` :
Par exemple, la formule :
`F1(X)=X*LN(X)`
définit la fonction :
$f_1(x) = x.ln(x)$
- soit, si la fonction est définie par morceaux en utilisant les booléens :
`X>0` etc...
Par exemple, une formule de la forme :
`F1(X)=X*(X<0)+2*X*(X>0)`
définit la fonction :
$f_1(x) = \ x$ si $x < 0$ et
$f_1(x) = 2.x$ si $x > 0$
– `Parametric` pour tracer des courbes en coordonées paramètriques.
– `Polar` pour tracer des courbes en coordonées polaires.
– `Solve` pour résoudre des équations numériques.
– `Statistics` pour faire des statistiques.
– `Inference` pour faire des statistiques inférentielles.

1.3 Exemples utilisant l'`Aplet Sequence`

Écriture en base b

Étant donnés a et b, on veut obtenir, la suite q_n $(n \geq 1)$ et r_n $(n \geq 2)$ des quotients et des restes de la division par b des q_i définies par :
$q_1 = a$
$q_1 = b.q_2 + r_2$ $(0 \leq r_2 < b)$
$q_2 = b.q_3 + r_3$ $(0 \leq r_3 < b)$
......
$q_{n-1} = b.q_n + r_n$ $(0 \leq r_n < b)$
On remarquera que si $r_{n+1} = 0$, le nombre $r_n r_{n-1}.....r_3 r_2$ est l'écriture en base b de a, lorsqu'on suppose $2 \leq b \leq 10$.
On met dans `B` la valeur de la base par exemple :
`7 STO▷ B`
et dans `A` le nombre à écrire en base `B` (par exemple `1789 STO▷ A`)
On définit ensuite deux suites :

```
U1(1)=A
U1(2)=FLOOR(A/B)
U1(N)=FLOOR(U1(N-1)/B)
```

puis

```
U2(1)=0
U2(2)=A MOD B
U2(N)=U1(N-1) MOD B
```

Ainsi q_n=`U1(N)` et r_n=`U2(N)`
On trouve :
`U2(2)=4 U2(3)=3 U2(4)=1 U2(5)=5 U2(6)=0` donc l'écriture en base 7 de 1789 est : 5134.

Le calcul de $PGCD$

Voici une mise en en œuvre de l'algorithme d'Euclide avec la `HP40G`.
Voici la description de cet algorithme :
On effectue des divisions euclidiennes successives :

$$\begin{array}{rll} A = & B \times Q_1 + R_1 & 0 \leq R_1 < B \\ B = & R_1 \times Q_2 + R_2 & 0 \leq R_2 < R_1 \\ R_1 = & R_2 \times Q_3 + R_3 & 0 \leq R_3 < R_2 \\ .... & & \end{array}$$

Après un nombre fini d'étapes (au plus B), il existe un entier n tel que : $R_n = 0$.
on a alors :
$PGCD(A, B) = PGCD(B, R_1) = ...$
$PGCD(R_{n-1}, R_n) = PGCD(R_{n-1}, 0) = R_{n-1}$
À l'aide des suites, on écrit la suite des restes.
Avec la HP40G, on utilise l'`Aplet Sequence` (touche `APLET` puis on sélectionne `Sequence` puis `START` du bandeau).
Si l'on veut déterminer le PGCD(78,56), on définit la suite :

`U1(1)` $=$ `78`

`U1(2)` $=$ `56`

`U1(N)` $=$ `U1(N` $-$ `2) MOD U1(N` $-$ `1)`

On tape sur `NUM` pour avoir la liste numérique des `U1(N)` c'est à dire la liste des restes des divisions successives...

Le dernier reste non nul est `2` donc le PGCD(78,56)=2.

REMARQUE
On peut utiliser dans `HOME` les variables `A` et `B` pour stocker les deux nombres et mettre alors `U1(1)=A U1(2)=B`.
Il faut aussi remarquer que `A MOD 0 = A`.

Le calcul des coefficients de l'identité de Bézout

L'algorithme d'Euclide permet de trouver un couple U, V vérifiant :

$A \times U + B \times V = PGCD(A, B)$

Avec les suites :
On va définir "la suites des restes" R_n et deux suites U_n et V_n, de façon qu'à chaque étape on ait :
$R_n = U_n \times A + V_n \times B$.

Puisque on a : $R_n = R_{n-2} - Q_n \times R_{n-1}$, U_n et V_n vont vérifier la même relation de recurrence (Q_n =quotient entier de R_{n-2} par R_{n-1}).

On a au début :

$R_1 = A$ $R_2 = B$

$U_1 = 1$ $U_2 = 0$ puisque $A = 1 \times A + 0 \times B$

$V_1 = 0$ $V_2 = 1$ puisque $B = 0 \times A + 1 \times B$

Avec la HP40G, grâce à l'Aplet Sequence, on va définir la suite `U1` des restes et les suites `U2` et `U3` qui seront telles que pour tout `N` on ait : `U1(N)=A*U2(N)+B*U3(N)`.

Pour cela on a besoin de la suite des quotients que l'on mettra en U4.

Les suites U1, U2, U3 vérifient la même relation de récurrence :
$U_n = U_{n-2} - Q_n \times U_{n-1}$ avec
$Q_n = \texttt{U4(N)} = \texttt{FLOOR(U1(N}-\texttt{2)/U1(N}-\texttt{1))}$
On définit donc :
U1(1) = A
U1(2) = B
U1(N) = U1(N − 2) − U4(N) ∗ U1(N − 1)
U2(1) = 1
U2(2) = 0
U2(N) = U2(N − 2) − U4(N) ∗ U2(N − 1)
U3(1) = 0
U3(2) = 1
U3(N) = U3(N − 2) − U4(N) ∗ U3(N − 1)
U4(1) = 0
U4(2) = 0
U4(N) = FLOOR(U1(N − 2)/U1(N − 1))

Il faut remarquer que l'on n'utilise U4(N) que pour N > 2, on a donc défini les deux premières valeurs (qui sont inutiles !) par zéro.
NUM va alors afficher les valeurs de ces différentes suites et sur la ligne du dernier reste non nul on pourra lire le pgcd et les coefficients de l'identité de Bézout.

1.4 Les touches SYMB NUM PLOT

Une Aplet est visible en général de trois façons diffèrentes :
- une vue symbolique qui correspond à la touche SYMB
- une vue numérique qui correspond à la touche NUM
- une vue graphique qui correspond à la touche PLOT
Quand ces touches sont shiftées (SETUP), cela correspond au choix des différents paramètres utilisés (choix de l'unité d'angle, des paramètres de la fenêtre graphique etc...).

Chapitre 2

Le Clavier et le CAS

2.1 Qu'est ce que le CAS ?

Le `CAS` permet de faire du calcul formel ou symbolique :
`CAS = Computer Algebra System`.
Il faut bien voir la différence entre :
- calcul formel ou symbolique, c'est celui que l'on fait avec les fonctions du `CAS`. On travaille alors en `mode exact`, en précision infinie et on a la possibilité de faire les calculs en pas à pas,
- calcul numérique, c'est celui que l'on fait avec les fonctions du répertoire `MTH` de la touche `MATH`, dans l'écran `HOME` ou depuis les `Aplets` ou en programmation. On travaille alors en `mode approximatif`, avec une précision de 10^{-12}.
Exemple :
Si on est en `Radians` dans `HOME` :
`ARG(1+i)` vaut 0.785398163397
alors que dans le `CAS` où on est toujours en `Radians` :
`ARG(1+i)` vaut $\frac{\pi}{4}$

2.2 La variable courante

Lorsqu'on utilise des fonctions de calcul formel, on travaille avec des variables symboliques (variables ne contenant aucune valeur).
Le nom de la variable symbolique contenu dans `VX` s'appelle la va-

riable courante : c'est le plus souvent `X`.
L'action de certaines fonctions dépend de la variable courante, par exemple la fonction `DERVX` effectue une dérivation par rapport à la variable courante.
Ainsi,
$\texttt{DERVX}(2 * \texttt{X} + \texttt{Y}) = 2$ si `VX=X`, et $\texttt{DERVX}(2 * \texttt{X} + \texttt{Y}) = 1$ si `VX=Y`.

2.3 Comment faire du calcul formel ?

La `HP40G` a été conçue pour utiliser les fonctions de calcul formel depuis l'éditeur d'équations.
Pour ouvrir l'éditeur d'équations appuyer sur `CAS` du bandeau de l'écran `HOME`.
Pour sortir de l'éditeur d'équations appuyer sur `ON`, on revient ainsi à l'écran `HOME`.
On peut néammoins faire du calcul formel depuis l'écran `HOME` moyennant quelques précautions (cf 2.6).
On se reportera aux chapitres suivants pour savoir utiliser les fonctions du `CAS`.

2.4 Le CAS depuis l'`éditeur d'équations`

L' éditeur d'équations va vous permettre d'écrire comme sur le papier les expressions que vous voulez simplifier, factoriser, dériver, intégrer etc...
C'est un éditeur muni d'un bandeau contenant des répertoires :

1. Le répertoire `TOOL` contient les commandes :
   `Cursor mode`
   `Edit expr.`
   `Change font`
   `Cut`
   `Copy`
   `Paste`

   - `Cursor mode` permet de passer en mode curseur (cf 3.1.4).
   - `Edit expr.` permet d'éditer l'expression mise en surbrillance, ce qui permet de la modifier.
   - `Change font` permet de choisir d'écrire avec de gros ou de petits caractères (on peut faire ce choix à tous moments).

- `Cut` recopie la sélection dans le buffer et l'efface.
- `Copy` recopie la sélection dans le buffer.
- `Paste` recopie la sélection là où se trouve le curseur (il faut avoir fait avant, soit `Copy`, soit `Cut`, pour que la sélection soit dans le buffer).

2. Le répertoire `ALGB` contient des fonctions qui permettent de faire de l'algèbre : factorisation, développement, simplification, substitution...

3. Le répertoire `DIFF&INT` contient des fonctions qui permettent de faire du calcul différentiel : dérivation, intégration, développement limité...

4. Le répertoire `REWRITE` contient des fonctions qui permettent de réécrire une expression sous une autre forme.

5. Le répertoire `TRIG` contient des fonctions qui permettent de transformer des expressions triogonomètriques.

6. Le répertoire `SOLVE` contient des fonctions qui permettent de résoudre des équations, des systèmes linéaires et des équations différentielles.

Vous trouverez dans le chapitre 3, comment écrire une expression dans l'éditeur d'équations, comment sélectionner une sous-expression et comment appeler les fonctions du `CAS`.
Vous trouverez dans le chapitre 4, toutes les fonctions de calcul formel contenues dans ces différents répertoires avec un exemple d'utilisation.
Vous pouvez consulter l'aide en ligne avec `SHIFT 2 (SYNTAX)` (cf 2.5.4), pour avoir l'aide sur les autres fonctions disponibles, et utiliser `SHIFT MATH (CMDS)` (cf 2.5.2) pour les taper.

2.5 Le clavier depuis l'`éditeur d'équations`

Les touches, commentées dans ce paragraphe, n'ont pas la même fonction selon qu'on les utilise depuis l'`éditeur d'équations` ou depuis l'écran `HOME`. Pour la fonctionnalité de ces touches, en dehors de l'`éditeur d'équations`, on se reportera à la section 2.7 ou (et) on consultera le manuel général.

2.5.1 La touche MATH

La touche MATH, pressée depuis l'éditeur d'équations, affiche les fonctions utiles en calcul formel. Ces fonctions sont contenues dans les répertoires :

- les cinq répertoires précédents (cf 2.4) :
  ALGEBRA DIFF&INT REWRITE TRIG SOLVE
- le répertoire Complex... contenant des fonctions qui permettent de travailler avec des complexes.
- le répertoire Constant... (e i ∞ pi)
- le répertoire Integer... contenant des fonctions qui permettent de faire de l'arithmétique entière.
- le répertoire Hyperb.... contenant les fonctions hyperboliques.
- le répertoire Modular... contenant des fonctions qui permettent de faire des calculs dans Z/pZ ou dans $Z/pZ[X]$, p étant la valeur contenue dans la variable MODULO.
- le répertoire Polynom... contenant des fonctions qui permettent de faire des calculs avec des polynômes.
- le répertoire Tests... contenant :
  ASSUME UNASSUME (pour faire des hypothèses sur les paramètres et modifier ainsi la variable REALASSUME cf 3.3.3)
  $> \geq < \leq == \neq$ AND OR NOT
  IFTE (pour écrire une fonction algébrique ayant le même résultat qu'un IF THEN ELSE) .

On se reportera à la section 4.1.8, pour avoir la liste des fonctions se trouvant dans les différents répertoires.

2.5.2 Les touches SHIFT MATH (CMDS)

La combinaison de ces touches ouvre le catalogue de toutes les fonctions de calcul formel utilisables depuis l'éditeur d'équations.
Ainsi les fonctions, qui ne sont pas présentes ailleurs, pourront être appelées depuis ce menu, ce qui vous évite de les taper en mode Alpha.

2.5.3 La touche VARS

Cette touche pressée lorsqu'on est dans l'éditeur d'équations fait apparaître les noms des variables définies dans le CAS.
On remarquera namVX qui contient le nom de la variable courante.
Pour voir le contenu d'une variable il suffit de mettre son nom en

surbrillance et d'appuyer sur F2 pour VIEW du bandeau.
Pour modifier le contenu d'une variable il suffit de mettre son nom en surbrillance et d'appuyer sur F3 pour EDIT du bandeau.
On remarquera aussi dans le bandeau :
PURGE qui permet de détruire une variable existante.
RENAME qui permet de changer le nom d'une variable existante.
NEW qui permet de définir une nouvelle variable : il suffit d'entrer le contenu (object), puis son nom (name).
Pour plus de détails, on se reportera à la section 3.3.

2.5.4 Les touches SHIFT 2 (SYNTAX)

Lorsque que l'on est dans l'éditeur d'équations la combinaison des touches : SHIFT 2 (SYNTAX) ouvre le menu CAS HELP ON.
Pour avoir l'aide en français, choisir Français dans le menu du répertoire CFG permettant de changer votre configuration (cf 4.1.1).
Si dans l'éditeur il n'y a pas de fonction du CAS séléctionnée, ce menu propose la liste des fonctions utilisables depuis l'éditeur d'équations.
Il suffit alors de mettre en surbrillance une fonction et de taper OK pour avoir de l'aide sur cette fonction.
Si dans l'éditeur il y a une fonction du CAS séléctionnée, par exemple : FACTOR(45), le menu CAS HELP ON ouvre directement l'aide à la page de FACTOR. L'aide consiste en une description succinte de la commande, d'un exemple et de sa réponse. Chaque exemple peut être mis dans l'éditeur d'équations avec ECHO du bandeau et être traité tel quel, ou modifié.
Il faut noter que dans les exemples de l'aide, on a choisi comme variable courante VX=X. Si ce n'est pas le cas, l'exemple sera automatiquement transformé, en tenant compte de votre VX, lors du transfert par ECHO.
Vous avez aussi la possibilité d'aller directement voir l'aide d'une commande signalée dans See : avec SEE1, SEE2... du bandeau.

2.5.5 La touche HOME

La touche HOME pressée, depuis l'éditeur d'équations, permet un accès à l'historique du CAS.
L'historique des calculs faits dans le CAS et l'historique des calculs faits dans HOME sont distincts.
Comme dans l'historique de l'écran HOME, les calculs demandés sont

inscrits à gauche et les résultats sont inscrits à droite. On peut grâce à la flèche vers le haut remonter dans l'historique.
Vous pouvez grâce à `ENTER` ou `ECHO` du bandeau, recopier un résultat précédent ou une commande déjà effectuée.

2.5.6 Les touches `SHIFT SYMB`

Lorsque que l'on est dans l'éditeur d'équations la combinaison des touches :
`SHIFT SYMB (SETUP)` est l'analogue de `CFG` (le premier choix des menus `ALGB etc...` du bandeau cf 4.1.1).
Cela vous permet de préciser :
- le nom de la variable contenue dans `VX`, en tapant son nom devant `Indep var`,
- la valeur de `MODULO`, en tapant sa valeur devant `Modulo`,
- si vous voulez travailler en `mode exact` (ou en `mode approximatif` si vous cochez `Approx` avec `CHK` du bandeau),
- si vous voulez travailler en `mode réel` (ou en `mode complexe` si vous cochez `Complex` avec `CHK` du bandeau),
- si vous voulez travailler en mode `Direct` (ou en mode `Step/Step` si vous cochez `Step/Step` avec `CHK` du bandeau),
- si vos polynômes sont écrits selon les puissances décroissantes (ou croissantes si vous cochez `Incr Pow` avec `CHK` du bandeau),
- si vous interdisez des facteurs numériques (ou autorisez des facteurs numériques si vous cochez `Num.Factor` avec `CHK` du bandeau).
- si vous voulez travailler en `mode non rigoureux` (ou en `mode rigoureux` si vous cochez `Rigourous` avec `CHK` du bandeau, pour ne pas négliger les valeurs absolues!),
On valide avec `OK` ou `ENTER`.

2.5.7 La touche `SHIFT ,`

Lorsque que l'on est dans l'éditeur d'équations les touches :
`SHIFT , (MEMORY)` jouent le rôle de "undo".
Cela est très utile quand on s'est trompé, car cela permet d'annuler la dernière commande.

2.5.8 La touche `PLOT`

Lorsqu'on appuie sur `PLOT` depuis l'éditeur d'équations, une boite de dialogues vous demande si vous voulez tracer une fonction, une

courbe en paramètrique ou une courbe en polaire.
Selon ce que vous sélectionnez, l'expression mise en surbrillance sera recopiée vers l'`aplet` correspondante, à l'endroit que vous spécifiez comme destination.
ATTENTION : cela suppose que la variable courante est aussi la variable de la fonction à représenter, car lors de la recopie, l'expression est évaluée et la variable courante (celle contenue dans `VX`) est changée en `X` `T` ou θ, selon la nature du graphique.
ATTENTION : si la fonction dépend d'un paramètre, il est préférable de donner une valeur à ce paramètre avant d'appuyer sur `PLOT`. Si toutefois, vous voulez que l'expression paramètrée soit recopiée avec son paramètre, le nom de ce paramètre doit être composé d'une seule lettre différente de `X` `T` ou θ, pour qu'il n'y ait pas de confusion.
Si l'expression sélectionnée est à valeurs réelles :
l'`Aplet Function` ou l'`Aplet Polar` peut être sélectionnée, le graphe sera alors du type : `Function` ou `Polar`.
Si l'expression sélectionnée est à valeurs complexes :
l'`Aplet Parametric` doit être sélectionnée et le graphe sera du type : `Parametric`.
Si vous choisissez :

- l'`Aplet Function`, l'expression mise en surbrillance sera recopiée dans la fonction `Fi` choisie, et la variable courante sera transformée en `X` lors de la recopie,
- l'`Aplet Parametric`, la partie réelle et la partie imaginaire de l'expression mise en surbrillance seront recopiées dans les fonctions `Xi, Yi` choisies, et la variable courante sera transformée en `T` lors de la recopie,
- l'`Aplet Polar`, l'expression mise en surbrillance sera recopiée dans la fonction `Ri` choisie, et la variable courante sera transformée en θ lors de la recopie.

2.5.9 La touche `NUM`

Lorsqu'on appuie sur `NUM` depuis l'éditeur d'équations l'expression mise en surbrillance est remplacée par une approximation numérique.
`NUM` fait passer en mode approximatif.
`SHIFT NUM` effectue l'opération inverse : on passe en mode exact.

2.5.10 La touche VIEWS

Lorsqu'on appuie sur VIEWS depuis l'éditeur d'équations l'expression mise en surbrillance peut être vue entièrement en faisant bouger le curseur grâce aux flèches $\triangleright$ et $\triangleleft$. Appuyer sur OK du bandeau pour revenir à l'éditeur d'équations.

2.5.11 Les raccourcis avec le clavier

Il faut noter que depuis l'éditeur d'équations on a, avec le clavier, les raccourcis suivants :
SHIFT 0 pour ∞
SHIFT 1 pour i
SHIFT 3 pour π
SHIFT 5 pour $<$
SHIFT 6 pour $>$
SHIFT 8 pour $\leq$
SHIFT 9 pour $\geq$

2.6 Le CAS depuis HOME

On peut utiliser certaines fonctions de calcul formel directement depuis l'écran HOME moyennant quelques précautions :
- utiliser les fonctions de calcul formel que l'on trouve dans CAS du bandeau de la touche MATH (pressée depuis l'écran HOME), la variable courante est alors systématiquement la variable S1, par exemple :
$\texttt{DERVX}(\texttt{S1}^2 - 4 * \texttt{S2}) = 2 * \texttt{S1}$
- utiliser les variables S1,S2,...S5 comme variables symboliques,
- si vous voulez travailler avec des matrices symboliques, il faut les stocker dans L1, L2, ... L9, L0 car ces matrices seront interprétées en tant que listes de listes (alors que les matrices numériques sont stockées dans M1, M2, ... M9, M0). On écrira par exemple :
$[\texttt{S1} + 1, \texttt{XQ}(\sqrt{2})]$ STO $\triangleright$ L1
ATTENTION, certains calculs seront effectués en mode approximatif en raison de l'ambiguité entre réels et entiers dans HOME. L'utilisation de la commande XQ permet de convertir un argument approximatif en argument exact, dans l'exemple précédent vu au paragraphe 2.1, on a depuis l'écran HOME (voir aussi 2.7.1 et 2.7.3) :

$$\texttt{ARG}(\texttt{XQ}(1 + \texttt{i})) = \frac{\pi}{4}$$

Vous pouvez aussi, grâce aux commandes PUSH et POP, transférer des expressions de l'historique de l'écran HOME dans l'historique du CAS.

2.6.1 PUSH

Vous pouvez envoyer, depuis l'écran HOME, des expressions dans l'historique du CAS grâce à la commande PUSH.
On tape depuis l'écran HOME :
`PUSH(S1+1)`
et S1+1 s'inscrit dans l'historique du CAS.

2.6.2 POP

Vous pouvez récupérer, depuis l'écran HOME, la dernière expression écrite dans l'historique du CAS grâce à la commande POP.
On tape depuis l'écran HOME :
`POP`
et par exemple S1+1 s'inscrit dans l'historique de l'écran HOME.

2.7 Le clavier depuis HOME

2.7.1 La touche MATH

Cela ouvre le menu des fonctions mathématiques.
Cette touche pressée depuis l'écran HOME ouvre une fenêtre contenant des fonctions mathématiques (numériques) classées par thèmes, car l'option MTH du bandeau (touche F1) est cochée par défaut.
Si on coche CAS du bandeau de cette fenêtre (touche F3), on trouve les mêmes répertoires que lorsqu'on appuie sur la touche MATH depuis l'éditeur d'équations : on a ainsi accés aux fonctions de calcul formel classées par thèmes et utilisables à partir de l'écran HOME (ne pas oublier que, depuis l'écran HOME, les seules variables symboliques sont S1,S2...S5).

2.7.2 La touche SHIFT F6

La combinaison des touches SHIFT F6 (SHIFT CAS du bandeau) ouvre l'écran de configuration du CAS ce qui permet de changer la configuration du CAS depuis l'écran HOME (cf 2.5.6).

2.7.3 La touche SHIFT 2 (SYNTAX)

La combinaison des touches SHIFT 2 (SYNTAX) place HELPWITH dans la ligne de commande. Il suffit de compléter cette ligne par le nom de la commade ou par le nom de la fonction du CAS pour laquelle vous voulez de l'aide. On peut rentrer le nom d'une fonction du CAS avec MATH CAS, mais il faut prendre garde à enlever la parenthèse.
Par exemple : HELPWITH DERVX vous ouvre l'aide du CAS à la page DERVX.
Si on veut avoir l'aide générale du CAS depuis l'écran HOME il faut taper HELP, puis ENTER : on a ainsi l'aide sur les fonctions du CAS utilisables depuis l'écran HOME.
Pour avoir l'aide en français, choisir Français dans le menu du répertoire CFG permettant de changer votre configuration (on est obligé de le faire depuis l'éditeur d'équations cf 4.1.1).
Chaque exemple peut être mis dans l'historique de l'écran HOME avec ECHO du bandeau et donc être traité tel quel, ou modifié (bien sûr la variable X sera remplacée par S1).
De plus, on sera aussi quelquefois obligé de changer dans HOME les réels en entiers grâce à la fonction XQ.
Par exemple :

$$\mathtt{PROPFRAC}(\frac{43}{12}) = 3.5833..$$

alors que

$$\mathtt{PROPFRAC}(\mathtt{XQ}(\frac{43}{12})) = 3 + \frac{7}{12}$$

2.7.4 La touche SHIFT 1 (PROGRAM)

La combinaison de ces touches pressée lorsqu'on est dans HOME ouvre l'écran
PROGRAM CATALOG.
On voit apparaître :
- la liste des progammes que vous avez écrits,
- un bandeau contenant les commandes :
EDIT NEW RUN SEND RECV.
EDIT permet d'éditer le programme mis en surbrillance,
NEW permet de créer un nouveau programme,
RUN permet d'exécuter le programme mis en surbrillance (cf 6.1).
SEND et RECV sont les fonctions qui permettent de faire dialoguer votre

calculatrice avec votre ordinateur ou une autre calculatrice.
Par exemple :
Si on tape sur SEND du bandeau on vous demande :
HP40G ou Disk drive
vous mettez en surbrillance HP40G pour envoyer un programme vers une autre HP40G ou vous mettez en surbrillance Disk drive pour envoyer un programme vers un ordinateur.
Puis OK du bandeau.
Pour les utilisateurs de Windows, le logiciel de connexion se trouve à l'URL www.hp.com/calculators/france.
Pour les utilisateurs de Linux on utilisera le programme C-Kermit version 7 (que l'on trouve à l'URL www.columbia.edu/kermit ou que l'on peut télécharger par ftp anonyme sur le site kermit.columbia.edu) :
-On branche la calculatrice au cordon de transfert.
-Sur l'ordinateur on tape :
kermit
set line /dev/ttyS0 (ou S1 ...selon votre numéro de port série, attention S0 correspond à COM1 sous MS-DOS)
Il se peut que cette commande provoque l'erreur "access to device denied". Dans ce cas, vous devrez exécuter en tant que root la commande suivante :
chmod 666 /dev/ttyS0 (ou /dev/ttyS1 ... selon votre numéro de port série)
Il se peut aussi que cette commande provoque l'erreur "write access to UUCP lockfile directory denied", dans ce cas vous devrez exécuter en tant que root la commande suivante :
chmod 1777 /var/lock
Puis tapez les commandes suivantes :
set speed 9600
set carrier-watch off
serv
-Sur la HP40G :
on met en surbrillance le programme de nom NOM puis on appuie sur SEND du bandeau et on met en surbrillance Disk drive. Puis OK du bandeau, pour que le programme de nom NOM qui se trouve dans votre HP40G soit recopié sur votre ordinateur.
-Ou
Sur la HP40G :
on appuie sur RECV du bandeau et on met en surbrillance Disk drive.
Puis OK du bandeau : la calculatrice affiche alors la liste des pro-

grammes qui sont sur votre ordinateur (bien sûr il faut avoir créer un répertoire sur votre ordinateur où des programmes de HP40G sont stockés).
On met alors en surbrillance PGCD pour que le programme de nom PGCD qui se trouve sur votre ordinateur soit recopié sur votre HP40G.
Notez qu'on peut automatiser l'exécution des commandes de kermit en les plaçant dans le fichier .kermrc du répertoire racine de l'utilisateur, par exemple créez le fichier suivant :

```
set line /dev/ttyS0
set speed 9600
set carrier-watch off
set file names literal
```

et sauvegardez-le sous le nom ~/.kermrc
Pour en savoir plus sur l'utilisation de Kermit avec les calculatrices HP, vous pouvez consulter l'URL :
http ://www.columbia.edu/kermit/hp48.html

Chapitre 3

Écriture des expressions dans l'éditeur d'équations

3.1 L'éditeur d'équations

3.1.1 Accès à l'éditeur d'équations

La touche `CAS` du bandeau vous permet d'entrer dans l'éditeur d'équations et la touche `ON (CANCEL)` vous permet d'en sortir.
C'est un éditeur très performant pour écrire, simplifier et transformer des expressions mathématiques.

Lorsque l'on est dans l'éditeur d'équations on peut écrire des expressions en sachant que l'opérateur que l'on est en train de taper porte toujours sur l'expression adjacente ou sur l'expression sélectionnée.
On ne se préoccupe pas de mettre des parenthèses, on sélectionne ! ! !
Il faut voir les expressions mathématiques comme un arbre, pas forcément binaire, et comprendre que les quatre flèches permettent de parcourir l'arbre de façon naturelle :
- les flèches droite et gauche permettent d'aller d'un sous-arbre à l'autre,
- les flèches haut et bas de monter ou de descendre dans l'arbre,
- les flèches droite et gauche "shiftées" permettent diverses sélections (cf page 29 l'exemple 3).

3.1.2 Comment sélectionner ?

On peut entrer dans le mode sélection de deux façons

- La flèche $\triangle$ vous fait entrer dans le mode sélection et sélectionne l'élément adjacent au curseur.
  Exemple :
  $$1+2+3+4\ \triangle$$
  sélectionne 4, puis $\triangle$ sélectionne l'arbre tout entier $1+2+3+4$.
- La flèche $\triangleright$ vous fait entrer dans le mode sélection et sélectionne le sous-arbre adjacent au curseur.
  Si vous appuyez à nouveau sur $\triangleright$ vous augmentez votre sélection du sous-arbre contigu, à gauche de votre sélection.
  Exemple :
  $$1+2+3+4\ \triangleright$$
  sélectionne $3+4$, puis $\triangleright$ sélectionne $2+3+4$, puis $\triangleright$ sélectionne $1+2+3+4$
- ATTENTION : si on est en train de taper une fonction ayant plusieurs arguments (comme par exemple une $\sum$ ou une $\int$ ou `SUBST` etc...), la flèche $\triangleright$ permet de progresser dans l'écriture, en changeant le curseur d'emplacement. En effet ce sont les flèches $\triangleright$ et $\triangleleft$ qui permettent le passage d'un argument à l'autre. Il faut donc toujours dans ce cas sélectionner avec la flèche $\triangle$ (cf 3.2.1).

Exemples de fonctionnement de cet éditeur :
On tape sur `CAS` du bandeau pour ouvrir l'éditeur d'équations, puis on entre les expressions des exemples.

- Exemple 1
  On tape :
  $$2\ +\ \mathtt{X}\ \times\ 3\ -\ \mathtt{X}$$
  et on obtient :
  $$2+\mathtt{X}\cdot 3-\mathtt{X}$$
  $\triangleright\ \triangleright\ \triangleright$ pour sélectionner l'expression,
  puis `ENTER` donne le résultat :
  $$2+2\cdot\mathtt{X}$$
  On tape :
  $$2\ +\ \mathtt{X}\ \triangleright\ \times 3\ -\ \mathtt{X}$$

et on obtient :

$$(2+\mathtt{X})\cdot 3-\mathtt{X}$$

$\triangleright$ $\triangleright$ pour sélectionner l'expression,
puis `ENTER` donne le résultat :

$$6+2\cdot\mathtt{X}$$

On tape :

$$2\ +\ \mathtt{X}\ \triangleright\ \times\ 3\ \triangle\ -\ \mathtt{X}$$

et on obtient :

$$(2+\mathtt{X})\cdot(3-\mathtt{X})$$

$\triangleright$ $\triangleright$ $\triangleright$ pour sélectionner l'expression
puis `ENTER` donne le résultat :

$$-(\mathtt{X}^2-\mathtt{X}-6)$$

- Exemple 2
  Si on veut taper :

  $$\mathtt{X}^2-3\cdot\mathtt{X}+1$$

  On tape :

  $$\mathtt{X}\ \mathtt{x^y}\ 2\ \triangleright\ -\ 3\ \mathtt{X}\ +\ 1$$

  Si on veut taper :

  $$-\mathtt{X}^2-3\cdot\mathtt{X}+1$$

  On tape :

  $$(-)\ \mathtt{X}\ \mathtt{x^y}\ 2\ \triangleright\triangleright\ -\ 3\ \mathtt{X}\ +\ 1$$

  En effet, il faut sélectionner $-\mathtt{X}^2$ avant de taper la suite.
- Exemple 3
  Si on veut taper :

  $$\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\frac{1}{5}$$

  Ici, le sommet de l'arbre est un + et il y a 4 sous arbres ; chacun de ces sous-arbres a comme sommet un ÷ et possède deux feuilles.
  On tape sur `CAS` du bandeau pour ouvrir l'éditeur d'équations, puis on écrit le premier sous -arbre :

  $$1\div 2$$

puis on sélectionne cet arbre avec

$$\triangleright$$

puis on tape

$$+$$

et le second sous-arbre :

$$1 \div 3$$

puis on sélectionne cet arbre avec

$$\triangleright$$

puis on tape

$$+$$

et le troisième sous-arbre :

$$1 \div 4$$

puis on sélectionne cet arbre avec

$$\triangleright$$

puis on tape

$$+$$

et le quatrième sous-arbre :

$$1 \div 5$$

puis on sélectionne cet arbre avec

$$\triangleright$$

Maintenant, l'expression voulue

$$\frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5}$$

se trouve écrite dans l'éditeur d'équations et $\frac{1}{5}$ est sélectionnée.
Parcourez l'arbre pour sélectionner

$$\frac{1}{3} + \frac{1}{4}$$

Il faut taper

$$\lhd \ \lhd$$

pour sélectionner $\frac{1}{3}$ puis

`SHIFT`$\rhd$

permet de sélectionner deux sous-arbres contigus, le sélectionné et son voisin de droite, ici :

$$\frac{1}{3}+\frac{1}{4}$$

Intérêt : On peut demander d'effectuer le calcul de la partie sélectionnée en tapant `ENTER`.
On obtient :

$$\frac{1}{2}+\frac{7}{12}+\frac{1}{5}$$

où la fraction $\frac{7}{12}$ est sélectionnée.
Si on veut effectuer maintenant le calcul partiel

$$\frac{1}{2}+\frac{1}{5}$$

il faut tout d'abord faire une permutation pour que $\frac{1}{2}$ et $\frac{1}{5}$ soient côte à côte en tapant

`SHIFT`$\lhd$

qui échange l'élément sélectionné avec son voisin de gauche.
On obtient :

$$\frac{7}{12}+\frac{1}{2}+\frac{1}{5}$$

et $\frac{7}{12}$ est sélectionné, puis

$\rhd$`SHIFT`$\rhd$

sélectionne

$$\frac{1}{2}+\frac{1}{5}$$

On peut alors faire à nouveau `ENTER`.

- En résumé : `SHIFT`$\rhd$ permet de sélectionner l'élément sélectionné et son voisin de droite.
  `SHIFT`$\lhd$ permet d'échanger l'élément sélectionné avec son voisin de gauche.
  L'élément sélectionné a changé de place, mais il reste sélectionné.

3.1.3 Comment modifier une expression

Si vous êtes en train de taper votre expression, la touche `DEL` vous permet d'effacer ce qui vient d'être tapé.

Si vous êtes en train d'effectuer des sélections vous pouvez :
- soit supprimer la sélection, sans supprimer l'expression, en tapant :

`DEL`

Le curseur se trouve alors à la fin de l'expression que l'on vient de désélectionner.
- soit remplacer la sélection par une expression, il suffit de taper l'expression voulue,
- soit transformer l'expression sélectionnée en lui appliquant une fonction du `CAS` : on appelle alors la fonction depuis le menu d'un des répertoires du `CAS`,
- soit supprimer l'expression sélectionnée en tapant :

`ALPHA SHIFT DEL` (`ALPHA CLEAR`)

- soit supprimer un opérateur unaire, sommet de l'arbre sélectionné, en tapant :

`SHIFT DEL` (`CLEAR`)

Par exemple pour remplacer `SIN(expr)` par `COS(expr)`, on efface `SIN` en sélectionnant `SIN(expr)`, puis `SHIFT DEL`, puis on tape `COS`.
- soit supprimer l'argument d'un opérateur binaire infixé et cet opérateur, en mettant en surbrillance l'argument à supprimer et en tapant :

`SHIFT DEL` (`CLEAR`)

Par exemple pour enlever $\mathtt{F(X)} =$ de $\mathtt{F(X)} = \mathtt{X}^2 - \mathtt{X} + \mathtt{1}$ on sélectionne `F(X)`, puis `SHIFT DEL`, et on obtient $\mathtt{X} = \mathtt{X}^2 - \mathtt{X} + \mathtt{1}$ (on a supprimer l'opérateur unaire `F`), puis `SHIFT DEL`, et on obtient $\mathtt{X}^2 - \mathtt{X} + \mathtt{1}$ (on a supprimer `X` et l'opérateur $=$).
- soit supprimer un opérateur binaire en éditant l'expression : on sélectionne

`Edit expr.`

du menu `TOOL` du bandeau et on fait la correction.
- soit recopier un élément de l'historique en tapant `HOME`. La recopie se fait à la place du curseur, ou à la place de la sélection, lorsqu'on appuie depuis l'historique sur `ENTER`, ou sur `ECHO` du bandeau.

Vous pouvez aussi utiliser les commandes `Cut, Copy, Paste` du menu `TOOL` du bandeau, pour supprimer, recopier, dupliquer des expressions comme dans un traitement de texte (cf 2.4).

3.1.4 Le mode curseur

Le mode curseur permet de sélectionner une grande expression rapidement. Pour passer en mode curseur, sélectionner :
`Cursor mode` du menu `TOOL`,
puis utiliser les flèches pour inclure votre sélection dans une boite (quand vous relachez la touche flèche, l'expression pointée par le curseur est encadrée),
puis `ENTER` pour sélectionner le contenu de la boite.

3.1.5 Pour tout voir

En sélectionnant `Change font` du menu `TOOL` du bandeau, on grossit, ou on diminue la taille de l'écriture : cela permet dans certains cas, de voir en entier une grande expression.
Si cela est insuffisant, il faut passer en mode curseur :
`Cursor mode` du menu `TOOL`, puis utiliser la flèche $\triangleright$
ou encore utiliser :
la touche `VIEWS`, puis utiliser la flèche $\triangleright$.

3.2 La saisie des fonctions du CAS

Lorsque vous êtes dans l'éditeur d'équations, vous pouvez utiliser toutes les fonctions du `CAS` et vous pouvez faire la saisie de différentes façons.
Principe général :
Quand vous avez écrit une expression dans l'éditeur d'équations, il suffit d'appuyer sur `ENTER` pour évaluer ce qui a été sélectionné, ou pour évaluer toute l'expression si rien n'a été sélectionné.

3.2.1 Comment écrire $\int$ et $\sum$

$\sum$ se trouvent sur le clavier, il suffit de taper :

$$\texttt{SHIFT} + \ (\sum)$$

. Le signe $\int$ se trouve aussi sur le clavier, il suffit de taper :

$$\texttt{SHIFT d/dX} \left(\int\right)$$

Les symboles $\int$ et $\sum$ sont considérés comme des fonctions préfixées ayant plusieurs arguments.
$\int$ et $\sum$ se placent automatiquement devant (fonctions préfixées) l'élément sélectionné, si il y en a un.
Le curseur se place aux endroits voulus et se déplace à l'aide de

$$\triangleright \ \triangleleft$$

Les expressions que l'on rentre suivent la loi de la sélection expliquée précédemment, mais il faut entrer dans le mode sélection avec $\triangle$.
ATTENTION, ne pas utiliser l'indice i pour définir la somme car i désigne le nombre complexe solution de $x^2 + 1 = 0$.
En mode numérique $\sum$ effectue des calculs approchés.
Par exemple :

$$\sum_{k=0}^{4} \frac{1}{k!} = 2.70833333334$$

alors que

$$1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} = \frac{65}{24}$$

Le symbole ! s'obtient en tapant `SHIFT` $\times$.
Il faut savoir que $\sum$ sait calculer symboliquement les sommes de fractions rationnelles et les series hypergéométriques qui admettent une primitive discrète.
Exemple :
On tape :

$$\sum_{\mathtt{K}=1}^{\infty} \frac{1}{\mathtt{K}.(\mathtt{K}+1)}$$

On sélectionne le tout, `ENTER` et on obtient :

$$1$$

3.2.2 Comment écrire les fonctions infixées

Ces fonctions s'écrivent entre leurs arguments, par exemple :
`AND | MOD ,`
On peut :
- soit les écrire en mode `Alpha` (pour `AND MOD`), puis taper les arguments,
- soit les appeler depuis le menu d'un répertoire du `CAS` ou à l'aide d'une touche du clavier, à condition d'avoir écrit et sélectionné le premier argument.
On passe d'un argument à l'autre à l'aide des flèches : $\rhd$ $\lhd$
La virgule `,` permet d'écrire un nombre complexe.
On écrit `1+2.i` ou `(1,2)` et les parenthèses se mettent automatiquement quand on tape la virgule.
Si vous voulez taper `(-1,2)` il faut bien sûr sélectionner `-1` avant de taper la virgule.

3.2.3 Comment écrire les fonctions préfixées

Ces fonctions s'écrivent avant leurs arguments, c'est le cas général.
Dans ce cas tout est possible ! ! !
On peut taper le premier argument, le sélectionner puis appeler la fonction à l'aide d'un menu.
ou bien, on peut appeler la fonction depuis un menu ou la taper en mode `Alpha`, puis taper son ou ses arguments.
On va détailler sur l'exemple suivant les différentes saisies possibles.
Exemple :
Vous voulez factoriser l'expression $x^2 - 4$ puis, avoir sa valeur pour $x = 4$. Vous savez que `FACTOR` sert à factoriser et que cette fonction se trouve dans le menu `ALGB`.
Vous savez que `SUBST` sert à substituer dans une expression une variable par une valeur et que cette fonction se trouve dans le menu `ALGB`.

Première possibilité : saisie avant les arguments

Vous appuyez sur la touche `F2` qui active `ALGB` du bandeau, et vous mettez en surbrillance `FACTOR` puis `ENTER`.
`FACTOR(` ◀`)` s'inscrit dans l'éditeur avec le curseur entre les parenthèses.

Vous entrez votre expression avec les règles de sélection vues précèdemment :

$$\mathtt{X}\ \mathtt{x^y}\ 2\ \triangleright - 4 \triangleright\ \triangleright$$

ce qui sélectionne :

$$\mathtt{FACTOR}(\mathtt{X}^2 - 4)$$

puis `ENTER` donne le résultat :

$$(\mathtt{X}+2).(\mathtt{X}-2)$$

Le résultat est sélectionné et remplace la commande.
On ne le voit pas, mais aprés chaque `ENTER`, il y a sauvegarde dans l'historique, ainsi $\mathtt{FACTOR}(\mathtt{X}^2 - 4)$ et $(\mathtt{X}+2).(\mathtt{X}-2)$ se sont inscrits dans l'historique.
Maintenant vous effacez le résultat précédent avec `ALPHA SHIFT DEL (CLEAR)` car ce résultat est sélectionné.
Vous appuyez sur la touche qui active `ALGB` du bandeau, et vous mettez en surbrillance `SUBST` puis `ENTER`.

$$\mathtt{SUBST}(\blacktriangleleft, \bullet)$$

s'inscrit dans l'éditeur avec le curseur entre les parenthèses à la place du premier argument.
Vous entrez votre expression avec les règles de sélection vues précèdemment.
ATTENTION ! ici `SUBST` a deux arguments il faut donc entrer dans le mode sélection avec $\triangle$:

$$\mathtt{X}\ \mathtt{x^y}\ 2\ \triangle\ \triangle\ -\ 4 \triangleright \mathtt{X} = 4\ \triangleright\ \triangleright$$

ce qui sélectionne :

$$\mathtt{SUBST}(\mathtt{X}^2 - 4, \mathtt{X} = 4)$$

puis `ENTER` donne le résultat :

$$4^2 - 4$$

Ce résultat remplace la commande, il est sélectionné, enfin `ENTER` donne le résultat simplifié :

$$12$$

Bien sûr $\mathtt{SUBST}(\mathtt{X}^2 - 4, \mathtt{X} = 4)$, $4^2 - 4$ et 12 se sont inscrits dans l'historique.

REMARQUE :
Quand on saisit une fonction du CAS avant ses arguments, on peut la taper en mode Alpha avec ses parenthèses.

Deuxième possibilité : saisie après les arguments

On tape d'abord l'expression que l'on sélectionne avec les règles de sélection vues précèdemment.
Ici on tape :

$$\texttt{X}\ \texttt{x}^{\texttt{y}}\ 2\ \rhd - 4 \rhd\ \rhd$$

puis on appelle FACTOR :
vous appuyez sur la touche qui active ALGB du bandeau, et vous mettez en surbrillance FACTOR, puis ENTER.
On obtient :

$$\texttt{FACTOR}(\texttt{X}^2 - 4)$$

puis ENTER donne le résultat :

$$(\texttt{X}+2).(\texttt{X}-2)$$

Ce résultat remplace la commande et est sélectionné.
Bien sûr $\texttt{FACTOR}(\texttt{X}^2 - 4)$ et $(\texttt{X}+2).(\texttt{X}-2)$ se sont inscrits dans l'historique.
Vous remarquez que le résultat est sélectionné, vous êtes donc prêt à appliquer une autre commande à votre résultat.
Vous pouvez donc appeller SUBST : vous appuyez sur la touche qui active ALGB du bandeau, et vous mettez en surbrillance SUBST puis ENTER.

$$\texttt{SUBST}((\texttt{X}+2).(\texttt{X}-2), \blacktriangleleft)$$

s'inscrit dans l'éditeur avec ses parenthèses, avec votre expression comme premier argument et avec le curseur à la place du deuxième argument.
Il ne vous reste plus qu'à taper :
$\texttt{X} = 4$ puis $\rhd$ $\rhd$ et ENTER.
On obtient :

$$(4+2).(4-2)$$

puis `ENTER` pour obtenir :

`12`

Bien sûr SUBST($X^2 - 4, X = 4$), $(4 + 2).(4 - 2)$ et 12 se sont inscrits dans l'historique.
REMARQUE :
Au cours de l'écriture d'une expression, vous pouvez appeler une fonction du `CAS`, cette fonction aura comme premier argument ce qui est sélectionné ou rien, si rien n'a été sélectionné, le curseur se trouve au bon endroit pour compléter les arguments.

3.3 Les variables

Vous pouvez stocker des objets dans des variables, et les réutiliser en utilisant le nom de la variable.
ATTENTION ! ! !
1- Les variables utilisées dans le `CAS` ne sont pas utilisables dans `HOME` et réciproquement.
2- On utilise `STO▷` pour stocker un objet dans une variable de `HOME` ou de l'éditeur de programme et on le note dans la suite `STO▷` ou ▷.
3- Dans le `CAS` il faut utiliser la commande `STORE` (cf 3.3.2) pour stocker des valeurs dans des variables.
4- La touche `VARS` affiche un menu qui contient toutes les variables qui sont à votre disposition.
Cette touche pressée lorsque l'on est dans `HOME` fait apparaître les noms des variables autorisées dans `HOME` et dans les `Aplets`.
Cette touche pressée lorsque l'on est dans l'éditeur d'équations fait apparaître les noms des variables définies dans le `CAS`.

3.3.1 STO▷

`STO▷` permet de stocker un objet dans une variable de `HOME`.
Les noms des variables numériques de `HOME` sont les 26 lettres de l'alphabet et les noms des variables symboliques de `HOME` sont `S1..S5`
ATTENTION Les variables `A..Z` sont toujours à votre disposition et contiennent toujours une valeur réelle.
Par exemple en utilisant le `STO▷` du bandeau de `HOME` ou de l'éditeur de programme, on tape :
`1 STO ▷ A`

cela se traduit à l'écran par :
1 ▷ A et a pour effet d'écraser la valeur précédente de A et de la remplacer par 1.
A est évalué et désigne le contenu de A.

Remarque :
La variable symbolique S1 de HOME sert de variable courante lorsqu'on veut utiliser certaines fonctions du CAS depuis HOME.
Exemple : même si il y a X dans VX, on écrit dans HOME :
$\mathtt{DERVX}(\mathtt{S1}^2 + 2 \times \mathtt{S1})$
pour obtenir $2 \times \mathtt{S1} + 2$.

3.3.2 STORE

Dans le CAS, il faut utiliser la commande STORE pour stocker un objet dans une variable, ou utiliser la touche VARS depuis l'éditeur d'équations (puis NEW ou EDIT du bandeau cf 2.5.3).
On a droit à n'importe quel nom de variable.
STORE se trouve dans le menu ALGB du bandeau de l'éditeur d'équations.
Exemple :
On tape :

$$\mathtt{STORE}(\mathtt{X}^2 - 4, \mathtt{ABC})$$

ou on tape :

$$\mathtt{X}^2 - 4$$

que l'on sélectionne, puis on appelle STORE,
puis on tape ABC
ENTER valide la définition de la variable ABC.
Pour détruire la variable, il faut utiliser la touche VARS depuis l'éditeur d'équations (puis PURGE du bandeau cf 2.5.3), ou utiliser la commande UNASSIGN du menu ALGB, on tape par exemple :

$$\mathtt{UNASSIGN}(\mathtt{ABC})$$

3.3.3 Les variables prédéfinies du CAS

VX contient le nom de la variable symbolique courante.
C'est en général X, il ne faut donc pas utiliser X comme nom de variable numérique, ou effacer le contenu de X à l'aide de la commande UNASSIGN du menu ALGB avant de faire du calcul symbolique, en tapant par exemple : UNASSIGN(X).

`EPS` contient la valeur de epsilon utilisé dans la commande `EPSX0` (cf 4.13.2).
`MODULO` contient la valeur de p, pour faire du calcul symbolique dans $Z/p.Z$. On peut changer la valeur de p grâce à la commande `MODSTO` du menu `MODULAR` en tapant par exemple : `MODSTO(13)` pour donner à p la valeur 13, ou utiliser `CFG` des menus du `CAS`.
`PERIOD` doit contenir la période de la fonction dont on veut les coefficients de Fourier (cf 4.11.6).
`PRIMIT` contient la primitive de la dernière fonction intégrée.
`REALASSUME` contient la liste des noms des variables symboliques que l'on considère comme réelles, lorsqu'on a choisi dans le menu de configuration `CFG` l'option `Cmplx vars`, ce sont par défaut :
`X, Y, t, S1, S2` et toutes les variables d'intégration utilisées.
Bien sûr, si dans le menu de configuration `CFG`, vous choisissez l'option `Real vars` toutes les variables symboliques sont considérées comme réelles (cf 4.1.1).
On peut aussi faire des suppositions sur le domaine de définition d'une variable comme par exemple `X>1`.
Dans ce cas, il faut utiliser la commande `ASSUME(X>1)` pour que `REALASSUME` contienne `X>1`.
La commande `UNASSUME(X)` enlèvera toutes les hypothèses faites sur `X`.
Pour voir ces variables et aussi celles que vous avez définies dans le `CAS`, il faut appuyer sur `VARS` depuis l'éditeur d'équations (cf 2.5.3).

Chapitre 4

Les fonctions de Calcul formel

4.1 Le bandeau du CAS

Seul le menu `TOOL` contient des commandes, les autres menus permettent la mise à jour de la configuration et contiennent des fonctions algébriques que l'on peut taper en mode `Alpha`.

4.1.1 `CFG`

Tous les menus sauf `TOOL`, affichent l'état de votre configuration et vous avez la possibilité d'en changer.
Par exemple vous voyez en première ligne d'un menu :

$$\texttt{CFG}: \texttt{R} = \texttt{X S}$$

cela veut dire que vous êtes en mode réel exact et que `X` est la variable courante et que vous êtes en mode pas à pas (`S`).
On met en surbrillance `CFG` et on appuie sur `OK`.
Il apparait un menu d'en tête :

$$\texttt{CFG}: \texttt{R} = \texttt{STEP} \uparrow \texttt{'X'}\ 13\ ||$$

cela veut dire que vous êtes en mode réel exact, que le mode pas à pas est sélectionné, que les polynômes sont écrits selon les puissances

croissantes, que X est la variable courante, que les calculs modulaires se feront dans Z/13Z ($p = 13$), et que vous êtes en mode rigourous (on met des valeurs absolues).
Vous pouvez changer cette configuration en sélectionnant ce qui vous convient parmi :
Quit config (lorsqu'on a fini les changements),
Complex (ou Real),
Approx (ou Exact),
Direct (ou Step/Step si on veut être en mode pas à pas),
$1 + x + x^2$... (ou ...$x^2 + x + 1$ pour l'écriture des polynômes)
Sloopy (ou Rigourous pour ne pas mettre les valeurs absolues)
Num. factor (ou Symb factor),
Cmplx vars (ou Real vars pour que toutes les variables symboliques soient considérées comme réelles cf 3.3.3),
English (ou Français pour avoir l'aide en ligne en français)
Default cfg (configuration R = STEP ↓ X 13 ||).
Appuyer sur OK pour valider chacun de vos choix.
On sort du menu CFG en appuyant sur CANCEL ou en validant Quit config par OK.
Le nom de la variable courante contenu dans VX et la valeur de la variable MODULO peuvent se changer à l'aide des touches SHIFT SYMB (SETUP) ou l'aide de la touche VARS (cf 2.5.6 et 2.5.3).
Remarque : dans le CAS les angles sont toujours exprimés en radians.
Vous pouvez aussi changer votre configuration à l'aide des touches SHIFT SYMB (SETUP) se reporter pour cela à la section 2.5.6

4.1.2 TOOL

On se reportera à la section 2.4 pour la description des fonctions qui se trouvent dans le répertoire TOOL.
Cursor mode
Edit expr.
Change font
Cut
Copy
Paste

4.1.3 ALGB

COLLECT
DEF
EXPAND
FACTOR
PARTFRAC
QUOTE
STORE
|
SUBST
TEXPAND
UNASSIGN

4.1.4 DIFF&INT

DERIV
DERVX
DIVPC
FOURIER
IBP
INTVX
LIMIT
PREVAL
RISCH
SERIES
TABVAR
TAYLOR0
TRUNC

4.1.5 REWRITE

DISTRIB
EPSX0
EXPLN
EXP2POW
FDISTRIB
LIN
LNCOLLECT

POWEXPAND
SINCOS
SIMPLIFY
XNUM
XQ

4.1.6 SOLVE

DESOLVE
ISOLATE
LDEC
LINSOLVE
SOLVE
SOLVEVX

4.1.7 TRIG

ACOS2S
ASIN2C
ASIN2T
ATAN2S
FOURIER
HALFTAN
SINCOS
TAN2CS2
TAN2SC
TAN2SC2
TCOLLECT
TEXPAND
TLIN
TRIG
TRIGCOS
TRIGSIN
TRIGTAN

4.1.8 La touche MATH

En plus des répertoires ci-dessus (ALGEBRA DIFF&INT REWRITE TRIG SOLVE) on trouve :
Complex... (i ABS ARG CONJ DROITE FLOOR IM MOD - RE SIGN)
Constant... (e i ∞ pi)
Hyperb.... (ACOSH ASINH ATANH COSH SINH TANH)
Integer... (DIVIS EULER FACTOR GCD IEGCD IQUOT IREMAINDER ISPRIME ? LCM NEXTPRIME PREVPRIME)
Modular... (ADDTMOD DIVMOD EXPANDMOD FACTORMOD GCDMOD INVMOD MODSTO MULTMOD POWMOD SUBTMOD)
Polynom.... (EGCD FACTOR GCD HERMITE LCM LEGENDRE PARTFRAC PROPFRAC PTAYL QUOT REMAINDER TCHEBYCHEFF)
Tests... (ASSUME UNASSUME $> \geq < \leq == \neq$ AND OR NOT IFTE)
On pourra se reporter aux sections 2.4 et 2.5.1, pour avoir la description des différents répertoires.

4.2 Le pas à pas

Le mode pas à pas (Step/Step ou en abrégé S) est choisi quand on veut avoir le détail des calculs.
Le détail des calculs s'affiche sur un écran et il faut appuyer sur OK du bandeau pour avoir le "pas" suivant.
Mais quelquefois, l'écran n'est pas assez grand pour afficher toutes les informations : on le voit grâce à une flèche qui se trouve soit en bas (▼), soit en haut (▲) de l'écran. Il faut alors se servir des flèches $\bigtriangledown\bigtriangleup$ pour faire dérouler l'écran et voir ainsi ce qui manque.
Si on ne veut pas le détail des calculs, il faut choisir le mode Direct (en abrégé D).

4.3 Ecriture normale

La calculatrice peut gérer des nombres entiers en précision infinie, essayez :

$$100!$$

Le symbole ! s'obtient en tapant : SHIFT $\times$
L'écriture décimale de 100! étant très longue, on peut voir le résultat grâce à la touche VIEWS.

4.3.1 DEF

Soit l'exercice suivant :
Calculer les six premiers nombres de Fermat $F_k = 2^{2^k} + 1$ pour $k = 1..6$ et dire s'ils sont premiers.
On tape l'expression

$$2^{2^2} + 1$$

On trouve 17, puis on lance la commande `ISPRIME?()`. Cette commande se trouve dans le menu `Integer` de la touche `MATH`.
La réponse est `1.`, ce qui veut dire `vrai`. Grâce à l'historique (touche `HOME`) je recopie l'expression $2^{2^2} + 1$ dans l'éditeur d'équations et je la modifie en

$$2^{2^3} + 1$$

Ou bien, et c'est la meilleure méthode, on définit la fonction `F(K)` à l'aide de `DEF` du menu `ALGB` du bandeau en tapant :

$$\mathtt{DEF}(\mathtt{F}(\mathtt{K}) = \mathtt{2}^{\mathtt{2}^{\mathtt{K}}} + \mathtt{1})$$

La réponse est $\mathtt{2}^{\mathtt{2}^{\mathtt{K}}} + \mathtt{1}$ et `F` s'inscrit parmi les variables (appuyer sur `VARS` pour le vérifier).
Pour $K = 5$ on tape :

$$\mathtt{F}(\mathtt{5})$$

On obtient :

$$4294967297$$

On peut factoriser F_5 avec `FACTOR` que l'on trouve dans le menu : `ALGB` du bandeau.
On tape :

$$\mathtt{FACTOR}(\mathtt{F}(\mathtt{5}))$$

On obtient

$$641 \cdot 6700417$$

Pour `F(6)` on trouve :

$$18446744073709551617$$

On factorise avec `FACTOR`, on trouve :

$$274177 \cdot 67280421310721$$

ATTENTION à la différence entre :

$$2\ .\ 5 = \frac{5}{2}$$

et

$$2 \cdot 5 = 10$$

4.4 Les entiers (et les entiers de Gauss)

Toutes les fonctions de ce paragraphe se trouve dans le menu `Integer` de la touche `MATH`.
Pour certaines fonctions, on peut utiliser des entiers de Gauss, nombres de la forme $a + ib$ avec a et b entiers, à la place des entiers.

4.4.1 DIVIS

`DIVIS` donne la liste des diviseurs d'un entier.
On tape :

`DIVIS(12)`

On obtient :

`12 OR 6 OR 3 OR 4 OR 2 OR 1`

4.4.2 EULER

`EULER` désigne l'indicadrice d'EULER d'un entier.
EULER(n) est égale au cardinal de l'ensemble des nombres inférieurs à n et premiers avec n.
On tape :

`EULER(21)`

On obtient :

`12`

En effet l'ensemble :
E={2,4,5,7,8,10,11,13,15,16,17,19} correspond aux nombres, plus petits que 21, premiers avec 21, et E a comme cardinal 12.

4.4.3 FACTOR

`FACTOR` décompose l'entier en produit de facteurs premiers.
On tape :

$$\texttt{FACTOR}(90)$$

On obtient :

$$2 \cdot 3^2 \cdot 5$$

4.4.4 GCD

`GCD` désigne le PGCD de deux entiers.
On tape :

$$\texttt{GCD}(18, 15)$$

On obtient :

$$3$$

En mode pas à pas, on tape :

$$\texttt{GCD}(78, 24)$$

On obtient : 78 mod 24 = 6
24 mod 6 = 0
`Result 6`
`ENTER` renvoie 6 dans l'éditeur d'équations

4.4.5 IEGCD

`IEGCD(A,B)` désigne le PGCD étendu (identité de Bézout) de deux entiers.
`IEGCD(A,B)` renvoie `U AND V = D` avec `U, V, D` vérifiant :
`AU+BV=D` et `D=PGCD(A,B)`.
On tape :

$$\texttt{IEGCD}(48, 30)$$

On obtient :

$$2 \texttt{ AND } -3 = 6$$

En effet :

$$2 \cdot 48 + (-3) \cdot 30 = 6$$

En mode pas à pas on obtient :
`z=u*48+v*30`

```
[48,1,0]
[30,0,1]*-1
[18,1,-1]*-1
[12,1,-2]*-1
[6,2,-3]*-2
Result : [6,2,-3]
```

puis `ENTER`,

$$\texttt{2 AND } -3 = 6$$

s'écrit dans l'éditeur d'équations.

4.4.6 IQUOT

`IQUOT` désigne le quotient entier de la division euclidienne de deux entiers.
On tape :

$$\texttt{IQUOT}(148, 5)$$

On obtient :

$$\texttt{29}$$

En mode pas à pas, la division se fait comme à l'école :

$$\begin{array}{rcl} 148 & | & 5 \\ 48 & | & --- \\ 3 & | & 29 \end{array}$$

`OK` pour exécuter la division au pas à pas, puis `ENTER` et 29 s'inscrit dans l'éditeur d'équations.

4.4.7 IREMAINDER MOD

`IREMAINDER` désigne le reste entier de la division euclidienne de deux entiers.
`IREMAINDER` se trouve dans le menu `Integer` et `MOD` se trouve dans le menu `Complex` de la touche `MATH`.
On tape :

$$\texttt{IREMAINDER}(148, 5)$$

ou

$$\texttt{148 MOD 5}$$

On obtient :

3

IREMAINDER travaille avec des entiers ou des entiers de Gauss, c'est ce qui le différencie de MOD.
Exemple : IREMAINDER(2 + 3.i, 1 + i) renvoie i
MOD accepte des réels ($7.5 \mod 2 = 1.5$) mais pas des entiers de Gauss.
Essayer :

IREMAINDER(148!, 5! + 2)

(! s'obtient avec SHIFT $\times$).

En mode pas à pas, la division se fait comme à l'école, avec l'algorithme dit de la "potence" (cf 4.4.6).

4.4.8 ISPRIME ?

ISPRIME ?(N) renvoie 1. (vrai) si N est pseudo-premier et renvoie 0. (faux) si N n'est pas premier.
Définition : Pour les nombres inférieurs à 10^{14} être pseudo-premier et premier c'est la même chose ! ...mais au delà de 10^{14} un nombre pseudo-premier est premier avec une probabilité très forte (cf l'algorithme de Rabin section7.6).
On tape :

ISPRIME?(13)

On obtient :

1.

On tape :

ISPRIME?(14)

On obtient :

0.

4.4.9 LCM

LCM désigne le PPCM de deux entiers.
On tape :

LCM(18, 15)

On obtient :

90

4.4.10 NEXTPRIME

NEXTPRIME(N) désigne le premier nombre pseudo-premier trouvé après N.
On tape :

NEXTPRIME(75)

On obtient :

79

4.4.11 PREVPRIME

PREVPRIME(N) désigne le premier nombre pseudo-premier trouvé avant N.
On tape :

PREVPRIME(75)

On obtient :

73

4.5 Le calcul modulaire

Toutes les fonctions de ce paragraphe se trouve dans le menu Modular de la touche MATH.
On peut faire des calculs modulo p c'est à dire dans Z/pZ ou dans $Z/pZ[X]$.
ATTENTION, pour certaines commandes il faut choisir un nombre p premier.
DANS LA SUITE, LES EXEMPLES SERONT TRAÎTÉS AVEC p=13.
On suppose donc que l'on a tapé :

MODSTO(13)

ou que l'on a changé MODULO dans la fenêtre ouverte avec les touches SHIFT SYMB (SETUP).
La représentation choisie est la représentation symétrique (-1 au lieu de 6 modulo 7).

4.5.1 ADDTMOD

ADDTMOD réalise une addition dans $Z/pZ[X]$.
On tape :

ADDTMOD(11X + 5, 8X + 6)

On obtient :

$$\mathtt{6X-2}$$

4.5.2 DIVMOD

Les arguments sont deux polynômes $A[X]$ et $B[X]$. Le résultat est la fraction rationnelle $\frac{A[X]}{B[X]}$ simplifiée dans $Z/pZ[X]$.
On tape :

$$\mathtt{DIVMOD(2X^2+5,5X^2+2X-3)}$$

On obtient :

$$\frac{\mathtt{5X+3}}{\mathtt{6X+6}}$$

4.5.3 EXPANDMOD

EXPANDMOD a comme argument une expression polynomiale.
EXPANDMOD développe cette expression dans $Z/pZ[X]$.
On tape :

$$\mathtt{EXPANDMOD((2X^2+12).(5X-4))}$$

On obtient :

$$-(\mathtt{3X^3-5X^2+5X-4})$$

4.5.4 FACTORMOD

FACTORMOD a comme argument un polynôme.
FACTORMOD factorise ce polynôme dans $Z/pZ[X]$ à condition que l'on ait $\mathtt{p} \leq 97$ et p premier.
On tape :

$$\mathtt{FACTORMOD(-(3X^3-5X^2+5X-4))}$$

On obtient :

$$-((\mathtt{3X-5})(\mathtt{X^2+6}))$$

4.5.5 GCDMOD

GCDMOD a deux polynômes comme arguments.
GCDMOD calcule le PGCD des deux polynômes dans $Z/pZ[X]$.
On tape :

$$\mathtt{GCDMOD(2X^2+5,5X^2+2X-3)}$$

On obtient :

$$-(\mathtt{4X-5})$$

4.5.6 INVMOD

INVMOD a comme argument un entier.
INVMOD calcule l'inverse de cet entier dans Z/pZ.
On tape :

$$\texttt{INVMOD(5)}$$

On obtient (car $5 \times -5 = -25 = 1 \pmod{13}$) :

$$-5$$

4.5.7 MODSTO

On met dans la variable MODULO, la valeur de p grâce à la commande MODSTO.
Ici, les exemples sont traîtés avec p=13 qui est la valeur par défaut, sinon on suppose que l'on a tapé :

$$\texttt{MODSTO(13)}$$

4.5.8 MULTMOD

MULTMOD réalise une multiplication dans $Z/pZ[X]$.
On tape :

$$\texttt{MULTMOD}(11X + 5, 8X + 6)$$

On obtient :

$$-(3X^2 - 2X - 4)$$

4.5.9 POWMOD

POWMOD(A,N) calcule A à la puissance N dans Z/pZ, et POWMOD(A(X),N) calcule A(X) à la puissance N dans $Z/pZ[X]$.
Le contenu p de MODULO doit être un nombre premier inférieur à 100.
On tape :

$$\texttt{POWMOD}(11, 195)$$

On obtient :

$$5$$

En effet, $11^12 = 1 \mod 13$ donc $11^195 = 11^3 = 5 \mod 13$ On tape :

$$\texttt{POWMOD}(2X + 1, 5)$$

On obtient :

$$\mathtt{6.X^5 + 2.X^4 + 2.X^3 + X^2 - 3.X + 1}$$

car :
$10 = -3 \pmod{13}$ $40 = 1 \pmod{13}$ $80 = 2 \pmod{13}$ $32 = 6 \pmod{13}$.

4.5.10 SUBTMOD

SUBTMOD réalise une soustraction dans $Z/pZ[X]$.
On tape :

$$\mathtt{SUBTMOD(11X + 5, 8X + 6)}$$

On obtient :

$$\mathtt{3X - 1}$$

4.6 Les rationnels

Essayez :

$$\frac{123}{12} + \frac{57}{21}$$

Vous sélectionnez puis ENTER la réponse est :

$$\frac{363}{28}$$

Si on applique la fonction XNUM du menu REWRITE, ou si on appuie sur la touche NUM, la réponse est :

$$12.9642857143$$

Si on mélange les deux représentations par exemple :

$$\frac{1}{2} + 0\ .\ 5$$

la caculatrice demande à passer en mode approx pour faire le calcul ; il faut alors répondre yes pour obtenir :

$$1.$$

Revenez ensuite en mode exact (CFG etc...).

4.6.1 PROPFRAC

PROPFRAC se trouve dans le menu Polynom. de la touche MATH.
PROPFRAC($\frac{A}{B}$) écrit la fraction $\frac{A}{B}$ sous la forme :

$$Q + \frac{R}{B} \quad avec \quad 0 \leq R < B$$

On tape :

$$\text{PROPFRAC}(\frac{43}{12})$$

On obtient :

$$3 + \frac{7}{12}$$

4.7 Les réels

Essayez :

$$\text{EXP}(\pi \times \sqrt{20})$$

Vous sélectionnez, puis ENTER la réponse est :

$$\text{EXP}(2 \times \sqrt{5} \times \pi)$$

Si on applique la fonction XNUM du menu REWRITE la réponse est :

$$1263794.7537$$

Vous trouverez dans le menu Complex de la touche MATH, les fonctions FLOOR et MOD expliquées dans ce qui suit :

4.7.1 FLOOR

FLOOR a comme argument un nombre réel, et renvoie sa partie entière.
On tape :

$$\text{FLOOR}(3.53)$$

On obtient :

$$3$$

4.7.2 MOD

`MOD` est une fonction infixée ayant comme argument deux nombres entiers.
`MOD` renvoie le reste de la division euclidienne des arguments.
On tape :

`3 MOD 2`

On obtient :

`1`

4.8 Les complexes

Notation : les complexes de la forme $a + b.i$, avec a et b réels, peuvent être notés (a, b) ou $a + b.i$.
Les opérations `+,-,*,/,^` sont effectuées.
On tape :

$$(1 + 2.\mathtt{i})^2$$

Vous sélectionnez, puis `ENTER`.
Si on n'est pas en mode `complex`, la caculatrice demande à changer de mode, il faut alors répondre `yes` pour obtenir la réponse :

$$-(3 - 4 \cdot \mathtt{i})$$

Il faut noter que cette expression ne sera pas simplifiée davatange (les résultats mettront toujours en évidence un nombre complexe de partie réelle positive en mode exact).
Vous trouverez dans le menu `Complex` de la touche `MATH`, les fonctions suivantes ayant comme paramètre une expression à valeur complexe :
`DROITE` a comme paramètre deux nombres complexes z_1, z_2.
`DROITE` renvoie alors l'équation de la droite passant par les deux points d'affixe z_1, z_2.
`ARG` pour déterminer l'argument du paramètre,
`ABS` pour déterminer le module du paramètre,
`CONJ` pour déterminer le conjugué du paramètre,
`RE` pour déterminer la partie réelle du paramètre,
`IM` pour déterminer la partie imaginaire du paramètre,
`-` pour déterminer l'opposé du paramètre,
`SIGN` pour déterminer le quotient du paramètre par son module.

4.8.1 ARG

On tape :

$$\mathtt{ARG}(3 + 4.\mathtt{i})$$

On obtient (car dans le `CAS`, on est en `Radians`) :

$$\mathtt{ATAN}(\frac{4}{3})$$

Remarque :
Vous pouvez faire le même calcul dans `HOME` mais vous obtenez un résultat numérique `0.64250...`(si vous êtes en `Radians`).
Dans `HOME` il faut taper :

$$\mathtt{ARG}(\mathtt{XQ}(3 + 4.\mathtt{i}))$$

pour obtenir :

$$\mathtt{ATAN}(\frac{4}{3})$$

4.8.2 CONJ

On tape :

$$\mathtt{CONJ}(1 + 2.\mathtt{i})$$

On obtient :

$$1 - 2.\mathtt{i}$$

ATTENTION si vous choisissez `Real vars` dans le menu de configuration `CFG` vous aurez `CONJ(Z)=Z`, et si vous choisissez `Cmplx vars`, `CONJ(Z)` sera différent de `Z` à condition que `Z` ne figure pas dans la liste que contient la variable `REALASSUME`. Il est souvent préférable d'écrire l'expression quotée :
`QUOTE(expression)`, pour éviter une réécriture de cette expression par exemple, si on a séléctionné `Real vars` et si on tape :

$$\mathtt{SUBST}(\mathtt{QUOTE}(\mathtt{CONJ}(\mathtt{Z})), \mathtt{Z} = 1 + \mathtt{i})$$

On obtient :

$$\mathtt{CONJ}(1 + \mathtt{i})$$

alors que :

$$\mathtt{SUBST}(\mathtt{CONJ}(\mathtt{Z}), \mathtt{Z} = 1 + \mathtt{i})$$

donne :

$$\mathtt{1 + i}$$

Bien sûr, si on a séléctionné `Cmplx vars` et que `Z` ne figure pas dans la liste que contient la variable `REALASSUME` :

$$\mathtt{SUBST(CONJ(Z), Z = 1 + i)}$$

donne :

$$\mathtt{CONJ(1 + i)}$$

4.8.3 DROITE

On tape :

$$\mathtt{DROITE((1, 2), (0, 1))}$$

ou
On tape :

$$\mathtt{DROITE(1 + 2.i, i)}$$

On obtient :

$$\mathtt{y = x - 1 + 2}$$

puis `ENTER` donne :

$$\mathtt{y = x + 1}$$

4.9 Les expressions algébriques

Toutes les fonctions de ce paragraphe se trouve dans le menu `ALGB` du bandeau.

4.9.1 COLLECT

`COLLECT` a comme paramètre une expression.
`COLLECT` factorise cette expression sur les entiers.
Exemples :
Factoriser sur les entiers :

$$x^2 - 4$$

On tape :

$$\mathtt{COLLECT(X^2 - 4)}$$

On trouve en mode réel :

$$(\mathtt{X}+2)\cdot(\mathtt{X}-2)$$

Factoriser sur les entiers :

$$x^2-2$$

On tape :

$$\mathtt{COLLECT}(\mathtt{X}^2-2)$$

On trouve :

$$\mathtt{X}^2-2$$

4.9.2 EXPAND

EXPAND a comme paramètre une expression.
EXPAND développe et simplifie cette expression.
On tape :

$$\mathtt{EXPAND}((\mathtt{X}^2+\sqrt{2}\cdot\mathtt{X}+1)\cdot(\mathtt{X}^2-\sqrt{2}\cdot\mathtt{X}+1))$$

On obtient :

$$\mathtt{X}^4+1.$$

4.9.3 FACTOR

FACTOR a comme paramètre une expression.
FACTOR factorise cette expression.
Exemple :
Factoriser

$$x^4+1$$

On tape :

$$\mathtt{FACTOR}(\mathtt{X}^4+1)$$

On trouve FACTOR dans le menu de ALGB.
On trouve en mode réel :

$$(\mathtt{X}^2+\sqrt{2}\cdot\mathtt{X}+1)\cdot(\mathtt{X}^2-\sqrt{2}\cdot\mathtt{X}+1)$$

On trouve en mode complexe (pour cela utiliser CFG) :

$$\frac{(2.\mathtt{X}+(1+\mathtt{i}).\sqrt{2}).(2.\mathtt{X}-(1+\mathtt{i})\cdot\sqrt{2})}{16}\times$$
$$\frac{(2.\mathtt{X}+(1-\mathtt{i}).\sqrt{2}).(2.\mathtt{X}-(1-\mathtt{i}).\sqrt{2})}{16}$$

4.9.4 |

| est un opérateur infixé utile pour remplacer une variable dans une expression (un peu comme la fonction SUBST).
On tape :

$$\mathtt{X}^2 - 1 \,|_{\mathtt{X} = 2}$$

On obtient :

$$2^2 - 1$$

4.9.5 SUBST

SUBST a deux paramètres : une expression dépendant d'un paramètre et une égalité (paramètre=valeur de substitution).
SUBST effectue la substitution demandée dans l'expression.
On tape :

$$\mathtt{SUBST}(\mathtt{A}^2 + 1, \mathtt{A} = 2)$$

On obtient :

$$2^2 + 1$$

4.10 Les polynômes

Toutes les fonctions de ce paragraphe se trouve dans le menu Polynom. de la touche MATH.

4.10.1 DEGREE

DEGREE a comme argument un polynôme de la variable courante.
DEGREE renvoie le degré de ce polynôme.
Attention : le degré d'un polynôme nul est égal à -1.
On tape :

$$\mathtt{DEGREE}(\mathtt{X}^2 + \mathtt{X} + 1)$$

On obtient :

$$2$$

4.10.2 EGCD

Il s'agit de l'identité de Bézout (Extended Greatest Common Divisor). $\mathtt{EGCD(A(X),B(X))}$ renvoie $\mathtt{U(X)\ AND\ V(X)} = \mathtt{D(X)}$ avec D,U,V vérifiant :

$$\mathtt{D(X)} = \mathtt{U(X)} \cdot \mathtt{A(X)} + \mathtt{V(X)} \cdot \mathtt{B(X)}$$

On tape :

$$\mathtt{EGCD}(\mathtt{X}^2 + 2 \cdot \mathtt{X} + 1, \mathtt{X}^2 - 1)$$

On obtient :

$$1\ \mathtt{AND}\ -1 \ = \ 2 \cdot \mathtt{X} + 2$$

On tape :

$$\mathtt{EGCD}(\mathtt{X}^2 + 2 \cdot \mathtt{X} + 1, \mathtt{X}^3 - 1)$$

On obtient :

$$-(\mathtt{X} + 2)\ \mathtt{AND}\ 1 \ = \ 3 \cdot \mathtt{X} + 3$$

4.10.3 FACTOR

FACTOR a pour argument un polynôme.
FACTOR factorise ce polynôme.
On tape :

$$\mathtt{FACTOR}(\mathtt{X}^2 - 2)$$

On obtient :

$$(\mathtt{X} + \sqrt{2}).(\mathtt{X} - \sqrt{2})$$

On tape :

$$\mathtt{FACTOR}(\mathtt{X}^2 + 2 \cdot \mathtt{X} + 1)$$

On obtient :

$$(\mathtt{X} + 1)^2$$

On tape :

$$\mathtt{FACTOR}(\mathtt{X}^4 - 2.\mathtt{X}^2 + 1)$$

On obtient :

$$(\mathtt{X} - 1)^2.(\mathtt{X} + 1)^2$$

On tape :

$$\mathtt{FACTOR}(\mathtt{X}^3 - 2.\mathtt{X}^2 + 1)$$

On obtient :

$$\frac{(\mathtt{X} - 1).(2.\mathtt{X} + -1 + \sqrt{5}).(2.\mathtt{X} - (1 + \sqrt{5}))}{4}$$

4.10.4 GCD

`GCD` désigne le PGCD (plus grand commun diviseur) de deux polynômes.
On tape :

$$\mathtt{GCD}(\mathtt{X}^2 + 2 \cdot \mathtt{X} + 1, \mathtt{X}^2 - 1)$$

On obtient :

$$\mathtt{X} + 1$$

4.10.5 HERMITE

`HERMITE` a comme argument un entier n.
`HERMITE` renvoie le polynôme de HERMITE de degré n.
Il s'agit du polynôme :

$$H_n(x) = (-1)^n.e^{\frac{x^2}{2}} \frac{d^n}{dx^n} e^{\frac{-x^2}{2}}$$

On a :
pour $n \geq 0$

$$H_n^{''}(x) - xH_n^{'}(x) + nH_n(x) = 0$$

et pour $n \geq 1$

$$H_{n+1}(x) - xH_n(x) + nH_{n-1}(x) = 0$$

$$H_n^{'}(x) = nH_{n-1}(x)$$

On tape :

$$\mathtt{HERMITE}(6)$$

On obtient :

$$64.X^6 - 480.X^4 + 720.X^2 - 120$$

4.10.6 LCM

`LCM` désigne le PPCM (plus petit commun multiple) de deux polynômes.
On tape :

$$\mathtt{LCM}(\mathtt{X}^2 + 2 \cdot \mathtt{X} + 1, \mathtt{X}^2 - 1)$$

On obtient :

$$(\mathtt{X}^2 + 2 \cdot \mathtt{X} + 1) \cdot (\mathtt{X} - 1)$$

4.10.7 LEGENDRE

`LEGENDRE` a comme argument un entier n.
`LEGENDRE` renvoie le polynôme L_n non nul solution de l'équation différentielle :

$$(x^2-1).y''-2.x.y'-n(n+1).y=0$$

On a :
pour $n \geq 0$ la formule de Rodriguès

$$L_n(x)=\frac{1}{n!2^n}\frac{d^n}{dx^n}(x^2-1)^n$$

et pour $n \geq 1$

$$(n+1)L_{n+1}(x)=(2n+1)xL_n(x)-nL_{n-1}(x)$$

On tape :

`LEGENDRE(4)`

On obtient :

$$\frac{35.X^4-30.X^2+3}{8}$$

4.10.8 PARTFRAC

Décomposer en éléments simples la fraction rationnelle :

$$\frac{x^5-2\times x^3+1}{x^4-2\times x^3+2\times x^2-2\times x+1}$$

On utilise la commande `PARTFRAC`.
On tape :

$$\texttt{PARTFRAC}(\frac{X^5-2*X^3+1}{X^4-2*X^3+2*X^2-2*X+1})$$

On obtient en mode réel :

$$X+2+\frac{\frac{-1}{2}}{X-1}+\frac{\frac{X-3}{2}}{X^2+1}$$

On obtient en mode complexe :

$$X+2+\frac{\frac{1-3.i}{4}}{X+i}+\frac{\frac{-1}{2}}{X-1}+\frac{\frac{1+3.i}{4}}{X-i}$$

4.10.9 PROPFRAC

`PROPFRAC` a comme argument une fraction rationnelle.
`PROPFRAC` renvoie cette fraction rationnelle écrite de manière à mettre en évidence sa partie entière.
`PROPFRAC(A(X)/B(X))` écrit la fraction rationnelle $\frac{A[X]}{B[X]}$ sous la forme :

$$Q[X] + \frac{R[X]}{B[X]}$$

avec $R[X] = 0$ ou $0 \leq deg(R[X]) < deg(B[X])$.
On tape :

$$\texttt{PROPFRAC}(\frac{(5.X + 3).(X - 1)}{X + 2})$$

On obtient :

$$5.X - 12 + \frac{21}{X + 2}$$

4.10.10 PTAYL

Il s'agit d'écrire un polynôme $P[X]$ selon les puissances de $X - a$.
`PTAYL` a deux paramètres : un polynôme P et un nombre a.
On tape :

$$\texttt{PTAYL}(X^2 + 2 \cdot X + 1, 2)$$

On obtient le polynôme Q[X] :

$$X^2 + 6 \cdot X + 9$$

ATTENTION, on a :

$$P(X) = Q(X - 2)$$

4.10.11 QUOT

`QUOT` donne le quotient de deux polynômes dans la division selon les puissances décroissantes.
On tape :

$$\texttt{QUOT}(X^2 + 2 \cdot X + 1, X)$$

On obtient :

$$X + 2$$

4.10.12 REMAINDER

`REMAINDER` donne le reste de la division de deux polynômes (division selon les puissances décroissantes).
On tape :

$$\mathtt{REMAINDER}(\mathtt{X}^3 - 1, \mathtt{X}^2 - 1)$$

On obtient :

$$\mathtt{X} - 1$$

4.10.13 TCHEBYCHEFF

`TCHEBYCHEFF` a comme argument un entier n.
Si $n > 0$, `TCHEBYCHEFF` renvoie le polynôme T_n tel que :

$$T_n[x] = \cos(n.\arccos(x))$$

On a :
pour $n \geq 0$

$$T_n(x) = \sum_{k=0}^{[n/2]} C_n^{2k}(x^2-1)^k x^{n-2k}$$

pour $n \geq 0$

$$(1-x^2)T_n^{''}(x) - xT_n^{'}(x) + n^2 T_n(x) = 0$$

pour $n \geq 1$

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

Si $n < 0$ `TCHEBYCHEFF` renvoie le polynôme de Tchebycheff de seconde espèce :

$$T_n[x] = \frac{\sin(n.\arccos(x))}{\sin(\arccos(x))}$$

On tape :

$$\mathtt{TCHEBYCHEFF}(4)$$

On obtient :

$$8.\mathtt{X}^4 - 8.\mathtt{X}^2 + 1$$

en effet :
$\cos(4.x) = Re((\cos(x) + i.\sin(x))^4)$

$\cos(4.x) = \cos(x)^4 - 6.\cos(x)^2.(1 - \cos(x)^2) + ((1 - \cos(x)^2)^2$
$\cos(4.x) = T_4(\cos(x))$
On tape :

$$\mathtt{TCHEBYCHEFF}(-4)$$

On obtient :

$$\mathtt{8.X}^3 - \mathtt{4.X}$$

en effet :
$\sin(4.x) = \sin(x).(8.\cos(x)^3 - 4.\cos(x))$.

4.11 Les fonctions

Toutes les fonctions de ce paragraphe se trouve dans le menu `DIFF` du bandeau, sauf `DEF` qui se trouve dans le menu `ALGB`, et `IFTE` qui se trouve dans le menu `Tests` de la touche `MATH`.

4.11.1 DEF

`DEF` a comme argument une égalité entre le nom d'une fonction avec des parenthèses contenant le nom de la variable et une expression définissant la fonction.
`DEF` définit cette fonction et renvoie l'égalité.
On tape :

$$\mathtt{DEF(U(N)} = \mathtt{2}^{\mathtt{N}} + \mathtt{1)}$$

On obtient :

$$\mathtt{U(N)} = \mathtt{2}^{\mathtt{N}} + \mathtt{1}$$

Puis on tape :

$$\mathtt{U(3)}$$

On obtient :

$$\mathtt{9}$$

4.11.2 IFTE

`IFTE` a trois arguments, un boolèen (attention au $==$ pour le test !) et deux expressions *expr*1, *expr*2.
`IFTE` évalue le test, renvoie *expr*1 si le test est vrai, et renvoie *expr*2

si le test est faux.
On tape :

$$\mathtt{STORE(2,N)}$$
$$\mathtt{IFTE(N==0,1,\frac{N+1}{N})}$$

On obtient :

$$\frac{3}{2}$$

On peut bien sûr définir une fonction à l'aide de `IFTE` par exemple :

$$\mathtt{DEF(F(X)=IFTE(X==0,1,\frac{SIN(X)}{X}))}$$

définit la fonction f par :

$$f(x) = \begin{cases} 1 & \text{si } x = 0 \\ \dfrac{\sin(x)}{x} & \text{si } x \neq 0 \end{cases}$$

4.11.3 DERVX

Soit

$$f(x) = \frac{x}{x^2-1} + \ln(\frac{x+1}{x-1})$$

Calculer la dérivée de f.
On tape :

$$\mathtt{DERVX(\frac{X}{X^2-1}+LN(\frac{X+1}{X-1}))}$$

ou si on a stocké l'expression de $f(x)$ dans `F` c'est à dire si on a tapé :

$$\mathtt{STORE(\frac{X}{X^2-1}+LN(\frac{X+1}{X-1}),F)}$$
$$\mathtt{DERVX(F)}$$

ou si on a défini $F(X)$ à l'aide de `DEF` :

$$\mathtt{DEF(F(X)=\frac{X}{X^2-1}+LN(\frac{X+1}{X-1}))}$$
$$\mathtt{DERVX(F(X))}$$

On trouve une expression compliquée que l'on simplifie en faisant :
`ENTER`.
On obtient :

$$-\frac{3\cdot X^2-1}{X^4-2\cdot X^2+1}$$

4.11.4 DERIV

`DERIV` a deux arguments : une expression (ou une fonction) et une variable.
`DERIV` renvoie la dérivée de l' expression (ou de la fonction) par rapport à la variable donnée comme deuxième paramètre (utile pour calculer des dérivées partielles !).
Exemple :
Soit à calculer :

$$\frac{\partial(x.y^2.z^3 + x.y)}{\partial z}$$

On tape :

$$\texttt{DERIV(X.Y}^2\texttt{.Z}^3 + \texttt{X.Y , Z)}$$

On obtient :

$$\texttt{3.X.Y}^2\texttt{.Z}^2$$

4.11.5 TABVAR

`TABVAR` a comme paramètre une expression ayant une derivée rationnelle.
`TABVAR` renvoie (en mode pas à pas) le tableau de variations de l'expression, en fonction de la variable courante.
On tape :

$$\texttt{TABVAR(LN(X)} + \texttt{X)}$$

On obtient en mode pas à pas :
$\texttt{F} =: (\texttt{LN(X)} + \texttt{X})$
$\texttt{F}' =: (\frac{\texttt{1}}{\texttt{X}} + \texttt{1}$
$\rightarrow: \frac{\texttt{X+1}}{\texttt{X}}$
`Variation table :`

$$\begin{bmatrix} -\infty & ? & 0 & + & +\infty & X \\ ? & ? & -\infty & \uparrow & +\infty & F \end{bmatrix}$$

4.11.6 FOURIER

`FOURIER` a deux paramètres : une expression $f(x)$ et un entier n.
`FOURIER` renvoie le coefficient de Fourier c_n de $f(x)$ considérée comme une fonction définie sur $[0, T]$ et périodique de période T (T étant

égale au contenu de la variable `PERIOD`).
On a si f est continue par morceaux :

$$f(x) = \sum_{n=-\infty}^{+\infty} c_n e^{\frac{2inx\pi}{T}}$$

Exemple : Déterminer les coefficients de Fourier de la fonction f périodique de période $2.\pi$ et définie sur $[0\ 2.\pi[$ par $f(x) = x^2$.
On tape :

$$\texttt{STORE}(2.\pi\ ,\ \texttt{PERIOD})$$

$$\texttt{FOURIER}(\texttt{X}^2, \texttt{N})$$

On obtient après simplification :

$$\frac{2.\texttt{i}.\texttt{N}.\pi + 2}{\texttt{N}^2}$$

Donc si $n \neq 0$ on a :

$$c_n = \frac{2.i.N.\pi + 2}{N^2}$$

Puis on tape :

$$\texttt{FOURIER}(\texttt{X}^2, 0)$$

On obtient :

$$\frac{4.\pi^2}{3}$$

Donc si $n = 0$ on a :

$$c_0 = \frac{4.\pi^2}{3}$$

4.11.7 IBP

`IBP` a deux paramètres : une expression de la forme $u(x).v'(x)$ et $v(x)$.
`IBP` renvoie le `AND` de $u(x).v(x)$ et de $-v(x).u'(x)$, c'est à dire les termes que l'on doit calculer quand on fait une intégration par parties.
Il reste alors à calculer l'intégrale du deuxième terme du `AND`, puis à faire la somme avec le premier terme du `AND` pour obtenir une primitive de $u(x).v'(x)$.
On tape :

$$\texttt{IBP}(\texttt{LN}(\texttt{X}), \texttt{X})$$

On obtient :

$$\texttt{X.LN(X) AND } -1$$

On termine l'intégration en appelant `INTVX` :
`INTVX(X.LN(X) AND -1)`
on obtient alors :

$$\texttt{X} \cdot \texttt{LN(X)} - \texttt{X}$$

Remarque : Si le premier paramètre de `IBP` est le `AND` de deux éléments, `IBP` n'agit que sur le dernier élément de `AND` et ajoute le terme intégré au premier élément de `AND` (de façon à pouvoir faire plusieurs `IBP` à la suite).

4.11.8 INTVX

Exercice 1
Calculer une primitive de $\sin(x) \times \cos(x)$.
On tape :

$$\texttt{INTVX(SIN(X).COS(X))}$$

On obtient en pas à pas :
`COS(X). SIN(X)`
`Int[u'*F(u)] with u=SIN(X)`
puis `OK` et le résultat s'inscrit dans l'éditeur d'équations :

$$\frac{\texttt{SIN(X)}^2}{2}$$

Exercice 2
Soit

$$f(x) = \frac{x}{x^2-1} + \ln(\frac{x+1}{x-1})$$

Calculer une primitive de f.
On tape :

$$\texttt{INTVX}(\frac{\texttt{X}}{\texttt{X}^2-1} + \texttt{LN}(\frac{\texttt{X}+1}{\texttt{X}-1}))$$

ou si on a stocké l'expression de $f(x)$ dans `F`

$$\texttt{INTVX(F)}$$

ou si on a défini $F(X)$ à l'aide de `DEF` ($\texttt{DEF(F(X)} = \frac{\texttt{X}}{\texttt{X}^2-1} + \texttt{LN}(\frac{\texttt{X}+1}{\texttt{X}-1})$)

$$\texttt{INTVX(F(X))}$$

On trouve :

$$\mathtt{X}\cdot\mathtt{LN}(\frac{\mathtt{X}+1}{\mathtt{X}-1})+\frac{3}{2}\cdot\mathtt{LN}(|\mathtt{X}-1|)+\frac{3}{2}\cdot\mathtt{LN}(|\mathtt{X}+1|)$$

Exercice 3
Calculer :

$$\int\frac{2}{x^6+2\cdot x^4+x^2}\ dx$$

On tape :

$$\mathtt{INTVX}(\frac{2}{\mathtt{X}^6+2\cdot\mathtt{X}^4+\mathtt{X}^2})$$

On trouve :

$$-3\cdot\mathtt{ATAN}(\mathtt{X})-\frac{2}{\mathtt{X}}-\frac{\mathtt{X}}{\mathtt{X}^2+1}$$

Remarque :
On peut aussi taper :

$$\int_1^{\mathtt{X}}\frac{2}{\mathtt{X}^6+2\cdot\mathtt{X}^4+\mathtt{X}^2}\ \mathtt{dX}$$

qui donne le même résultat plus une constante égale à :

$$\frac{3.\pi+10}{4}$$

Exercice 4
Calculer :

$$\int\frac{1}{\sin(x)+\sin(2\cdot x)}\ dx$$

On tape :

$$\mathtt{INTVX}(\frac{1}{\mathtt{SIN}(\mathtt{X})+\mathtt{SIN}(2\cdot\mathtt{X})})$$

On trouve :

$$\frac{1}{6}\cdot\mathtt{LN}(|\mathtt{COS}(\mathtt{X})-1|)+\frac{1}{2}\cdot\mathtt{LN}(|\mathtt{COS}(\mathtt{X})+1|)+$$

$$\frac{-2}{3}\cdot\mathtt{LN}(|2\cdot\mathtt{COS}(\mathtt{X})+1|)$$

Remarque : si le paramètre de `INTVX` est le `AND` de deux éléments, `INTVX` n'agit que sur le deuxième élément du `AND`.

4.11.9 LIMIT

Trouver pour $n > 2$, la limite quand x tend vers 0 de :

$$\frac{n \times \tan(x) - \tan(n \times x)}{\sin(n \times x) - n \times \sin(x)}$$

On utilise la commande LIMIT.
On tape :

$$\mathtt{LIMIT}\left(\frac{\mathtt{N} \cdot \mathtt{TAN(X)} - \mathtt{TAN(N} \cdot \mathtt{X)}}{\mathtt{SIN(N} \cdot \mathtt{X)} - \mathtt{N} \cdot \mathtt{SIN(X)}}, 0\right)$$

On obtient :

$$2$$

Attention!!!
Pour trouver une limite quand x tend vers a^+ (resp a^-), le deuxième argument s'écrit :
$\mathtt{X} = \mathtt{A} + 0$ (resp $\mathtt{X} = \mathtt{A} - 0$), voir aussi 4.12.2.
Trouver la limite quand x tend vers $+\infty$ de :

$$\sqrt{x + \sqrt{x + \sqrt{x}}} - \sqrt{x}$$

On tape :

$$\mathtt{LIMIT}(\sqrt{\mathtt{X} + \sqrt{\mathtt{X} + \sqrt{\mathtt{X}}}} - \sqrt{\mathtt{X}}, +\infty)$$

On obtient au bout d'un moment :

$$\frac{1}{2}$$

Attention!!!
∞ peut s'obtenir grâce au raccourci clavier : SHIFT 0
$-\infty$ s'obtient alors en tapant :

$$(-)\ \infty$$

$+\infty$ s'obtient alors en tapant :

$$(-)\ (-)\ \infty$$

On trouve aussi ∞ dans le menu Constant de la touche MATH.

4.11.10 LIMIT et $\int$

Déterminer la limite quand a tend vers plus l'infini de :

$$\int_2^a \left(\frac{x}{x^2-1} + \ln\left(\frac{x+1}{x-1}\right)\right)\, dx$$

On tape dans l'éditeur d'équations :

$$\int_2^{+\infty} \left(\frac{\mathtt{X}}{\mathtt{X}^2-1} + \mathtt{LN}\left(\frac{\mathtt{X}+1}{\mathtt{X}-1}\right)\right)\, \mathtt{dX}$$

Attention, $+\infty$ s'obtient en tapant :

$$(-)\ (-)\ \infty\ (\mathtt{SHIFT}\ 0)$$

On obtient :

$$+\infty - \frac{7.\mathtt{LN}(3)}{2}$$

et après simplification :

$$+\infty$$

4.11.11 PREVAL

`PREVAL` a trois paramètres : une expression `F(VX)` dépendant de la variable contenue dans `VX` et deux expressions `A` et `B`.
`PREVAL(F(X,A,B)` renvoie `F(B)-F(A)`.
`PREVAL` est utile pour calculer une intégrale définie à partir d'une primitive : on évalue cette primitive entre les deux bornes de l'intégrale.
On tape :

$$\mathtt{PREVAL}(\mathtt{X}^2 + \mathtt{X}, 2, 3)$$

On obtient :

$$12 - 6$$

4.11.12 RISCH

`RISCH` a deux paramètres : une expression et un nom de variable. `RISCH` renvoie une primitive du premier paramètre par rapport à la variable spécifiée en deuxième paramètre.
On tape :

$$\mathtt{RISCH}((2.\mathtt{X}^2 + 1).\mathtt{EXP}(\mathtt{X}^2 + 1), \mathtt{X})$$

On obtient :

$$\mathtt{X.EXP(X^2+1)}$$

Remarque : si le paramètre de `RISCH` est le `AND` de deux éléments, `RISCH` n'agit que sur le deuxième élément du `AND`.

4.12 Développements limités et asymptotiques

Toutes les fonctions de ce paragraphe se trouve dans le menu `DIFF` du bandeau.
Il est d'usage d'écrire les développements selon les puissances croissantes de la variable, on fera donc le choix $\mathtt{1+x+x^2}$... dans `CFG`.

4.12.1 DIVPC

`DIVPC` a trois arguments : deux polynômes `A(X)`, `B(X)` (avec $\mathtt{B(0)} \neq \mathtt{0}$) et un entier `n`.
`DIVPC` renvoie le quotient `Q(X)` de la division de `A(X)` par `B(X)` selon les puissances croissantes avec $\deg(\mathtt{Q}) \leq \mathtt{n}$ ou $\mathtt{Q} = \mathtt{0}$.
$Q[X]$ est donc le développement limité d'ordre n de $\frac{A[X]}{B[X]}$ au voisinage de $X = 0$.
On tape :

$$\mathtt{DIVPC(1+X^2+X^3, 1+X^2, 5)}$$

On obtient :

$$\mathtt{1+X^3-X^5}$$

ATTENTION : la machine demande à passer en “puissances croissantes”, répondre `yes`.

4.12.2 LIMIT

`LIMIT` a comme arguments une expression dépendant d'une variable et une égalité (variable = la valeur où l'on veut calculer la limite).
Le nom de la variable peut être omis quand il s'agit de la variable courante (celle dont le nom se trouve dans `VX`).
Il est souvent préférable d'écrire l'expression quotée :
`QUOTE(expression)`, pour éviter une réécriture de cette expression sous forme normale (pour ne pas avoir une simplification rationnelle

des arguments) avant l'exécution de la commande LIMIT.
On tape par exemple :

$$\mathtt{LIMIT(QUOTE((2X-1)\cdot EXP(\frac{1}{X-1})),X=+\infty)}$$

On obtient :

$$+\infty$$

Pour calculer une limite à droite (resp une limite à gauche) on tape par exemple :

$$\mathtt{LIMIT(\frac{1}{X-1},QUOTE(1+0))}$$

On obtient si X est la variable courante :

$$+\infty$$

$$\mathtt{LIMIT(\frac{1}{X-1},QUOTE(1-0))}$$

On obtient si X est la variable courante :

$$-\infty$$

Il n'est pas nècessaire de quoter le deuxième argument quand il est écrit avec le signe = par exemple :

$$\mathtt{LIMIT(\frac{1}{X-1},X=1+0)}$$

On obtient :

$$+\infty$$

4.12.3 SERIES

- développement au voisinage de x=a
  Exemple :
  Donner un développement limité à l'ordre 4 au voisinage de $x=\frac{\pi}{6}$ de $\cos(2\times x)^2$.
  On utilise la commande SERIES.
  On tape :

  $$\mathtt{SERIES(COS(2\cdot X)^2,X=\frac{\pi}{6},4)}$$

  On obtient :

  $$\left(\frac{1}{4}-\sqrt{3}h+2h^2+\frac{8\sqrt{3}}{3}h^3-\frac{8}{3}h^4\right)\Big|_{h=X-\frac{\pi}{6}}$$

- développement au voisinage de x=+∞ ou x=-∞

  Exemple 1 :
  Donner un développement de $\arctan(x)$ à l'ordre 5 au voisinage de x=+∞ en prenant comme infiniment petit $h = \frac{1}{x}$.
  On tape :

  $$\texttt{SERIES(ATAN(X)}, \texttt{X} = +\infty, 5)$$

  On obtient :

  $$\left(\frac{\pi}{2} - \texttt{h} + \frac{\texttt{h}^3}{3} - \frac{\texttt{h}^5}{5}\right)\Big|_{\texttt{h} = \frac{1}{\texttt{X}}}$$

  Exemple 2 :
  Donner un développement de $(2x-1)e^{\frac{1}{x-1}}$ à l'ordre 2 au voisinage de x=+∞ en prenant comme infiniment petit $h = \frac{1}{x}$.
  On tape :

  $$\texttt{SERIES}((2\texttt{X} - 1) \cdot \texttt{EXP}(\frac{1}{\texttt{X} - 1}), \texttt{X} = +\infty, 3)$$

  On obtient :

  $$\left(\frac{2 + \texttt{h} + 2\texttt{h}^2 + \frac{17\texttt{h}^3}{6}}{\texttt{h}}\right)\Big|_{\texttt{h} = \frac{1}{\texttt{X}}}$$

  Exemple 3 :
  Donner un développement de $(2x-1)e^{\frac{1}{x-1}})$ à l'ordre 2 au voisinage de x=-∞ en prenant comme infiniment petit $h = -\frac{1}{x}$.
  On tape :

  $$\texttt{SERIES}((2\texttt{X} - 1) \cdot \texttt{EXP}(\frac{1}{\texttt{X} - 1}), \texttt{X} = -\infty, 3)$$

  On obtient :

  $$\left(\frac{-2 + \texttt{h} - 2\texttt{h}^2 + \frac{17\texttt{h}^3}{6}}{\texttt{h}}\right)\Big|_{\texttt{h} = -\frac{1}{\texttt{X}}}$$

- développement unidirectionnel

  Il faut utiliser pour l'ordre un réel positif (par exemple 4.) pour faire un développement au voisinage de $x = a$ avec $x > a$ et un réel négatif (par exemple -4.) pour faire un développement au voisinage de $x = a$ avec $x < a$.
  Exemple 1 :

Donner un développement de $\frac{(1+X)^{\frac{1}{X}}}{X^3}$ à l'ordre 2, au voisinage de $X = 0^+$.
On tape :

$$\mathtt{SERIES}(\frac{(1+\mathtt{X})^{\frac{1}{\mathtt{X}}}}{\mathtt{X}^3}, \mathtt{X}, 2.)$$

On obtient :

$$(-\frac{-2.\mathsf{e} + \mathsf{e.h}}{2.\mathsf{h}^3})|\mathtt{h} = \mathtt{X}$$

Exemple 2 :
Donner un développement de $\frac{(1+X)^{\frac{1}{X}}}{X^3}$ à l'ordre 2, au voisinage de $X = 0^-$.
On tape :

$$\mathtt{SERIES}(\frac{(1+\mathtt{X})^{\frac{1}{\mathtt{X}}}}{\mathtt{X}^3}, \mathtt{X}, -2.)$$

On obtient :

$$(-\frac{-2.\mathsf{e} + \mathsf{e.h}}{2.\mathsf{h}^3})|\mathtt{h} = \mathtt{X}$$

Exemple 3 :
Donner un développement de $\frac{(1+X)^{\frac{1}{X}}}{X^3}$ à l'ordre 2, au voisinage de $X = 0$.
On tape :

$$\mathtt{SERIES}(\frac{(1+\mathtt{X})^{\frac{1}{\mathtt{X}}}}{\mathtt{X}^3}, \mathtt{X}, 2)$$

On obtient :

$$(-\frac{-2.\mathsf{e} + \mathsf{e.h}}{2.\mathsf{h}^3})|\mathtt{h} = \mathtt{X}$$

4.12.4 `TAYLOR0`

`TAYLOR0` a un seul argument : la fonction de x à développer, et renvoie son développement limité à l'ordre relatif 4 au voisinage de $x = 0$ (si x est la variable courante).
On tape :

$$\mathtt{TAYLOR0}(\frac{\mathtt{TAN(P.X)} - \mathtt{SIN(P.X)}}{\mathtt{TAN(Q.X)} - \mathtt{SIN(Q.X)}})$$

On obtient :

$$\frac{\mathtt{P}^3}{\mathtt{Q}^3} + \frac{\mathtt{P}^5 - \mathtt{Q}^2.\mathtt{P}^3}{4.\mathtt{Q}^3}.\mathtt{X}^2$$

ATTENTION : l'ordre 4 veut dire que l'on développe à l'ordre relatif 4 le numérateur et le dénominateur (ici ordre absolu 5 pour le numérateur et le dénominateur, ce qui donne en fin de compte, un ordre 2 (5-3) puisque la valuation du dénominateur est égale à 3).

4.12.5 TRUNC

TRUNC permet de tronquer un polynôme à un ordre donné (utile quand on fait des développements limités).
TRUNC a deux arguments : un polynôme et X^n.
TRUNC renvoie le polynôme tronqué à l'ordre $n-1$: on n'a pas de termes de degré $\geq n$.
On tape :

$$\mathtt{TRUNC}((1+\mathtt{X}+\frac{1}{2}.\mathtt{X}^2)^3\ ,\ \mathtt{X}^4)$$

On obtient :

$$1+3.\mathtt{X}+\frac{9}{2}.\mathtt{X}^2+4.\mathtt{X}^3$$

4.13 Les Fonctions de réécriture

Toutes les fonctions de ce paragraphe se trouvent dans le menu REWRITE du bandeau.

4.13.1 DISTRIB

DISTRIB permet d'appliquer la distributivité de la multiplication par rapport à l'addition une fois.
DISTRIB permet, quand on l'applique plusieurs fois, d'effectuer la distributivité pas à pas.
On tape :

$$\mathtt{DISTRIB}((\mathtt{X}+1)\cdot(\mathtt{X}+2)\cdot(\mathtt{X}+3))$$

On obtient :

$$\mathtt{X}\cdot(\mathtt{X}+2)\cdot(\mathtt{X}+3)+1\cdot(\mathtt{X}+2)\cdot(\mathtt{X}+3)$$

4.13.2 EPSX0

EPSX0 a comme paramètre une expression de X et renvoie l'expression où les valeurs plus petites que EPS ont été remplacées par zéro.

On tape :

$$\mathtt{EPSX0}(0.001+\mathtt{X})$$

On obtient (avec EPS=0.01) :

$$0+\mathtt{X}$$

On obtient (avec EPS=0.0001) :

$$.001+\mathtt{X}$$

4.13.3 EXP2POW

EXP2POW permet de transformer une expression de la forme :
$\exp(n \times \ln(x))$ en une puissance de x.
On tape :

$$\mathtt{EXP2POW(EXP(N \cdot LN(X)))}$$

On obtient :

$$\mathtt{X^N}$$

Bien voir la différence avec LNCOLLECT :
on a :
$\mathtt{LNCOLLECT(EXP(N \cdot LN(X))) = EXP(N \cdot LN(X))}$
$\mathtt{LNCOLLECT(EXP(LN(X)/3)) \ = \ EXP(LN(X)/3)}$
$\mathtt{EXP2POW(EXP(LN(X)/3)) \ = \ X^{\frac{1}{3}}}$

4.13.4 EXPLN

EXPLN a comme argument une expression trigonométrique.
EXPLN transforme les fonctions trigonométriques en exponentielles et logarithmes SANS linéariser.
EXPLN fait passer en mode complexe.
On tape :

$$\mathtt{EXPLN(SIN(X))}$$

On obtient :

$$\frac{\mathtt{EXP(i.X)} - \frac{1}{\mathtt{EXP(i.X)}}}{\mathtt{2.i}}$$

4.13.5 FDISTRIB

FDISTRIB permet d'effectuer la distributivité de la multiplication par rapport à l'addition en une seule fois.
On tape :

$$\mathtt{FDISTRIB}((\mathtt{X}+1)\cdot(\mathtt{X}+2)\cdot(\mathtt{X}+3))$$

On obtient :

$$\mathtt{X}^3+6\cdot\mathtt{X}^2+11\cdot\mathtt{X}+6$$

4.13.6 LIN

LIN a comme argument une expression contenant des exponentielles et des fonctions trigonométriques.
LIN linéarise cette expression (l'exprime en fonction de $\exp(n.x)$).
LIN fait passer en `mode complexe` quand il y a des fonctions trigonométriques.
Exemple 1 :
On tape :

$$\mathtt{LIN}((\mathtt{SIN}(\mathtt{X}))$$

On obtient :

$$-(\frac{\mathtt{i}}{2}.\mathtt{EXP}(\mathtt{i}.\mathtt{X}))+\frac{\mathtt{i}}{2}.\mathtt{EXP}(-(\mathtt{i}.\mathtt{X}))$$

Exemple 2 :
On tape :

$$\mathtt{LIN}((\mathtt{COS}(\mathtt{X})^2)$$

On obtient :

$$-(\frac{1}{4}.\mathtt{EXP}(2.\mathtt{i}.\mathtt{X}))+\frac{1}{2}+\frac{1}{4}.\mathtt{EXP}(-(2.\mathtt{i}.\mathtt{X}))$$

Exemple 3 :
On tape :

$$\mathtt{LIN}((\mathtt{EXP}(\mathtt{X})+1)^3)$$

On obtient :

$$3.\mathtt{EXP}(\mathtt{X})+1+3.\mathtt{EXP}(2.\mathtt{X})+\mathtt{EXP3}.\mathtt{X})$$

4.13.7 LNCOLLECT

LNCOLLECT a comme argument une expression contenant des logarithmes.
LNCOLLECT regroupe les termes en logarithmes. Il est donc préférable de l'utiliser sur une expression factorisée (en utilisant FACTOR).
On tape :

$$\mathtt{LNCOLLECT(LN(X+1)+LN(X-1))}$$

On obtient :

$$\mathtt{LN((X+1)(X-1))}$$

4.13.8 POWEXPAND

POWEXPAND écrit une puissance sous la forme d'un produit.
On tape :

$$\mathtt{POWEXPAND((X+1)^3)}$$

On obtient :

$$\mathtt{(X+1)\cdot(X+1)\cdot(X+1)}$$

Cela permet ainsi de faire le développement de $(x+1)^3$ en pas à pas actif, en appliquant DISTRIB plusieurs fois au résultat précédent.

4.13.9 SIMPLIFY

SIMPLIFY simplifie l'expression de façon automatique.
Comme toute simplification automatique, il ne faut pas s'attendre à des miracles et pourtant...
On tape :

$$\mathtt{SIMPLIFY(\frac{SIN(3.X)+SIN(7.X)}{SIN(5.X)})}$$

On obtient après simplification :

$$\mathtt{4.COS(X)^2-2}$$

4.13.10 XNUM

XNUM a comme paramètre une expression.
XNUM fait passer en mode approximatif et renvoie la valeur numérique de l'expression.
On tape :

$$\mathtt{XNUM}(\sqrt{2})$$

On obtient :

$$1.41421356237$$

4.13.11 XQ

XQ a comme paramètre une expression numérique réelle.
XQ fait passer en mode exact et donne une approximation rationnelle ou réelle de l'expression.
On tape :

$$\mathtt{XQ}(1.41422)$$

On obtient :

$$\frac{66441}{46981}$$

On tape :

$$\mathtt{XQ}(1.414213562)$$

On obtient :

$$\sqrt{2}$$

4.14 Équations

Toutes les fonctions de ce paragraphe se trouve dans le menu SOLV du bandeau.

4.14.1 ISOLATE

ISOLATE isole une variable dans une expression ou une équation.
ISOLATE a deux paramètres une expression ou une équation et le nom de la variable à isoler.
ATTENTION : ISOLATE ne renvoie qu'une solution.
On tape :

$$\mathtt{ISOLATE}(\mathtt{X}^4 - 1 = 3, \mathtt{X})$$

On obtient :

$$(\mathtt{X} = \sqrt{2})$$

4.14.2 SOLVEVX

`SOLVEVX` a comme paramètre une équation entre deux expressions de la variable contenue dans `VX` ou une expression (`=0` est alors sous-entendu).
`SOLVEVX` résout l'équation dans $\mathbb{R}$ en mode réel et dans $\mathbb{C}$ en mode complexe (sans tenir compte de `REALASSUME`).
Exemple 1 :
On tape :

$$\mathtt{SOLVEVX}(\mathtt{X}^4 - 1 = 3)$$

On obtient en mode réel :

$$(\mathtt{X} = -\sqrt{2})\ \mathtt{OR}\ (\mathtt{X} = \sqrt{2})$$

On obtient en mode complexe :

$$(\mathtt{X} = -\sqrt{2})\ \mathtt{OR}\ (\mathtt{X} = \sqrt{2})\ \mathtt{OR}\ (\mathtt{X} = -\mathtt{i}.\sqrt{2})\ \mathtt{OR}\ (\mathtt{X} = \mathtt{i}\sqrt{2})$$

Exemple 2 :
On tape :

$$\mathtt{SOLVEVX}((\mathtt{X} - 2).\mathtt{SIN}(\mathtt{X}))$$

On obtient en mode réel :

$$(\mathtt{X} = -2.\pi.\mathtt{n}_1)\ \mathtt{OR}\ (\mathtt{X} = 2.\pi.\mathtt{n}_1)\ \mathtt{OR}\ (\mathtt{X} = 2)$$

4.14.3 SOLVE

`SOLVE` a comme arguments une équation entre deux expressions ou une expression (`=0` est alors sous-entendu), et le nom d'une variable.
`SOLVE` résout l'équation dans $\mathbb{R}$ en mode réel et dans $\mathbb{C}$ en mode complexe (sans tenir compte de `REALASSUME`).
On tape :

$$\mathtt{SOLVE}(\mathtt{X}^4 - 1 = 3, \mathtt{X})$$

On obtient en mode réel :

$$(\mathtt{X} = -\sqrt{2})\ \mathtt{OR}\ (\mathtt{X} = \sqrt{2})$$

On obtient en mode complexe :

$$(\mathtt{X} = -\sqrt{2})\ \mathtt{OR}\ (\mathtt{X} = \sqrt{2})\ \mathtt{OR}\ (\mathtt{X} = -\mathtt{i}.\sqrt{2})\ \mathtt{OR}\ (\mathtt{X} = \mathtt{i}\sqrt{2})$$

4.15 Les systèmes linéaires

Toutes les fonctions de ce paragraphe se trouve dans le menu `SOLV` du bandeau.

4.15.1 `LINSOLVE`

`LINSOLVE` permet de résoudre un système d'équations linéaires.
On suppose les différentes équations écrites sous la forme :
expression $= 0$.
`LINSOLVE` a deux arguments :
les premiers membres des différentes équations séparés par `AND` et les noms des différentes variables séparés par `AND`.
Exemple 1 :
On tape :

$$\mathtt{LINSOLVE}(\mathtt{X} + \mathtt{Y} + 3\ \mathtt{AND}\ \mathtt{X} - \mathtt{Y} + 1, \mathtt{X}\ \mathtt{AND}\ \mathtt{Y})$$

On obtient :

$$(\mathtt{X} = -2)\ \mathtt{AND}\ (\mathtt{Y} = -1)$$

On obtient si on est en mode pas à pas (`CFG` etc...) :
`L2=L2-L1`

$$\begin{bmatrix} 1 & 1 & 3 \\ 1 & -1 & 1 \end{bmatrix}$$

`ENTER`
`L1=2L1-L2`

$$\begin{bmatrix} 1 & 1 & 3 \\ 0 & -2 & -2 \end{bmatrix}$$

`ENTER`
`Reduction Result`

$$\begin{bmatrix} 2 & 0 & 4 \\ 0 & -2 & -2 \end{bmatrix}$$

`ENTER`
Il s'écrit alors dans l'éditeur :

$$(\mathtt{X} = -2)\ \mathtt{AND}\ (\mathtt{Y} = -1)$$

Exemple 2 :
On tape :

$$(2 \cdot \mathtt{X} + \mathtt{Y} + \mathtt{Z} = 1)\ \mathtt{AND}\ (\mathtt{X} + \mathtt{Y} + 2 \cdot \mathtt{Z} = 1)\ \mathtt{AND}\ (\mathtt{X} + 2 \cdot \mathtt{Y} + \mathtt{Z} = 4)$$

puis on appelle `LINSOLVE`
puis, on tape les inconnues :

$$\mathtt{X\ AND\ Y\ AND\ Z}$$

et `ENTER`
On obtient si on est en mode pas à pas (`CFG` etc...) :
`L2=2L2-L1`

$$\begin{bmatrix} 2 & 1 & 1 & -1 \\ 1 & 1 & 2 & -1 \\ 1 & 2 & 1 & -4 \end{bmatrix}$$

puis `ok`

`L3=2L3-L1`

$$\begin{bmatrix} 2 & 1 & 1 & -1 \\ 0 & 1 & 3 & -1 \\ 1 & 2 & 1 & -4 \end{bmatrix}$$

etc...à la fin
`Reduction Result`

$$\begin{bmatrix} 8 & 0 & 0 & 4 \\ 0 & 8 & 0 & -20 \\ 0 & 0 & 8 & -4 \end{bmatrix}$$

puis `ENTER`

$$(\mathtt{X} = -\frac{1}{2})\ \mathtt{AND}\ (\mathtt{Y} = \frac{5}{2})\ \mathtt{AND}\ (\mathtt{Z} = -\frac{1}{2})$$

s'écrit dans l'éditeur.

4.16 Les équations différentielles

Toutes les fonctions de ce paragraphe se trouve dans le menu `SOLV` du bandeau.

4.16.1 DESOLVE et SUBST

`DESOLVE` permet de résoudre d'autres équations différentielles.
Les paramètres sont : l'équation différentielle (où $y\prime$ s'écrit `d1Y(X)`) et l'inconnue `Y(X)`.
Exemple 1 :
Résoudre :

$$y'' + y = \cos(x) \;\; y(0) = c_0 \;\; y'(0) = c_1$$

On tape :

$$\mathtt{DESOLVE(d1d1Y(X) + Y(X) = COS(X), Y(X))}$$

On trouve :

$$\mathtt{Y(X)} = \mathtt{cC0} \cdot \mathtt{COS(X)} + \frac{\mathtt{X} + 2 \cdot \mathtt{cC1}}{2} \cdot \mathtt{SIN(X)}$$

`cC0` et `cC1` sont les constantes d'intégration ($\mathtt{y(0) = cC0 \;\; y\prime(0) = cC1}$).
On peut ensuite donner une valeur aux constantes en utilisant la commande `SUBST`. On écrit, si veut les solutions vérifiant $y(0) = 1$:

$$\mathtt{SUBST(Y(X)} = \mathtt{cC0} \cdot \mathtt{COS(X)} + \frac{\mathtt{X} + 2 \cdot \mathtt{cC1}}{2} \cdot \mathtt{SIN(X), cC0 = 1)}$$

On obtient :

$$\mathtt{Y(X)} = \frac{2 \cdot \mathtt{COS(X)} + (\mathtt{X} + 2 \cdot \mathtt{cC1}) \cdot \mathtt{SIN(X)}}{2}$$

Exemple 2 :
Résoudre :

$$y'' + y = \cos(x) \;\; y(0) = 1 \;\; y'(0) = c_1$$

Pour avoir les solutions vérifiant $y(0) = 1$ on peut aussi taper directement :

$$\mathtt{DESOLVE((d1d1Y(X) + Y(X) = COS(X))AND(Y(0) = 1), Y(X))}$$

On trouve alors :

$$\mathtt{Y(X)} = \mathtt{COS(X)} + \frac{\mathtt{X} + 2 \cdot \mathtt{cC1}}{2} \cdot \mathtt{SIN(X)}$$

4.16.2 LDEC

`LDEC` permet de résoudre directement les équations linéaires à coefficients constants.
Les paramètres sont le second membre et l'équation caractéristique.
Résoudre :

$$y'' - 6.y' + 9.y \ = \ x.e^{3.x}$$

On tape :

$$\mathtt{LDEC(X \cdot EXP(3 \cdot X), X^2 - 6 \cdot X + 9)}$$

On trouve :

$$(\frac{\mathtt{X}^3}{6} - (3 \cdot \mathtt{cC0} - \mathtt{cC1}) \cdot \mathtt{X} + \mathtt{cC0}) \cdot \mathtt{EXP}(3 \cdot \mathtt{X})$$

`cC0` et `cC1` sont les constantes d'intégration ($\mathtt{y}(0) = \mathtt{cC0}$ $\mathtt{y}'(0) = \mathtt{cC1}$).

4.17 Les expressions trigonométriques

Toutes les fonctions de ce paragraphe se trouve dans le menu `TRIG` du bandeau.

4.17.1 ACOS2S

`ACOS2S` a comme argument une expression trigonométrique.
`ACOS2S` transforme cette expression en remplaçant :
$\arccos(x)$ par $\frac{\pi}{2} - \arcsin(x)$.
On tape :

$$\mathtt{ACOS2S(ACOS(X) + ASIN(X))}$$

On obtient :

$$\frac{\pi}{2}$$

4.17.2 ASIN2C

`ASIN2C` a comme argument une expression trigonométrique.
`ASIN2C` transforme cette expression en remplaçant :
$\arcsin(x)$ par $\frac{\pi}{2} - \arccos(x)$.

On tape :

$$\texttt{ASIN2C(ACOS(X)} + \texttt{ASIN(X))}$$

On obtient :

$$\frac{\pi}{2}$$

4.17.3 ASIN2T

`ASIN2T` a comme argument une expression trigonométrique.
`ASIN2T` transforme cette expression en remplaçant :
$\arcsin(x)$ par $\arctan(\frac{x}{\sqrt{1-x^2}})$.
On tape :

$$\texttt{ASIN2T(ASIN(X))}$$

On obtient :

$$\texttt{ATAN}(\frac{\texttt{X}}{\sqrt{1-\texttt{X}^2}})$$

4.17.4 ATAN2S

`ATAN2S` a comme argument une expression trigonométrique.
`ATAN2S` transforme cette expression en remplaçant :
$\arctan(x)$ par $\arcsin(\frac{x}{\sqrt{1+x^2}})$.
On tape :

$$\texttt{ATAN2S(ATAN(X))}$$

On obtient :

$$\texttt{ASIN}(\frac{\texttt{X}}{\sqrt{\texttt{X}^2+1}})$$

4.17.5 HALFTAN

`HALFTAN` a comme argument une expression trigonométrique.
`HALFTAN` transforme les $\sin(x)\cos(x)$ et $\tan(x)$ contenus dans l'expression en fonction de $\tan(\frac{x}{2})$.

On tape :

$$\mathtt{HALFTAN}(\frac{\mathtt{SIN}(2.\mathtt{X})}{1+\mathtt{COS}(2.\mathtt{X})})$$

On obtient après simplification :

$$\mathtt{TAN}(\mathtt{X})$$

On tape :

$$\mathtt{HALFTAN}(\mathtt{SIN}(\mathtt{X})^2+\mathtt{COS}(\mathtt{X})^2)$$

On obtient ($\mathtt{SQ}(\mathtt{X})=\mathtt{X}^2$) :

$$\left(\frac{2.\mathtt{TAN}(\frac{\mathtt{X}}{2})}{\mathtt{SQ}(\mathtt{TAN}(\frac{\mathtt{X}}{2}))+1}\right)^2+\left(\frac{1-\mathtt{SQ}(\mathtt{TAN}(\frac{\mathtt{X}}{2}))}{\mathtt{SQ}(\mathtt{TAN}(\frac{\mathtt{X}}{2}))+1}\right)^2$$

On obtient après simplification :

$$1$$

4.17.6 SINCOS

SINCOS a comme argument une expression contenant des exponentielles complexes.
SINCOS transforme cette expression en fonction de $\sin(x)$ et de $\cos(x)$.
On tape :

$$\mathtt{SINCOS}(\mathtt{EXP}(\mathtt{i}.\mathtt{X}))$$

On obtient :

$$\mathtt{COS}(\mathtt{X})+\mathtt{i}.\mathtt{SIN}(\mathtt{X})$$

4.17.7 TAN2CS2

TAN2CS2 a comme argument une expression trigonométrique.
TAN2CS2 transforme cette expression en remplaçant :
$\tan(x)$ par $\dfrac{1-\cos(2.x)}{\sin(2.x)}$
On tape :

$$\mathtt{TAN2CS2}(\mathtt{TAN}(\mathtt{X}))$$

On obtient :

$$\frac{\texttt{1} - \texttt{COS}(2.\texttt{X})}{\texttt{SIN}(2.\texttt{X})}$$

4.17.8 TAN2SC

`TAN2SC` a comme argument une expression trigonométrique.
`TAN2SC` transforme cette expression en remplaçant :
$\tan(x)$ par $\dfrac{\sin(x)}{\cos(x)}$.
On tape :

$$\texttt{TAN2SC(TAN(X))}$$

On obtient :

$$\frac{\texttt{SIN(X)}}{\texttt{COS(X)}}$$

4.17.9 TAN2SC2

`TAN2SC2` a comme argument une expression trigonométrique.
`TAN2SC2` transforme cette expression en remplaçant :
$\tan(x)$ par $\dfrac{\sin(2.x)}{1+\cos(2.x)}$.
On tape :

$$\texttt{TAN2SC2(TAN(X))}$$

On obtient :

$$\frac{\texttt{SIN}(2.\texttt{X})}{\texttt{1} + \texttt{COS}(2.\texttt{X})}$$

4.17.10 TCOLLECT

`TCOLLECT` a comme argument une expression trigonométrique.
`TCOLLECT` linéarise cette expression en fonction de $\sin(n.x)$ et $\cos(n.x)$ puis rassemble, `en mode réel`, les sinus et les cosinus de même angle.
On tape :

$$\texttt{TCOLLECT(SIN(X)} + \texttt{COS(X))}$$

On obtient :

$$\sqrt{2}.\mathtt{COS}(\mathtt{X} - \frac{\pi}{4})$$

4.17.11 TEXPAND

TEXPAND a comme argument une expression trigonométrique.
TEXPAND développe cette expression en fonction de $\sin(x)$ et $\cos(x)$.
Exemple 1 :
On tape :

$$\mathtt{TEXPAND(COS(X + Y))}$$

On obtient :

$$\mathtt{COS(Y).COS(X) - SIN(Y).SIN(X)}$$

Exemple 2 :
On tape :

$$\mathtt{TEXPAND(COS(3.X))}$$

On obtient :

$$\mathtt{4.COS(X)}^3 - \mathtt{3.COS(X)}$$

Exemple 3 :
On tape :

$$\mathtt{TEXPAND}(\frac{\mathtt{SIN(3.X) + SIN(7.X)}}{\mathtt{SIN(5.X)}})$$

On obtient après une simplification (ENTER) :

$$\mathtt{4.COS(X)}^2 - 2$$

4.17.12 TLIN

TLIN a comme argument une expression trigonométrique.
TLIN linéarise cette expression en fonction de $\sin(n.x)$ et $\cos(n.x)$.
Exemple :
On tape :

$$\mathtt{TLIN(COS(X).COS(Y))}$$

On obtient :

$$\frac{1}{2}.\mathtt{COS}(\mathtt{X}-\mathtt{Y})+\frac{1}{2}.\mathtt{COS}(\mathtt{X}+\mathtt{Y})$$

Exemple 2 :
On tape :

$$\mathtt{TLIN}(\mathtt{COS}(\mathtt{X})^3)$$

On obtient :

$$\frac{1}{4}.\mathtt{COS}(3.\mathtt{X})+\frac{3}{4}.\mathtt{COS}(\mathtt{X})$$

Exemple 3 :
On tape :

$$\mathtt{TLIN}(4.\mathtt{COS}(\mathtt{X})^2-2)$$

On obtient :

$$2.\mathtt{COS}(2.\mathtt{X})$$

4.17.13 `TRIG`

`TRIG` a comme argument une expression trigonométrique.
`TRIG` simplifie cette expression à l'aide de $\sin(x)^2+\cos(x)^2=1$.
On tape :

$$\mathtt{TRIG}(\mathtt{SIN}(\mathtt{X})^2+\mathtt{COS}(\mathtt{X})^2+1)$$

On obtient :

$$2$$

4.17.14 `TRIGCOS`

`TRIGCOS` a comme argument une expression trigonométrique.
`TRIGCOS` simplifie cette expression, en privilégiant les cosinus, à l'aide de $\sin(x)^2+\cos(x)^2=1$.
On tape :

$$\mathtt{TRIGCOS}(\mathtt{SIN}(\mathtt{X})^4+\mathtt{COS}(\mathtt{X})^2+1)$$

On obtient :

$$\texttt{COS(X)}^4 - \texttt{COS(X)}^2 + 2$$

4.17.15 TRIGSIN

TRIGSIN a comme argument une expression trigonométrique.
TRIGSIN simplifie cette expression, en privilégiant les sinus, à l'aide de $\sin(x)^2 + \cos(x)^2 = 1$.
On tape :

$$\texttt{TRIGSIN(SIN(X)}^4 + \texttt{COS(X)}^2 + 1)$$

On obtient :

$$\texttt{SIN(X)}^4 - \texttt{SIN(X)}^2 + 2$$

4.17.16 TRIGTAN

TRIGTAN a comme argument une expression trigonométrique.
TRIGTAN simplifie cette expression, en privilégiant les tangentes, à l'aide de $\sin(x)^2 + \cos(x)^2 = 1$.
On tape :

$$\texttt{TRIGTAN(SIN(X)}^4 + \texttt{COS(X)}^2 + 1)$$

On obtient :

$$\frac{2.\texttt{TAN(X)}^4 + 3.\texttt{TAN(X)}^2 + 2}{\texttt{TAN(X)}^4 + 2.\texttt{TAN(X)}^2 + 1}$$

Chapitre 5

Exercices traités avec la HP40

5.1 Introduction

Commencez par sélectionner le `CAS` :
pour cela appuyer sur `F6` pour `CAS` du bandeau.
Les différentes commandes utilisées dans ce chapitre se trouvent :
- dans les menus de l'éditeur d'équations :
`ALGB(CFG DEF FACTOR SUBST TEXPAND)`
`DIFF (DERIVX DERIV INTVX INT LIMIT TABVAR)`
`REWRITE (DISTRIB LIN POWEXPAND XNUM)`
`SOLV (LINSOLV)`
- et dans le menu de la touche `MATH` :
`Complex (DROITE RE IM)` ,
`Integer (IEGCD ISPRIME? PROPFRAC)`.
Puis mettre la calculatrice en mode algébrique réel exact :
pour cela appuyer sur `ALG` du bandeau et mettre en surbrillance `CFG`, puis `OK` du bandeau.
Il suffit alors de choisir `Default cfg` puis `OK` du bandeau, vous pouvez aussi choisir le mode `Direct` ou le mode pas à pas (`Step/Step`), puis quitter ce menu de configuration avec `CANCEL` du bandeau.
Après chaque commande, il faut taper `ENTER`, on oubliera souvent de le spécifier ! ! !
Dans ce qui suit, vous trouverez une partie de l'épreuve de mathé-

matiques 1999 du Brevet d'Amiens et de l'épreuve de mathématiques 1999 (série S) du Bac.
On a essayé de faire faire le plus de choses possibles à la HP40G...
On remarquera, qu'il reste quand même à l'élève le soin de justifier les calculs et de connaitre la démarche à suivre lorsqu'il utilise le pas à pas actif....

5.2 Exercices donnés au Brevet

5.2.1 Exercice 1

On pose A :

$$\frac{\frac{3}{2}-1}{\frac{1}{2}+1}$$

Vous ferez apparaître chaque étape de calcul et vous donnerez le résultat de A sous la forme d'une fraction irréductible.

Dans l'éditeur d'équations on entre la valeur de A, on tape :
`3 ÷ 2 ▷ − 1 ▷▷ ÷ 1 ÷ 2 ▷ + 1`
▷ sélectionne le dénominateur.
`ENTER` fait la simplification du dénominateur, on obtient :

$$\frac{\frac{3}{2}-1}{\frac{3}{2}}$$

puis on sélectionne le numérateur avec ◁
`ENTER` fait la simplification du numérateur, on obtient :

$$\frac{\frac{1}{2}}{\frac{3}{2}}$$

△ sélectionne la fraction entière et `ENTER` fait la simplification de la fraction, on obtient :

$$\frac{1}{3}$$

5.2.2 Exercice 2

On considère le nombre C :

$$C = 2\sqrt{45} + 3\sqrt{12} - \sqrt{20} - 6\sqrt{3}$$

Écrire C sous la forme $d\sqrt{5}$ où d est un nombre entier.

Dans l'éditeur d'équations on entre la valeur de C, on tape :
2$\sqrt{}$ 45 ▷▷ + 3$\sqrt{}$ 12 ▷▷ − $\sqrt{}$ 20 ▷▷ − 6$\sqrt{}$ 3
▷▷▷ sélectionne $-6\sqrt{3}$ et
◁ sélectionne $-\sqrt{20}$
▽▽ sélectionne 20
On appelle la commande FACTOR qui se trouve dans le menu ALGB,
puis ENTER effectue la factorisation de 20 en $2^2 \cdot 5$,
△ sélectionne $\sqrt{2^2 \cdot 5}$ et ENTER renvoie $2\sqrt{5}$
▷ sélectionne $-2\sqrt{5}$
SHIFT◁ échange $3\sqrt{12}$ et $-2\sqrt{5}$
◁ sélectionne $2\sqrt{45}$
▽▽ sélectionne 45
On appelle la commande FACTOR qui se trouve dans le menu ALGB,
puis ENTER effectue la factorisation de 45 en $3^2 \cdot 5$,
△ sélectionne $\sqrt{3^2 \cdot 5}$ et ENTER remplace $\sqrt{3^2 \cdot 5}$ par $3\sqrt{5}$
△ sélectionne $2 \cdot 3\sqrt{5}$
SHIFT▷ sélectionne $2 \cdot 3\sqrt{5}$ et $-2\sqrt{5}$ puis ENTER effectue l'opération et on obtient :
$4\sqrt{5}$
Il reste à transformer $3\sqrt{12}$ et voir que ce terme se simplifie avec $-6\sqrt{3}$
Donc :
$C = 4\sqrt{5}$

5.2.3 Exercice 3

On considère l'expression $D = (3x-1)^2 - 81$
1.Développer et réduire D
2.Factoriser D
3.Résoudre l'équation : $(3x-10)(3x+8) = 0$
4.Calculer D pour $x = -5$

1. On écrit D dans l'éditeur d'équations :
On tape :
3 X − 1 ▷▷ x^y 2 ▷ − 81
On sélectionne $(3X-1)^2$ (▷ ◁) puis ENTER développe cette expression. On obtient :
$9X^2 - 6X + 1 - 81$

Pour faire du pas à pas actif on applique :
`POWEXPAND` à $(3.X - 1)^2$ puis on applique `DISTRIB` au résultat obtenu pour obtenir :
$9X^2 - 6X + 1$
$\triangle$ sélectionne toute l'expression et `ENTER` la réduit en :
$9X^2 - 6X - 80$
2. On va chercher D dans l'historique (touche `HOME`), on met en surbrillance D puis on valide avec `ENTER`.
On appelle `FACTOR` et on obtient :
$(3X + 8)(3X - 10)$
On aurait pu aussi sélectionner `81` pour le factoriser en 3^4 et reconnaitre la diffèrence de deux carrés...
3. On appelle la commande `SOLVEVX`, puis `ENTER` renvoie :

$$(X = -\frac{8}{3}) \text{ OR } (X = \frac{10}{3})$$

4.On va chercher D dans l'historique (touche `HOME`), on met en surbrillance D puis on valide avec `ENTER`)
On appelle la fonction `SUBST`, on complète le deuxième argument :
`X=-5` puis $\triangleright \triangleright \triangleright$ pour sélectionner le tout puis `ENTER`
On obtient :
$(3 \cdot (-5) - 1)^2 - 81$ puis `ENTER` donne le résultat :
`175`
Donc $D = 175$

5.2.4 Exercice 4

Un confiseur prépare deux sortes de boites contenant des tuiles et des macarons.
Dans le paquet de la première sorte, il place 17 tuiles et 20 macarons.
Dans le paquet de la deuxième sorte, il place 10 tuiles et 25 macarons.
Ces paquets sont vendus 90F.
Calculer le prix d'une tuile et celui d'un macaron.

Soit x le prix en Francs d'une tuile et y celui d'un macaron.
On a à résoudre :

$$\begin{cases} 17x + 20y & = 90 \\ 10x + 25y & = 90 \end{cases}$$

On tape dans l'éditeur d'équations :
$\texttt{LINSOLVE}(17 \cdot X + 20 \cdot Y - 90 \text{ AND } 10 \cdot X + 25 \cdot Y - 90\ ,\ X \text{ AND } Y)$

Si on est en mode pas à pas on obtient :

$$\begin{array}{l}\mathtt{L_2 = 17L_2 - 10L_1}\\ \begin{bmatrix} 17 & 20 & -90 \\ 10 & 25 & -90 \end{bmatrix}\end{array}$$

$$\begin{array}{l}\mathtt{L_1 = 45L_1 - 4L_2}\\ \begin{bmatrix} 17 & 20 & -90 \\ 0 & 225 & -630 \end{bmatrix}\end{array}$$

$$\begin{array}{l}\mathtt{Reduction\ Result}\\ \begin{bmatrix} 765 & 0 & -90 \\ 0 & 225 & -630 \end{bmatrix}\end{array}$$

puis ENTER donne le résultat :

$$(\mathtt{X} = 2)\ \mathtt{AND}\ (\mathtt{Y} = \frac{14}{5})$$

En mettant en surbrillance $\frac{14}{5}$, et en appuyant sur la touche NUM,ou en appellant XNUM, on obtient :

$$(\mathtt{X} = 2)\ \mathtt{AND}\ (\mathtt{Y} = 2.8)$$

ATTENTION ! vous êtes passer en mode Approx, repasser en mode Exact avec CFG.
Le prix d'une tuile est donc de 2 francs et celui d'un macaron de 2.80 francs

5.2.5 Exercice 5

Le plan muni d'un repère orthonormal (O,i,j) et l'unité de longueur est le centimètre. On appelle A et B les points dont les coordonnées sont :
A(-1 ; 3) et B(-3 ; -1).
1/ Calculer AB, en donnant sa valeur exacte en centimètres.
2/ Determiner l'équation de la droite AB.
Première façon :
On tape :
STORE((-1,3),A)
STORE((-3,-1),B)
Le vecteur $\overrightarrow{AB}$ a pour coordonnées B − A
1/ On tape :

$$\mathtt{ABS(B - A)}$$

On obient :

$$2\sqrt{5}$$

2/ On tape :

$$\texttt{DROITE(A, B)}$$

On obient :

$$\texttt{y} = 2.\texttt{x} + 5$$

OU deuxième façon :
1/ On tape directement :

$$(-3, -1) - (-1, 3)$$

On obient :

$$-2 - 4.\texttt{i}$$

On tape :

$$\texttt{ABS}(-2 - 4\texttt{i})$$

On obient :

$$2\sqrt{5}$$

2/ On tape :

$$\texttt{DROITE}((-1, 3), (-3, -1))$$

On obient :

$$\texttt{y} = 2.(\texttt{x} - -1) + 3$$

puis `ENTER` donne :

$$\texttt{y} = 2.\texttt{x} + 5$$

5.3 Exercices donnés au Bac

5.3.1 Exercice 1

L'objet de cet exercice est de tracer la courbe Γ décrite par M d'affixe $\frac{1}{2} \cdot z^2 - z$, lorsque m d'affixe z décrit le cercle C de centre O et de rayon 1. Soit t un réel de $[-\pi, \pi]$ et m le point de C d'affixe $z = e^{i \cdot t}$.

1. Calcul des coordonnées de M :
   On entre tout d'abord l'expression $\frac{1}{2} \cdot z^2 - z$ dans l'éditeur d'équations.

On tape dans l'éditeur d'équations :

$$\texttt{ALPHA Z x}^{\texttt{y}}\ 2\ \rhd\ \div\ 2\ \rhd\ -\ \texttt{ALPHA Z}\ \rhd\ \rhd$$

L'expression $\frac{\mathtt{Z}^2}{2} - \mathtt{Z}$ est sélectionnée.
Puisque $z = e^{i \cdot t}$, on appelle `SUBST` et on complète le deuxième argument :

$$\mathtt{SUBST}(\frac{\mathtt{Z}^2}{2} - \mathtt{Z}, \mathtt{Z} = \mathtt{EXP}(\mathtt{i} \times \mathtt{t}))$$

la réponse est :

$$\frac{\mathtt{EXP}(\mathtt{i} \cdot \mathtt{t})^2}{2} - \mathtt{EXP}(\mathtt{i} \cdot \mathtt{t})$$

On linéarise ensuite l'expression avec l'appel de :
`LIN`
la réponse est :

$$\frac{1}{2} \cdot \mathtt{EXP}(2 \cdot \mathtt{i} \cdot \mathtt{t}) + -1 \cdot \mathtt{EXP}(\mathtt{i} \cdot \mathtt{t})$$

– Puis on appelle `STORE` que l'on compléte pour avoir :

$$\mathtt{STORE}(\frac{1}{2} \cdot \mathtt{EXP}(2 \cdot \mathtt{i} \cdot \mathtt{t}) + -1 \cdot \mathtt{EXP}(\mathtt{i} \cdot \mathtt{t}), \mathtt{M})$$

puis `ENTER`
On cherche maintenant la partie réelle de cette expression avec l'appel de :

`RE`

la réponse est :

$$\frac{\mathtt{COS}(\mathtt{t} \cdot 2) - 2 \cdot \mathtt{COS}(\mathtt{t})}{2}$$

On définit alors la fonction $x(t)$, on appelle `DEF` :
ATTENTION!!! Il faut taper `= X(t)` puis échanger `X(t)` et l'expression $\frac{\mathtt{COS}(\mathtt{t} \cdot 2) - 2 \cdot \mathtt{COS}(\mathtt{t})}{2}$ en mettant en surbrillance `X(t)` avec $\rhd$ puis en tapant `SHIFT`$\lhd$ pour l'échange.
On obtient :

$$\mathtt{DEF}\ (\mathtt{X}(\mathtt{t}) = \frac{\mathtt{COS}(\mathtt{t} \cdot 2) - 2 \cdot \mathtt{COS}(\mathtt{t})}{2})$$

puis `ENTER`

– On cherche ensuite la partie imaginaire on tape :

$$\mathtt{IM(M)}$$

la réponse est :

$$\frac{\mathtt{SIN(t \cdot 2) - 2 \cdot SIN(t)}}{\mathtt{2}}$$

On définit alors la fonction $y(t)$ (de la même façon que $x(t)$) :

$$\mathtt{DEF(Y(t)} = \frac{\mathtt{SIN(t \cdot 2) - 2 \cdot SIN(t)}}{\mathtt{2}}$$

puis `ENTER`

2. On cherche un axe de symétrie de Γ, pour cela on calcule $x(-t)$ et $y(-t)$ en tapant :

$$\mathtt{X(-t)\ ENTER}$$

la réponse est :

$$\frac{\mathtt{COS(t \cdot 2) - 2 \cdot COS(t)}}{\mathtt{2}}$$

On a donc : $x(-t) \ = \ x(t)$
puis :

$$\mathtt{Y(-t)\ ENTER}$$

la réponse est :

$$\frac{\mathtt{-SIN(t \cdot 2) + 2 \cdot SIN(t)}}{\mathtt{2}}$$

On a donc : $y(-t) \ = \ -y(t)$
Si $M_1(x(t), y(t))$ est sur Γ, $M_2(x(-t), y(-t)$ est aussi sur Γ.
On vient de monter que M_1 et M_2 sont symétriques par rapport à Ox, donc on en déduit que l'axe Ox est un axe de symétrie de Γ.

3. Calcul de $x'(t)$:
On tape :

$$\mathtt{DERIV(X(t), t)}$$

la réponse est :

$$\frac{\mathtt{2 \cdot (-2 \cdot SIN(t \cdot 2) - 2 \cdot (-SIN(t)))}}{\mathtt{4}}$$

après simplification (ENTER) :

$$-(\mathtt{SIN(t \cdot 2)} - \mathtt{SIN(t)})$$

On développe l'expression (transformation de $\mathtt{SIN(2 \cdot t)}$), on appelle TEXPAND et on obtient :

$$\mathtt{TEXPAND}(-(\mathtt{SIN(t \cdot 2)} - \mathtt{SIN(t)}))$$

puis ENTER
la réponse est :

$$-(\mathtt{SIN(t)} \cdot 2 \cdot \mathtt{COS(t)} - \mathtt{SIN(t)})$$

puis on factorise, on appelle FACTOR et on obtient : :

$$\mathtt{FACTOR}(-(\mathtt{SIN(t)} \cdot 2 \cdot \mathtt{COS(t)} - \mathtt{SIN(t)}))$$

puis ENTER
la réponse est :

$$-\mathtt{SIN(t)} \cdot (2 \cdot \mathtt{COS(t)} - 1)$$

On peut alors définir la fonction $x'(t)$ en appelant DEF.
Il faut taper `= X1(t)` puis,
échanger `X1(t)` et l'expression $-\mathtt{SIN(t)} \cdot (2 \cdot \mathtt{COS(t)} - 1)$, en mettant en surbrillance `X1(t)` ($\triangleright$) puis, taper SHIFT $\triangleleft$ pour l'échange.
On obtient :

$$\mathtt{DEF(\ X1(t)} = -\mathtt{SIN(t)} \cdot (2 \cdot \mathtt{COS(t)} - 1))$$

puis ENTER

4. Calcul de $y'(t)$:
   On tape :

   $$\mathtt{DERIV(Y(t), t)}$$

   la réponse est :

   $$\frac{2 \cdot (2 \cdot \mathtt{COS(t \cdot 2)}) - 2 \cdot \mathtt{COS(t)}}{4}$$

   après simplification (ENTER) :

   $$\mathtt{COS(t \cdot 2)} - \mathtt{COS(t)}$$

On développe l'expression (transformation de $\mathtt{COS}(2 \cdot \mathtt{t})$), on appelle `TEXPAND` :

$$\mathtt{TEXPAND}(\mathtt{COS}(\mathtt{t} \cdot 2) - \mathtt{COS}(\mathtt{t}))$$

puis `ENTER`
la réponse est :

$$2 \cdot \mathtt{COS}(\mathtt{t})^2 - 1 - \mathtt{COS}(\mathtt{t}))$$

puis on factorise :

$$\mathtt{FACTOR}(2 \cdot \mathtt{COS}(\mathtt{t})^2 - 1 - \mathtt{COS}(\mathtt{t})))$$

puis `ENTER`
la réponse est :

$$(\mathtt{COS}(\mathtt{t}) - 1) \cdot (2 \cdot \mathtt{COS}(\mathtt{t}) + 1)$$

On peut alors définir la fonction $y'(t)$, on tape (comme pour $x'(t)$:

$$\mathtt{DEF(\ Y1(t)} = (\mathtt{COS}(\mathtt{t}) - 1) \cdot (2 \cdot \mathtt{COS}(\mathtt{t}) + 1))$$

5. Variations de $x(t)$ et de $y(t)$
   Pour cela on trace sur le même graphique $x(t)$ et $y(t)$.
   On met `t` comme variable `VX` (touches `SHIFT SYMB (SETUP)`), puis on tape dans l'éditeur d'équations `X(t)` puis `ENTER`.
   On appuie alors sur la touche `PLOT`
   On sélectionne `Function` à l'aide de la boite de dialogues et `F1` comme destination.
   Puis on fait la même chose avec `Y(t)` en chosissant `F2` comme destination.
   Puis on quitte le `CAS` avec la touche `ON (CANCEL)`, pour faire le graphe des fonctions ainsi recopiées, on se place dans l'`Aplet Function`, et on coche `F1` et `F2`. Il faut régler les paramètres de la fenêtre (`SHIFT PLOT`), puis `PLOT` pour avoir le graphique.
6. Tracé de la courbe Γ :
   – Valeurs de $x(t)$ et de $y(t)$
     On trouve les valeurs de $x(t)$ et de $y(t)$ pour $t = 0, \frac{\pi}{3}, \frac{2 \cdot \pi}{3}, \pi$ en tapant successivement :

$$\mathtt{X(0)\ ENTER}$$

réponse : $\frac{-1}{2}$

$$\texttt{X}(\frac{\pi}{3})\ \texttt{ENTER}$$

réponse : $\frac{-3}{4}$

$$\texttt{X}(2\times\frac{\pi}{3})\ \texttt{ENTER}$$

réponse : $\frac{1}{4}$

$$\texttt{X}(\pi)\ \texttt{ENTER}$$

réponse : $\frac{3}{2}$

$$\texttt{Y}(0)\ \texttt{ENTER}$$

réponse : 0

$$\texttt{Y}(\frac{\pi}{3})\ \texttt{ENTER}$$

réponse : $\frac{-\sqrt{3}}{4}$

$$\texttt{Y}(2\times\frac{\pi}{3})\ \texttt{ENTER}$$

réponse : $\frac{-3\cdot\sqrt{3}}{4}$

$$\texttt{Y}(\pi)\ \texttt{ENTER}$$

réponse : 0

– Pente des tangentes ($m = \frac{y'(t)}{x'(t)}$)
  On trouve les valeurs de $\frac{y'(t)}{x'(t)}$ pour $t = 0, \frac{\pi}{3}, \frac{2\cdot\pi}{3}, \pi$ en tapant successivement :

$$\texttt{LIMIT}(\frac{\texttt{Y1(t)}}{\texttt{X1(t)}}, \texttt{t} = 0)\ \texttt{ENTER}$$

réponse : 0

$$\texttt{LIMIT}(\frac{\texttt{Y1(t)}}{\texttt{X1(t)}}, \texttt{t} = \pi \div 3)\ \texttt{ENTER}$$

réponse : ∞

$$\texttt{LIMIT}(\frac{\texttt{Y1(t)}}{\texttt{X1(t)}}, \texttt{t} = 2\times\pi \div 3)\ \texttt{ENTER}$$

réponse : 0

$$\texttt{LIMIT}(\frac{\texttt{Y1(t)}}{\texttt{X1(t)}}, \texttt{t} = \pi)\ \texttt{ENTER}$$

réponse : ∞
Voici les variations de $x(t)$ et de $y(t)$

t	0		$\frac{\pi}{3}$		$\frac{2\pi}{3}$		π
$x'(t)$	0	$-$	0	$+$	$?$	$+$	0
$x(t)$	$\frac{-1}{2}$	$\downarrow$	$\frac{-3}{4}$	$\uparrow$	$\frac{1}{4}$	$\uparrow$	$\frac{3}{2}$
$y(t)$	0	$\downarrow$	$\frac{\sqrt{3}}{4}$	$\downarrow$	$\frac{-3\sqrt{3}}{4}$	$\uparrow$	0
$y'(t)$	0	$-$	$?$	$-$	0	$+$	$?$
m	0		∞		0		∞

- Courbe Γ :
  On fait ensuite le tracé de la courbe en paramétrique.
  On tape `X(t)` + `i` × `Y(t)` dans l'éditeur d''équations puis `ENTER`.
  On tape ensuite :
  `PLOT` et on sélectionne `Parametric`, à l'aide de la boite de dialogues et `X1,Y1` comme destination. Puis on quitte le `CAS` avec la touche `ON (CANCEL)`, et pour faire le graphe de la courbe Γ on lance l'`Aplet Parametric`.

5.3.2 Exercice 2 (de spécialité)

On définit pour n entier naturel :

$$a_n = 4 \times 10^n - 1,\ b_n = 2 \times 10^n - 1 \text{ et } c_n = 2 \times 10^n + 1$$

On tape donc :

$$\texttt{DEF(A(N)} = \texttt{4} \cdot \texttt{10}^{\texttt{N}} - \texttt{1)}$$

$$\texttt{DEF(B(N)} = \texttt{2} \cdot \texttt{10}^{\texttt{N}} - \texttt{1)}$$

$$\texttt{DEF(C(N)} = \texttt{2} \cdot \texttt{10}^{\texttt{N}} + \texttt{1)}$$

1. – a) Calcul de $a_1, b_1, c_1, a_2, b_2, c_2, a_3, b_3, c_3$:
   Il suffit de taper :

   `A(1)`

   réponse 39

   `B(1)`

   réponse 19

   `C(1)`

   réponse 21

   `A(2)`

réponse 399

`B(2)`

réponse 199

`C(2)`

réponse 201

`A(3)`

réponse 3999

`B(3)`

réponse 1999

`C(3)`

réponse 2001

- b) nombre de chiffres et divisibilité
  Ici, la calculatrice n'est là que pour faire des essais pour différentes valeurs de n...
  On sait que les entiers n vérifiant :

$$10^n \leq n < 10^{n+1}$$

ont $(n+1)$ chiffres dans l'écriture décimale.
On a :

$$3 \cdot 10^n \ < \ a_n \ < \ 4 \cdot 10^n$$

$$10^n \ < \ b_n \ < \ 2 \cdot 10^n$$

$$2 \cdot 10^n \ < \ c_n \ < \ 3 \cdot 10^n$$

donc $a_n,\ b_n,\ c_n$ ont $(n+1)$ chiffres dans l'écriture décimale.
De plus $d_n = 10^n - 1$ est divisible par 9, car son écriture décimale ne comporte que des 9.
On a

$$a_n = 3 \cdot 10^n \ + \ d_n$$

et

$$c_n = 3 \cdot 10^n \ - \ d_n$$

donc a_n et c_n sont divisibles par 3.

- c) b_3 est premier
  On tape :

  `ISPRIME?(B(3))`

  On obtient :

  `1.`

  ce qui veut dire `vrai`
  Pour montrer que $b_3 = 1999$ est premier, il suffit de tester si 1999 est divisible par tous les nombres premiers inférieurs ou égaux à $\sqrt{1999}$.
  Comme on a $1999 < 2025 = 45^2$, on teste la divisibilité de 1999 avec :
  $n = 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41.$
  1999 n'étant divisible par aucun de ces nombres on en déduit que 1999 est premier.
- d) $a_n = b_n \times c_n$
  On tape :

  `B(N) · C(N)`

  On obtient :

  $$4 \cdot (10^{N})^2 - 1$$

  qui est bien la valeur de a_n
  Décomposition en facteur premier de a_6
  On tape :

  `FACTOR(A(6))`

  On obtient :

  `3 · 23 · 29 · 1999`

- e) b_n et c_n sont premiers entre eux.
  Ici, la calculatrice n'est là que pour faire des essais pour différentes valeurs de n...
  Pour montrer que c_n et b_n sont premiers entre eux il suffit de remarquer que :

  $$c_n = b_n + 2$$

  Ainsi, les diviseurs communs à c_n et b_n sont les diviseurs communs à b_n et 2 et sont aussi, les diviseurs communs à c_n et 2. b_n et 2 sont premiers entre eux car b_n est un nombre premier différent de 2. Donc

  $$PGCD(c_n,\ b_n) = PGCD(c_n,\ 2) = PGCD(b_n,\ 2) = 1$$

2. On considère l'équation :

$$b_3 \cdot x + c_3 \cdot y = 1$$

– a) Il y a au moins une solution car il s'agit de l'identité de Bézout.
En effet, le théorème de Bézout dit :
Si a et b sont premiers entre eux, il existe x et y vérifiant :

$$a \cdot x + b \cdot y = 1$$

Donc, l'équation :

$$b_3 \cdot x + c_3 \cdot y = 1$$

a au moins une solution.

– b)On tape :

$$\texttt{IEGCD(B(3), C(3))}$$

On obtient :

$$\texttt{1000 AND } -\texttt{999} = \texttt{1}$$

ce qui veut dire que l'on a :

$$b_3 \times 1000 + c_3 \times (-999) = 1$$

on a donc une solution particulière :
$x = 1000,\ y = -999$.
À la main, on écrit :
$c_3 = b_3 + 2$ et $b_3 = 999 \times 2 + 1$
donc, $b_3 = 999 \times (c_3 - b_3) + 1$ ainsi :

$$b_3 \times 1000 + c_3 \times (-999) = 1$$

– c) Ici, la calculatrice ne peut pas trouver la solution générale.
On a :

$$b_3 \cdot x + c_3 \cdot y = 1$$

et

$$b_3 \times 1000 + c_3 \times (-999) = 1$$

donc par soustraction, on a :

$$b_3 \cdot (x - 1000) + c_3 \cdot (y + 999) = 0$$

ou encore :

$$b_3 \cdot (x - 1000) \;=\; -c_3 \cdot (y + 999)$$

D'après le théorème de Gauss : c_3 est premier avec b_3 donc, c_3 divise $(x - 1000)$.
Il existe donc $k \in Z$ tel que :

$$(x - 1000) = k \times c_3$$

et

$$-(y + 999) = k \times b_3$$

Réciproquement, soit

$$x \;=\; 1000 \;+\; k \times \; c_3$$

et

$$y \;=\; -999 \;-\; k \times \; b_3 \quad \text{pour} \quad k \in Z$$

On a :

$$b_3 \cdot x + c_3 \cdot y = b_3 \times 1000 + c_3 \times (-999) \;=\; 1$$

La solution générale est donc pour tout $k \in Z$:

$$x \;=\; 1000 \;+\; k \times \; c_3$$

$$y \;=\; -999 \;-\; k \times \; b_3$$

5.3.3 Exercice 2 (pas de spécialité)

Vérifiez avant de commencer que vous êtes bien en mode réel exact avec `X` comme variable courante, sinon sélectionnez `Default cfg` de `CFG`.
On considère la suite

$$u_n = \int_0^2 \frac{2x+3}{x+2}\, e^{\frac{x}{n}}\, dx$$

1. – a) Variation de $g(x) = \frac{2x+3}{x+2}$ pour $x \in [0, 2]$
   On tape :

   $$\mathtt{DEF(G(X)} = \frac{\mathtt{2X+3}}{\mathtt{X+2}})$$

puis :

`TABVAR(G(X))`

On obtient :

$$\begin{array}{cccccc} -\infty & + & -2 & + & +\infty & X \\ 2 & \uparrow & \infty & \uparrow & 2 & F \end{array}$$

La première ligne donne le signe de `g'(x)` selon x, et la deuxième ligne les variations de `g(x)`. On remarquera que pour `TABVAR` la fonction s'appelle toujours F.
On en déduit donc que $g(x)$ est croissante sur $[0, 2]$.
Si on est en mode pas à pas (pour cela il faut valider `Step/Step` avec `OK` du bandeau de `CFG`), on obtient alors (quoiqu'il arrive la fonction est notée `F`) :

$$\mathtt{F} =: \frac{2 \cdot \mathtt{X} + 3}{\mathtt{X} + 2}$$

puis `ENTER`

$$\mathtt{F}' := \frac{2 \cdot (\mathtt{X} + 2) - (2 \cdot \mathtt{X} + 3)}{\mathtt{SQ}(\mathtt{X} + 2)}$$

puis en se servant de la flèche $\bigtriangledown$ pour faire défiler l'écran

$$\rightarrow \frac{1}{(\mathtt{X} + 2)^2}$$

puis `ENTER` pour obtenir le tableau de variations.
Si on n'est pas en mode pas à pas, on peut aussi demander le calcul de la dérivée en tapant :

`DERVX(G(X))`

ce qui donne le calcul ci-dessus.
On calcule $g(0)$ et $g(2)$, pour cela on tape :

`G(0)`

réponse $\frac{3}{2}$

`G(2)`

réponse $\frac{7}{4}$
d'où, l'encadrement

$$\frac{3}{2} \leq g(x) \leq \frac{7}{4} \quad \text{pour } x \in [0, 2]$$

.

– b) Là, la calculatrice ne peut rien ...il suffit de dire que

$$e^{\frac{x}{n}} \geq 0 \quad \text{pour } x \in [0,2]$$

pour montrer que, pour $x \in [0,2]$, on a :

$$\frac{3}{2}e^{\frac{x}{n}} \leq g(x)e^{\frac{x}{n}} \leq \frac{7}{4}e^{\frac{x}{n}}$$

– c) On intègre l'inégalité ci-dessus, on tape :

$$\int_0^2 \mathtt{e}^{\frac{\mathtt{X}}{\mathtt{N}}}\ \mathtt{dX}$$

On obtient :

$$\mathtt{N} \cdot \mathtt{e}^{\frac{2}{\mathtt{N}}} - \mathtt{N}$$

On en déduit donc :

$$\frac{3}{2}(ne^{\frac{2}{n}} - n) \leq u_n \leq \frac{7}{4}(ne^{\frac{2}{n}} - n)$$

Pour justifier le calcul précédent, il faut dire qu'une primitive de $e^{\frac{x}{n}}$ est $n \cdot e^{\frac{x}{n}}$.
Si on ne le sait pas, on peut toujours taper :

$$\mathtt{INTVX(EXP(}\frac{\mathtt{X}}{\mathtt{N}}\mathtt{))}$$

la réponse est : $\mathtt{N} \cdot \mathtt{e}^{\frac{\mathtt{X}}{\mathtt{N}}}$

– d) On cherche la limite de $(ne^{\frac{2}{n}} - n)$ quand $n \to +\infty$:

$$\mathtt{LIMIT(N \cdot EXP(}\frac{2}{\mathtt{N}}) - \mathtt{N}\ ,\ \mathtt{N} = +\infty)$$

On obtient :

$$2$$

Attention :
La variable `VX` est maintenant égale à `N`, utiliser les touches `SHIFT SYMB (SETUP)` pour remettre `VX` à `X`.
Pour justifier ce résultat, il faut dire que :

$$\lim_{x \to 0} \frac{e^x - 1}{x} = 1$$

et donc que :

$$\lim_{n\to+\infty} \frac{e^{\frac{2}{n}}-1}{\frac{2}{n}} = 1$$

ou encore :

$$\lim_{n\to+\infty} (e^{\frac{2}{n}}-1)\cdot n = 2$$

Si L existe, en faisant tendre n vers $+\infty$ dans les inégalités de 1b), on obtient :

$$\frac{3}{2}\cdot 2 \leq L \leq \frac{7}{4}\cdot 2$$

2. – a) $g(x) = 2 - \frac{1}{x+2}$ et calcul de $I = \int_0^2 g(x)dx$
On tape :

$$\texttt{PROPFRAC(G(X))}$$

On obtient

$$2 - \frac{1}{\texttt{X}+2}$$

Pour le calcul de l'intégrale I, on tape :

$$\int_0^2 \texttt{G(X)dX}$$

On obtient :

$$-(\texttt{LN}(2)-4)$$

À la main, on a $2x+3 = 2(x+2)-1$ donc

$$g(x) = 2 - \frac{1}{x+2}$$

On intègre ensuite terme à terme entre 0 et 2, on obtient :

$$\int_0^2 g(x)dx = [2x - \ln(x+2)]_{x=0}^{x=2}$$

c'est à dire, puisque $\ln 4 = 2\ln 2$:

$$\int_0^2 g(x)dx = 4 - \ln 2$$

- b) Là, la calculatrice ne peut rien...il suffit de dire que $e^{\frac{x}{n}}$ est croissante pour $x \in [0,2]$, pour obtenir l'inégalité :

$$1 \ \leq \ e^{\frac{x}{n}} \ \leq e^{\frac{2}{n}}$$

puis par multiplication, $g(x)$ étant positif sur $[0,2]$, on a :

$$g(x) \ \leq \ g(x)e^{\frac{x}{n}} \ \leq g(x)e^{\frac{2}{n}}$$

puis en intégrant on a :

$$I \ \leq u_n \ \leq \ e^{\frac{2}{n}}I$$

- c) Convergence de u_n
On cherche la limite de $e^{\frac{2}{n}}$ quand $n \to +\infty$:

$$\texttt{LIMIT(EXP(}\frac{\texttt{2}}{\texttt{N}}\texttt{)}\ ,\ \texttt{N} = +\infty)$$

On obtient :

$$\texttt{1}$$

En effet, $\frac{2}{n}$ tend vers 0 lorsque n tend vers $+\infty$ donc, $e^{\frac{2}{n}}$ tend vers $e^0 = 1$ lorsque n tend vers $+\infty$.
Lorsque n tend vers $+\infty$, u_n reste compris entre I et une quantité qui tend vers I (cf inégalités 2b)).
Donc u_n converge et sa limite vaut I.
On a donc montré que :

$$L = I = 4 - \ln 2$$

5.4 Conclusion

On voit qu'un bon maniement de la calculatrice HP40G permet de faire une bonne partie des questions...
Il faut cependant noter, qu'en arithmétique il faut faire plus de raisonnements : la calculatrice permet alors de faire des vérifications....

Chapitre 6

Programmation

6.1 Implémentation

6.1.1 Comment éditer et sauver un programme

Pour avoir accès au catalogue de programmes, on appuie sur les touches `SHIFT 1 (PROGRAM)`.
Il apparait alors un écran contenant la liste des programmes disponibles et un bandeau (`EDIT NEW SEND RECV RUN`).
Pour taper un nouveau programme, on appuie sur `F2 (NEW)`.
On vous demande le nom du programme : ATTENTION ! vous n'êtes pas en mode `Alpha` appuyer sur `F4 (A..Z)` pour y être !!!.
Tapez son nom puis `F6 (OK)`.
Vous entrez votre programme et votre travail est automatiquement sauvegardé lorsque vous sortez de l'éditeur en appuyant sur `HOME` ou sur `SHIFT 1 (PROGRAM)`.

6.1.2 Comment corriger un programme

Si la syntaxe est mauvaise, la machine vous dit :
`Invalid Syntax Edit program ?` Vous répondez `F6 (YES)`.
La machine vous met automatiquement le curseur là où le compilateur a détecté l'erreur. Il suffit donc de corriger !!!

6.1.3 Comment exécuter un programme

Pour exécuter un programme, on ouvre le catalogue de programmes, en appuyant sur les touches `SHIFT 1 (PROGRAM)`.
Il apparait alors un écran contenant la liste des programmes disponibles et le bandeau `EDIT NEW SEND RECV RUN`.
On met le nom du programme à exécuter en surbrillance et on appuie sur `F6 (RUN)`.

6.1.4 Comment modifier un programme

Pour modifier un programme (sans vouloir garder l'ancien) on ouvre le catalogue de programmes, en appuyant sur les touches `SHIFT 1 (PROGRAM)`. Il apparait alors un écran contenant la liste des programmes disponibles et le bandeau `EDIT NEW SEND RECV RUN`.
On met le nom du programme à modifier en surbrillance et on appuie sur `F1 (EDIT)`.
Si vous voulez avoir à la fois l'ancien et le nouveau programme il faut :
- ouvrir le catalogue de programmes (`SHIFT 1 (PROGRAM)`).
- appuyer sur `F2 (NEW)` et taper le nom du programme modifié puis `F6 (OK)`.
L'éditeur s'ouvre, on appuie alors sur `VARS` puis la lettre `P` pour mettre `Program` en surbrillance.
Avec les flèches mettre le nom du programme à modifier en surbrillance et appuyer sur `F4 (VALUE)` (pour cocher `VALUE` du bandeau) puis `F6 (OK)`.
Cela recopie le texte du programme dans l'éditeur.

6.2 Les commentaires

Il faut prendre l'habitude de commenter ses programmes.
En algorithmique un commentaire commence par `//` et se termine par un passage à la ligne.
Pour la `HP40G`, un commentaire commence par `@` et se termine par un passage à la ligne ou est entouré de deux `@`.
ATTENTION!!!
Ne pas oublier de mettre un espace après `@`.
Le caractère `@` est obtenu en tapant `shift VAR (CHARS)`, puis on met

ce caractère en surbrillance, puis `ECHO1` du bandeau.

6.3 Les variables

6.3.1 Leurs noms

Ce sont les endroits où l'on peut stocker des valeurs, des nombres, des expressions, des objets.
Avec la `HP40G`, on n'a droit en progammation qu'aux 26 lettres de l'alphabet pour stocker des nombres réels.

6.3.2 Notion de variables locales

Cette notion n'existe pas pour la calculatrice `HP40G`.
On ne peut utiliser que des variables globales.

6.3.3 Notion de paramètres

Quand on écrit un programme sur la `HP40G`, il n'est pas possible de lui passer des paramètres.
On ne peut donc pas écrire de fonctions ayant des paramètres, avec le langage de programmation de la `HP40G`.

6.4 Les Entrées

6.4.1 Traduction en Algorithmique

Pour que l'utilisateur puisse entrer une valeur dans la variable `A` au cours de l'exécution d'un programme, on écrira, en algorithmique :
`saisir A`
Et pour entrer des valeurs dans `A` et `B` on écrira :
`saisir A,B`

6.4.2 Traduction HP40G

`INPUT A;"TITRE";"A=";;0 :`
Si le fait d'avoir à écrire tous ces points virgule dans `INPUT` vous rebute, il est préférable d'utiliser la commande : `PROMPT` (Merci Jean Yves!!!).
`PROMPT A :` ouvre une fenêtre vous demandant d'entrer la valeur de `A`.
Dans ce qui suit, les programmes écrits avant l'existence de `PROMPT`, utilisent le sous-programme IN qui permet d'entrer deux valeurs dans `A` et `B`.

6.5 Les Sorties

6.5.1 Traduction en Algorithmique

En algorithmique on écrit :
`Afficher "A=",A`

6.5.2 Traduction HP40G

`DISP 3;"A="A :` 3 représente le numéro de la ligne où `A` sera affiché
ou
`MSGBOX "A="A :`

6.6 La séquence d'instructions ou action

Une action est une séquence d'une ou plusieurs instructions.

6.6.1 Traduction en Algorithmique

En langage algorithmique, on utilisera l'espace ou le passage à la ligne pour terminer une instruction.

6.6.2 Traduction HP40G

`:` indique la fin d'une instruction.

6.7 L'instruction d'affectation

L'affectation est utilisée pour stocker une valeur ou une expression dans une variable.

6.7.1 Traduction en Algorithmique

En algorithmique on écrira par exemple :
`2*A->B` pour stocker `2*A` dans `B`

6.7.2 Traduction HP40G

La flèche est obtenue à l'aide de la touche `STO▷` du bandeau.
On écrira par exemple :
`2 * A STO ▷ B`

6.8 Les instructions conditionnelles

6.8.1 Traduction en Algorithmique

```
Si condition alors
action
fsi
Si condition alors
action1 sinon
action2
fsi
```

Exemple :

```
Si A = 10 ou A < B alors
B-A->B sinon
A-B->A
fsi
```

6.8.2 Traduction HP40G

```
IF condition THEN
action :
END :
IF condition THEN
action1 : ELSE
```

```
action2 :
END :
```

ATTENTION au == pour traduire la condition d'égalité.
Exemple :

```
IF A == 10 OR A < B THEN
B − A STO ▷ B : ELSE
A − B STO ▷ A :
END :
```

6.9 Les instructions "Pour"

6.9.1 Traduction en Algorithmique

```
Pour I de A à B faire action fpour
Pour I de A à B (pas P) faire action fpour
```

6.9.2 Traduction HP40G

```
FOR I = A TO B STEP 1 ; action : END :
FOR I = A TO B STEP P ; action : END :
```

6.10 L'instruction "Tant que"

6.10.1 Traduction en Algorithmique

```
Tant que condition faire action ftantque
```

6.10.2 Traduction HP40G

```
WHILE condition REPEAT action : END :
```

6.11 Les expressions booléennes

Une condition est une fonction qui a comme valeur un booléen, à savoir elle est soit `vraie` soit `fausse`.

6.11.1 Traduction en Algorithmique

Pour exprimer une condition simple on utilise les opérateurs :
`= > >` $\leq$ $\geq$ $\neq$

6.11.2 Traduction HP40G

ATTENTION, pour la calculatrice `HP40G`, l'égalité se traduit par :
`==`
sinon les autres opérateurs sont les mêmes.

6.12 Les opérateurs logiques

6.12.1 Traduction en Algorithmique

Pour traduire des conditions complexes, on utilise les opérateurs logiques :
`ou et non`

6.12.2 Traduction HP40G

`ou et non` se traduisent sur la `HP40G` par `OR AND NOT`

6.13 Les listes

6.13.1 Traduction en Algorithmique

En algorithmique, on utilise les { } pour délimiter une liste.
Par exemple {} désigne la liste vide et {1, 2, 3} est une liste de 3 éléments.
Le + sera utilisé pour concaténer 2 listes, ou une liste et un élément, ou un élément et une liste :
`{1, 2, 3}->TAB`
`TAB + 4 ->TAB` (maintenant `TAB` désigne {1, 2, 3, 4})
`TAB[2]` désigne le deuxième élément de `TAB` ici 2.

6.13.2 Traduction HP40G

Les variables listes ont pour noms : `L0, L1, L2, ... L9`.
On utilise les { } pour délimiter une liste.
Par exemple {1, 2, 3} est une liste de 3 éléments.
Mais {} ne désigne pas la liste vide, il faut utiliser la commande :
`SYSEVAL 259588` pour initialiser la liste `L0` à vide (`SYSEVAL 259589` pour initialiser `L1` à vide...).
Voici quelques commandes utiles :

`MAKELIST(I*I, I , 1, 10, 2)` désigne la liste des carrés des 5 premiers entiers impairs (2 indique le pas de `I`).
`L1(I)` désigne le `I`ème élément de la liste.
`CONCAT (L1, {5})` désigne une liste ayant l'élément 5 en plus des éléments de la liste `L1`.
On peut aussi utiliser :
`AUGMENT(L1,5)` qui désigne une liste ayant l'élément 5 en plus des éléments de la liste `L1`.
`SUB L2 ;L1 ;2 ;4` est une commande qui met dans `L2` les éléments de `L1` ayant des indices allant de 2 à 4.
ATTENTION ! à la diffèrence entre fonctions et commandes :
les fonctions renvoient une valeur, elles ont des parenthèses et leurs arguments se situent dans les parenthèses, et sont séparés par des virgules alors que
les commandes ne renvoient pas de valeurs, et leurs arguments s'écrivent après le nom de la commande, et sont séparés par des points virgules.

6.14 Un exemple : le crible d'Eratosthène

6.14.1 Description

Pour trouver les nombres premiers inférieurs ou égaux à N :

1. On écrit les nombres de 1 à N dans une liste.
2. On barre 1 et on met 2 dans la case P.
   Si $P.P \leq N$ il faut traiter les éléments de P à N.
3. On barre tous les multiples de P à partir de $P.P$.
4. On augmente P de 1
   Si $P.P$ est inférieur ou égal à N, il reste à traiter les éléments non barrés de P à N.
5. On appelle P le plus petit élément non barré de la liste.
6. On refait les points 3 4 5 tant que $P.P$ reste inférieur ou égal à N.

6.14.2 Écriture de l'algorithme

```
Fonction crible(N)
local TAB PREM I P
// TAB et PREM sont des listes
```

```
{} ->TAB
{} ->PREM
pour I de 2 à N faire
  TAB+I -> TAB
fpour
0 +TAB -> TAB
2 -> P
// On a fait les points 1 et 2
//barrer 1 a été réalisé en le remplaçant par 0
//TAB est la liste 0 2 3 4 ...N
tant que P*P ≤ N faire
  pour I de P à E(N/P) faire
   //E(N/P) désigne la partie entière de N/P
   0 -> TAB[I*P]
  fpour
// On a barré tous les multiples de P à partir de P*P
  P+1 -> P
//On cherche le plus petit nombre <= N non barré,\\
// entre P et N
  tant que (P*P ≤ N) et (TAB[P]=0) faire
    P+1 -> P
  ftantque
ftantque
//on écrit le résultat dans une liste PREM
pour I de 2 à N faire
  si TAB[I] ≠ 0 alors
    PREM +I -> PREM
  fsi
fpour
résultat: PREM
```

6.14.3 Traduction HP40G

Voici le programme CRIBLE :
L'utilisateur doit entrer la valeur de N.
A la fin la liste L2 contient les nombres premiers inférieurs ou égaux à N.

```
INPUT N;"CRIBLE";"N=";;10:
```

```
ERASE:
MAKELIST(I,I,1,N,1) -> L1:
0 -> L1(1):
2->P:
WHILE P*P ≤ N REPEAT
  FOR I = P TO INT(N/P) STEP 1;
    0->L1(I*P):
  END:
  DISP 3;""L1:
  P+1->P:
  WHILE P*P ≤ N AND L1(P) == 0 REPEAT
    P+1->P:
  END:
END:
{2}->L2:
@ on sait que 2 est premier
FOR I=3 TO N  STEP 1;
 IF L1(I) ≠ 0 THEN
  CONCAT(L2,{I}) ->L2:
  END:
END:
DISP 3 ;"PREM" L2:
FREEZE:
```

Chapitre 7

Programmes d'arithmétique

7.1 Le PGCD et l'algorithme d'Euclide

Soient A et B deux entiers positifs dont on cherche le $PGCD$.
L'algorithme d'Euclide est basé sur la définition récursive du $PGCD$:

$$\begin{array}{rcl} PGCD(A,0) & = & A \\ PGCD(A,B) & = & PGCD(B, A \mod B) \ si \ B \neq 0 \end{array}$$

où $A \mod B$ désigne le reste de la division euclidienne de A par B.
Voici la description de cet algorithme :
on effectue des divisions euclidiennes successives :

$$\begin{array}{rll} A = & B \times Q_1 + R_1 & 0 \leq R_1 < B \\ B = & R_1 \times Q_2 + R_2 & 0 \leq R_2 < R_1 \\ R_1 = & R_2 \times Q_3 + R_3 & 0 \leq R_3 < R_2 \\ .... & & \end{array}$$

Après un nombre fini d'étapes, il existe un entier n tel que : $R_n = 0$.
on a alors :
$PGCD(A,B) = PGCD(B,R_1) = ...$
$PGCD(R_{n-1},R_n) = PGCD(R_{n-1},0) = R_{n-1}$

7.1.1 Traduction algorithmique

-Version itérative
Si B $\neq 0$ on calcule R=A mod B, puis avec B dans le rôle de A (en mettant B dans A) et R dans le rôle de B (en mettant R dans B) on recommence jusqu'à ce que B=0, le PGCD est alors A.

```
Fonction PGCD(A,B)
Local R
tant que B ≠ 0 faire
  A mod B->R
  B->A
  R->B
ftantque
résultat A
ffonction
```

-Version récursive
On écrit simplement la définition récursive vue plus haut.

```
Fonction PGCD(A,B)
Si B ≠ 0 alors
  résultat PGCD(B,A mod B)
  sinon
  résultat A
fsi
ffonction
```

7.1.2 Traduction HP40G

-Version itérative pour deux entiers
On écrit tout d'abord le sous-programme IN qui permet d'entrer deux nombres A et B :

```
INPUT A;"A";;;1:
INPUT B;"B";;;1:
ERASE:
```

puis on écrit le programme PGCD :

```
RUN IN:
DISP 3;"PGCD "{A,B}:
WHILE B ≠ 0 REPEAT
```

```
A MOD B ->R:
B ->A:
R ->B:
END:
DISP 4;"PGCD "A:
FREEZE:
```

-Version récursive pour deux entiers A et B
Avec la HP40G on ne peut pas écrire des programmes récursifs...mais on peut écrire le programme PGCDR :

```
DISP 3;"PGCD "{A,B}:
FREEZE:
IF B ≠ 0 THEN
A MOD B ->R:
B ->A:
R ->B:
PGCDR:
ELSE
DISP 3;"PGCD "A:
FREEZE:
END:
```

On stocke tout d'abord les valeurs dans A et B.
Le programme PGCDR affiche le PGCD qu'il est en train de calculer.
L'appel récursif PGCDR renvoie au programme PGCDR qu'il faut faire exécuter en appuyant sur `RUN` du bandeau.
Le programme PGCDR affiche ainsi les PGCD intermédiaires calculés.
On peut aussi remplacer `PGCDR` dans le programme précédent par `RUN PGCDR`, pour ne pas avoir à appuyer sur `RUN` du bandeau, et supprimer les affichages intermédiaires, pour utiliser ce programme dans un programme effectuant les entrées et les sorties :
le programme récursif `PGCDR` devient le programme récursif `PR` :

```
IF B ≠ 0 THEN
A MOD B ->R:
B ->A:
R ->B:
RUN PR:
END:
```

On insère le programme `PR` dans un programme effectuant les entrées et les sorties :

```
PROMPT A:
PROMPT B:
RUN PR:
ERASE:
MSGBOX A:
```

-Version itérative pour deux complexes
Si on utilise la fonction du calcul symbolique `IREMAINDER` à la place de `MOD` dans les programmes précédents, `PGCD` (ou `PR`) peut alors avoir comme paramètres des entiers de Gauss à condition de remplacer les noms des variables `A, B, R` par `Z1, Z2, Z3` et de changer le test d'arrêt.
Voici la version itérative :

```
PROMPT Z1:
PROMPT Z2:
DISP 3;"PGCD "{Z1,Z2}:
WHILE ABS(Z2) ≠ 0 REPEAT
XNUM(IREMAINDER(XQ(Z1),XQ(Z2)) ->Z3:
Z2 ->Z1:
Z3 ->Z2:
END:
DISP 4;"PGCD "Z1:
FREEZE:
```

-Version itérative pour deux polynômes
Les variables `E1, E2,...` permettent de stocker des expressions, c'est ce qu'il nous faut pour y mettre des polynômes!!! Si on utilise la fonction du calcul symbolique `REMAINDER` à la place de `MOD` dans les programmes précédents, `PGCD` (ou `PR`) peut alors avoir comme paramètres des polynômes à condition de remplacer les noms des variables `A, B, R` par `E1, E2, E3` et de changer le test d'arrêt.

```
PROMPT E1:
PROMPT E2:
WHILE DEGREE(E2) ≠ -1 REPEAT
REMAINDER(E1,E2) ->E3:
E2 ->E1:
E3 ->E2:
END:
DISP 4;"PGCD "E1:
FREEZE:
```

Vous entrez par exemple :
$\texttt{E1} = \texttt{S1}^2 - 1$ et $\texttt{E2} = \texttt{S1}^2 - 2 * \texttt{S1} + 1$ pour trouver le PGCD égal à `2*S1-2`.

7.2 Identité de Bézout

Dans ce paragraphe la fonction Bezout(A,B) renvoie la liste :
$\{U, V, PGCD(A, B)\}$ où U et V vérifient :
$A \times U + B \times V = PGCD(A, B)$.

7.2.1 Version itérative sans les listes

L'algorithme d'Euclide permet de trouver un couple U et V vérifiant :
$A \times U + B \times V = PGCD(A, B)$
En effet, si on note A_0 etB_0 les valeurs de A et de B du début on a :

$$\begin{array}{ll} A & = A_0 \times U + B_0 \times V \quad \text{avec} \quad U = 1 \quad \text{et} \quad V = 0 \\ B & = A_0 \times W + B_0 \times X \quad \text{avec} \quad W = 0 \quad \text{et} \quad X = 1 \end{array}$$

Puis on fait évoluer $A,\ B,\ U,\ V,\ W,\ X$ de façon que les deux relations ci-dessus soient toujours vérifiées.
Si :
$A = B \times Q + R \quad 0 \leq R < B \quad (R = A\ mod\ B\ et\ Q = E(A/B))$
On écrit alors :

$$\begin{array}{l} R = A - B \times Q = A_0 \times (U - W \times Q) + B_0 \times (V - X \times Q) = \\ \ A_0 \times S + B_0 \times T \quad \text{avec} \quad S = U - W \times Q \quad \text{et} \quad T = V - X \times Q \end{array}$$

Il reste alors à recommencer avec :
B dans le rôle de A (`B->A W->U  X->V`) et,
R dans le rôle de B (`R->B  S->W  T->X`)
d'où l'algorithme :

```
fonction Bezout (A,B)
local U,V,W,X,S,T,Q,R
1->U 0->V 0->W 1->X
tant que B ≠ 0 faire
A mod B->R
E(A/B)->Q
```

```
//R=A-B*Q
U-W*Q->S
V-X*Q->T
B->A W->U X->V
R->B S->W T->X
ftantque
résultat {U, V, A}
ffonction
```

7.2.2 Version itérative avec les listes

On peut simplifier l'écriture de l'algorithme ci-dessus en utilisant moins de variables : pour cela on utilise des listes LA LB LR pour mémoriser les triplets {U, V, A} {W, X, B} et {S, T, R}. Ceci est très commode car les calculatrices savent ajouter des listes de même longueur (en ajoutant les éléments de même indice) et savent aussi multiplier une liste par un nombre (en multipliant chacun des éléments de la liste par ce nombre).

```
fonction Bezout (A,B)
local LA LB LR
{1, 0, A}->LA
{0, 1, B}->LB
tant que LB[3] ≠ 0 faire
LA-LB*E(LA[3]/LB[3])->LR
LB->LA
LR->LB
ftantque
résultat LA
ffonction
```

7.2.3 Version récursive avec les listes

On peut définir récursivement la fonction Bezout par :

$Bezout(A, 0) = \{1,\ 0,\ A\}$

Si $B \neq 0$ il faut définir $Bezout(A, B)$ en fonction de $Bezout(B, R)$ lorsque

$R = A - B \times Q$ et $Q = E(A/B)$.

On a :

$$Bezout(B,R) = LT = \{W, X, pgcd(B,R)\}$$
$$\text{avec} \quad W \times B + X \times R = pgcd(B,R)$$

Donc :

$$\begin{aligned} W \times B + X \times (A - B \times Q) &= pgcd(B,R) \quad \text{ou encore} \\ X \times A + (W - X \times Q) \times B &= pgcd(A,B). \end{aligned}$$

D'où si $B \neq 0$ et si $Bezout(B,R) = LT$ on a :
$Bezout(A,B) = \{LT[2],\ LT[1] - LT[2] \times Q,\ LT[3]\}$.

```
fonction Bezout (A,B)
local LT Q R
Si B ≠ 0 faire
E(A/B)->Q
A-B*Q->R
Bezout(B,R)->LT
Résultat {LT[2], LT[1]-LT[2]*Q, LT[3]}
sinon Résultat {1, 0, A}
fsi
ffonction
```

7.2.4 Version récursive sans les listes

Si on utilise des variables globales pour `A B D U V T`, on peut voir la fonction `Bezour` comme calculant à partir de `A B`, des valeurs qu'elle met dans `U V D (AU+BV=D)` grâce à une variable locale `Q`.
On écrit donc une fonction sans paramètre : seule la variable `Q` doit être locale à la foncton alors que les autres variables `A, B ...` peuvent être globales.
`Bezour` fabrique `U, V, D` vérifiant `A*U+B*V=D` à partir de `A` et `B`. Avant l'appel récursif (on préserve `E(A/B)=Q` et on met `A` et `B` à jour (nouvelles valeurs), après l'appel les variables `U, V, D` vérifient `A*U+B*V=D` (avec `A` et `B` les nouvelles valeurs), il suffit alors de revenir aux premières valeurs de `A` et `B` en écrivant : `B*U+(A-B*Q)*V=A*V+B*(U-V*Q)`
On écrit alors :

```
Programme Bezour
local  Q
```

```
Si B ≠ 0 faire
E(A/B)->Q
A-B*Q->T
B->A
T->B
Bezour
U-V*Q->T
V->U
T->V
sinon
1->U
0->V
A->D
fsi
```

7.2.5 Traduction HP40G

-Version itérative avec les listes

On utilise ici aussi le programme IN qui permet de rentrer deux entiers A et B :

```
INPUT A;"A";;;1:
INPUT B;"B";;;1:
ERASE:
```

Puis on tape le programme BEZOUT :

```
RUN IN:
DISP 3;"BEZOUT "{A,B}:
{1,0,A} ->L1:
{0,1,B} ->L2:
WHILE L2(3) ≠ 0 REPEAT
L1-L2*FLOOR(L1(3)/L2(3)) ->L3:
L2 ->L1:
L3 ->L2:
END:
DISP 4;"U V PGCD "L1:
FREEZE:
```

-Version récursive sans les listes

On écrit le programme `BEZOUR`, grâce aux commandes (Merci Bernard!!!) :

`PUSH` (`PUSH(A)` pour mettre le contenu de `A` dans l'historique du `CAS`) et `POP` (pour récupérer les valeurs mises dans l'historique du `CAS`)
`PUSH` et `POP` permettent de simuler la variable locale `Q` On a remplacé dans la traduction (cf 7.2.4) les variables `R` et `W` par la variable temporaire `T`.

```
PROGRAM BEZOUR
IF B ≠ 0 THEN
PUSH (FLOOR(A/B)):
A MOD B->T:
B->A:
T->B:
RUN BEZOUR:
U-V*POP->T:
V->U:
T->V:
ELSE
 1->U:
 0->V:
 A->D:
END:
```

`PUSH (FLOOR(A/B))` a pour effet de mettre les différentes valeurs de `FLOOR(A/B)` sur une pile, et `POP` de les récuperer.
`T` est une variable auxillaire.
`BEZOUR` prend comme entrée les valeurs des variables globales `A` et `B` et remplit les variables globales `U` et `V` de façon que :
$\mathtt{A} \cdot \mathtt{U} + \mathtt{B} \cdot \mathtt{V} = \mathtt{PGCD}(\mathtt{A}, \mathtt{B})$.
On écrit ensuite le programme final `BEZOURT` permettant l'entrée de `A` et `B` et la sortie de `{U, V, D}`.

```
PROGRAM BEZOURT

PROMPT A:
PROMPT B:
RUN BEZOUR:
ERASE:
MSGBOX {U,V,D}:
```

Remarque :
Si on utilise la fonction de calcul symbolique `IREMAINDER` à la place de `MOD` et `IQUOT(A,B)` à la place de `FLOOR(A/B)` dans les programmes précédents, `BEZOUT` ou `BEZOUR` peut alors avoir comme paramètres des

entiers de Gauss à condition de remplacer les noms des variables `A, B, R...` par `Z1, Z2, Z3....`

REMARQUE :
Si on utilise la fonction du calcul symbolique `REMAINDER` à la place de `MOD` dans les programmes précédents, `BEZOUT` (ou `BEZOUR`) peut alors avoir comme paramètres des polynômes à condition de remplacer les noms des variables `A, B, R...` par `E1, E2, E3...` et de changer le test d'arrêt.

7.3 Décomposition en facteurs premiers

7.3.1 Les algorithmes et leurs traductions

- Premier algorithme

Soit N un entier.
On teste, pour tous les nombres D de 2 à N, la divisibilité de N par D.
Si D divise N, on cherche alors les diviseurs de N/D etc...N/D joue le rôle de N et on s'arrête quand $N = 1$
On met les diviseurs trouvés dans la liste FACT.

```
fonction facprem(N)
local D FACT
2 -> D
{} -> FACT
tant que N ≠ 1 faire
  si N mod D = 0 alors
    FACT + D -> FACT
    N/D -> N
   sinon
    D+1 -> D
 fsi
ftantque
résultat FACT
ffonction
```

- Première amélioration

On ne teste que les diviseurs D entre 2 et $E(\sqrt{N})$.
En effet si $N = D1 * D2$ alors on a :
soit $D1 \leq E(\sqrt{N})$, soit $D2 \leq E(\sqrt{N})$ car sinon on aurait :
$D1 * D2 \geq (E(\sqrt{N}) + 1)^2 > N$.

```
fonction facprem(N)
local D FACT
2 -> D
{} -> FACT
tant que D*D ≤ N faire
  si N mod D = 0 alors
    FACT + D -> FACT
    N/D-> N
   sinon
    D+1 -> D
 fsi
ftantque
FACT + N -> FACT
résultat FACT
ffonction
```

- Deuxième amélioration
On cherche si 2 divise N, puis on teste les diviseurs impairs D entre 3 et $E(\sqrt{N})$.
Dans la liste FACT, on fait suivre chaque diviseur par son exposant : decomp(12)={2,2,3,1}.

```
fonction  facprem(N)
local K D FACT
{}->FACT
0 -> K
tant que N mod 2 = 0 faire
    K+1 -> K
    N/2 -> N
ftantque
si K ≠0 alors
    FACT + {2 K} -> FACT
fsi
3 ->D
tant que D*D ≤ N faire
  0 -> K
  tant que N mod D = 0 faire
    K+1 -> K
    N/D -> N
  ftantque
```

```
  si K ≠0 alors
    FACT + {D K} -> FACT
  fsi
  D+2 -> D
ftantque
si N ≠ 1 alors
FACT + {N 1} -> FACT
fsi
résultat FACT
ffonction
```

7.3.2 Traduction HP40G

On traduit le dernier algorithme.
La HP40G ne connait pas la liste {}, donc pour initialiser L1 avec la liste vide on écrit : SYSEVAL 259589.
Voici le programme FACTPREM :

```
INPUT N;"N";;;1:
ERASE:
0 ->K:
SYSEVAL 259589:
WHILE N MOD 2 == 0 REPEAT
1+K -> K:
N/2 -> N:
END:
IF K ≠ 0 THEN
{2,K} ->L1:
END:
3 ->D:
WHILE D*D ≤ N REPEAT
0 -> K:
WHILE N MOD D == 0 REPEAT
K+1 -> K:
N/D -> N:
END:
IF K ≠ 0 THEN
CONCAT (L1,{D,K}) -> L1:
END:
```

```
2+D -> D:
END:
IF N ≠ 1 THEN
CONCAT (L1, {N,1}) -> L1:
END:
DISP 3; "FACT" L1:
FREEZE:
```

7.4 Calcul de A^P *mod* N

7.4.1 Traduction Algorithmique

-Premier algorithme
On utilise deux variables locales PUIS et I.
On fait un programme itératif de façon qu'à chaque étape PUIS représente A^I (mod N).

```
fonction puismod (A, P, N)
local PUIS, I
1->PUIS
pour I de 1 a P faire
  A*PUIS mod N ->PUIS
fpour
resultat PUIS
ffonction
```

-Deuxième algorithme
On utilise une seule variable locale PUI mais on fait varier P de façon qu'à chaque étape de l'itération on ait :
$resultat = PUI * A^P$ (mod N)

```
fonction puismod (A, P, N)
local PUI
1->PUI
tant que  P>0  faire
  A*PUI mod N ->PUI
  P-1->P
ftantque
resultat PUI
ffonction
```

-Troisième algorithme
On peut aisément modifier ce programme en remarquant que :

$A^{2*P} = (A*A)^P$.
Donc quand P est pair on a la relation :
$PUI * A^P = PUI * (A*A)^{P/2} \pmod N$
et quand P est impair on a la relation :
$PUI * A^P = PUI * A * A^{P-1} \pmod N$.
On obtient alors un algorithme rapide de $A^P \pmod N$.

```
fonction puismod (A, P, N)
local PUI
1->PUI
tant que  P>0  faire
  si P mod 2 =0 alors
    P/2->P
    A*A mod N->A
  sinon
    A*PUI mod N ->PUI
    P-1->P
  fsi
ftantque
resultat PUI
ffonction
```

On peut remarquer que si P est impair, P-1 est pair.
On peut donc écrire :

```
fonction puismod (A, P, N)
local PUI
1->PUI
tant que  P>0  faire
  si P mod 2 =1 alors
    A*PUI mod N ->PUI
    P-1->P
  fsi
P/2->P
A*A mod N->A
ftantque
resultat PUI
ffonction
```

7.4.2 Traduction HP40G

Le calcul de A^p mod N est utilisé dans le programme de la méthode probabiliste de Mr Rabin. On se reportera donc à ce sous-programme pour la traduction (cf 7.6).

7.5 La fonction "estpremier"

7.5.1 Traduction Algorithmique

- Premier algorithme

On va écrire un fonction booléenne de paramètre N, qui sera égale à VRAI quand N est premier et à FAUX sinon.

Pour cela, on cherche si N posséde un diviseur $\neq 1$ et $\leq$ à $E(\sqrt{N})$ (partie entière de racine de N).

On traite le cas N=1 à part !

On utilise une variable booléenne PREM, qui est au départ à VRAI, et qui passe à FAUX dès que l'on rencontre un diviseur de N.

```
Fonction estpremier(N)
local PREM, I, J
E(√N)-> J
Si N = 1 alors
  FAUX->PREM
  sinon
  VRAI->PREM
fsi
2->I
tant que PREM et I ≤J faire
  si N mod I = 0 alors
     FAUX->PREM
     sinon
     I+1->I
  fsi
ftantque
résultat PREM
ffonction
```

- Première amélioration

On peut remarquer que l'on peut tester si N est pair, et sinon regarder si N possède un diviseur impair.

```
Fonction estpremier(N)
local PREM, I, J
```

```
E(√N)- > J
Si (N = 1) ou (N mod 2 = 0) et (N≠2) alors
  FAUX->PREM
  sinon
  VRAI->PREM
fsi
3->I
tant que PREM et I ≤J faire
  si N mod I = 0 alors
     FAUX->PREM
     sinon
     I+2->I
  fsi
ftantque
résultat PREM
ffonction
```

- Deuxième amélioration

On regarde si N est divisible par 2 ou par 3, sinon on regarde si N posséde un diviseur de la forme $6 \times k - 1$ ou $6 \times k + 1$.

```
Fonction estpremier(N)
local PREM, I, J
E(√N)- > J
Si (N = 1) ou (N mod 2 = 0) ou ( N mod 3 = 0) alors
  FAUX->PREM
  sinon
  VRAI->PREM
fsi
si N=2 ou N=3 alors
VRAI->PREM
fsi
5->I
tant que PREM et I ≤J faire
  si (N mod I = 0) ou (N mod I+2 =0) alors
     FAUX->PREM
     sinon
     I+6->I
  fsi
ftantque
```

```
résultat PREM
ffonction
```

7.5.2 Traduction HP40G

```
INPUT N;"N";;;1:
IF N MOD 2== 0 OR N MOD 3==0 OR N==1 THEN
0 ->P:
ELSE
1->P:
END:
IF N==2 OR N==3 THEN
1->P:
END:
5->I:
FLOOR(√N)- > J :
WHILE I ≤ J AND P REPEAT
IF N MOD I==0 OR N MOD I+2==0 THEN
0 ->P:
ELSE
I+6 ->I:
END:
END:
ERASE:
DISP 5;P:
FREEZE:
```

7.6 Méthode probabiliste de Mr Rabin

Si N est premier alors tous les nombres K strictement inférieurs à N sont premiers avec N, donc d'après le petit théorème de Fermat on a :

$K^{N-1} = 1 \pmod N$

Si N n'est pas premier, les entiers K vérifiant :

$K^{N-1} = 1 \pmod N$

sont très peu nombreux.

Plus précisement on peut montrer que si $N > 4$, la probabilité d'obtenir un tel nombre K est inférieure à 0.25.

Un nombre N vérifiant $K^{N-1} = 1 \pmod N$ pour 20 tirages de K

est un nombre pseudo-premier. La méthode probabiliste de Rabin consiste à tirer au hasard un nombre K $(1 < K < N)$ et à calculer :

$K^{N-1} \pmod N$

Si $K^{N-1} = 1 \pmod N$ on refait un autre tirage et si $K^{N-1} \neq 1 \pmod N$ on est sûr que N n'est pas premier.

Si on obtient $K^{N-1} = 1 \pmod N$ pour 20 tirages de K on peut conclure que N est premier avec une probabilité d'erreur très faible inférieure à 0.25^{20} soit de l'ordre de 10^{-12}.

Bien sûr cette méthode est employée pour savoir si de grands nombres sont pseudo-premiers.

7.6.1 Traduction Algorithmique

On suppose que :

`Hasard(N)` donne un nombre entier au hasard entre 0 et $N-1$.

Le calcul de :

$K^{N-1} \bmod N$

se fait grâce à l'algorithme de la puissance rapide (cf page 137).

On notera :

`puismod(K, P, N)` la fonction qui calcule $\mathtt{K}^{\mathtt{P}}$ mod `N`

```
Fonction estprem(N)
local K, I, P
1->I
1->P
Tant que P = 1 et I < 20 faire
hasard(N-2)+2->K
puismod(K, N-1, N)->P
I+1->I
ftantque
Si P =1 alors
resultat VRAI
sinon
resultat FAUX
fsi
ffonction
```

7.6.2 Traduction HP40G

```
PROMPT N:
RANDSEED TIME:
```

```
1->I:
1->P:
WHILE I < 20 AND P==1 REPEAT
  FLOOR( RANDOM * (N-2))+2->K:
  N-1->M:
  @ Calcul de K puissance M mod N dans P.
  1->P:
  WHILE  0 < M REPEAT
    IF M MOD 2 == 0 THEN
      M / 2 -> M :
      (K * K)  MOD N ->K :
    ELSE
      K*P MOD N -> P:
      M - 1 -> M:
    END:
  END:
  @ P contient K puissance M mod N et M=N-1.
  I+1 ->I:
END:
ERASE:
IF P==1 THEN
  DISP 3;"PREMIER " N:
ELSE
  DISP 3;"NON PREMIER " N:
END:
FREEZE:
```

Remarque :
On peut aussi utiliser la fonction de calcul formel `POWMOD` et on écrit alors :

```
MODSTO(N) :
POWMOD(K,N-1) STO▷ P :
```

à la place des instructions comprises entre les @ on obtient :

```
PROMPT N:
RANDSEED TIME:
1->I:
1->P:
WHILE I < 20 AND P==1 REPEAT
  FLOOR( RANDOM * (N-2))+2->K:
 MODSTO(N):
```

```
POWMOD(K,N-1) STO$\triangleright$ P:
  I+1 ->I:
END:
ERASE:
IF P==1 THEN
  DISP 3;"PREMIER " N:
ELSE
  DISP 3;"NON PREMIER " N:
END:
FREEZE:
```

Chapitre 8

GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom : to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation : a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals ; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts : Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version :

* A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission. * B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five). * C. State on the Title page the name of the publisher of the Modified Version, as the publisher. * D. Preserve all the copyright notices of the Document. * E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices. * F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below. * G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice. * H. Include an unaltered copy of this License. * I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence. * J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission. * K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein. * L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. * M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version. * N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties-for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document. If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http ://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page :

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel

under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Index

Table des matières