i.r.e.m.
UNIVERSITE PARIS VII

TRAVAUX DIRIGES DE MATHEMATIQUES SUR MICRO-ORDINATEURS EN DEUG SSM

2ème partie

Par PIERRE JARRAUD

cahier de didactique des mathematiques
numéro
OCTOBRE 1987
45

INTRODUCTION

Ce cahier présente la suite de l'introduction de Travaux Dirigés de Mathématiques sur micro-ordinateurs dans la Section Expérimentale de DEUG SSM de l'Université Pierre et Marie Curie (Paris 6) dans le cadre du contrat de rénovation pédagogique n° 18-14-01 du Ministère de l'Education. La première partie avait fait l'objet d'un précédent Cahier de Didactique des Mathématiques[1].

La seconde partie conserve les mêmes principes: logiciels prêts à l'emploi, pas de programmation par les étudiants, Une amélioration est, suivant l'exemple de l'équipe de Lille (M.Rogalski, C.Sacré..), l'introduction de **feuilles de travaux dirigés** pour mieux guider les étudiants durant les séances sur machine.

Un autre fait marquant est l'utilisation, à côté de langages "classiques" comme Pascal ou Basic qui font essentiellement du **calcul approché sur des réels**, de langages comme **Mu-Math**[2] (ou **PC-Reduce**[3] l'an prochain) qui permettent du **calcul formel sur des objets algébriques** ou du calcul exact sur des rationnels, ce qui est très important pour la perception par les étudiants des possibilités des micro-ordinateurs, ce sentiment étant d'ailleurs renforcé par l'introduction de calulatrices de poche ayant des possibilités de calcul formel (comme la Hewlett-Packard HP 28C). Sauf dans la cas du programme "INTEGRAL" de P.Fortin nous n'avons pas utilisé les possibilités de programmation de Mu-Math et nous avons utilisé les machines comme des calculatrices formelles.

La fin de l'expérience (en particulier l'étude des équations différentielles) a été écourtée à cause du retard pris pendant les grèves des étudiants et des transports.

[1] C.LAURENT et P.JARRAUD
Travaux Dirigés de Mathématiques sur Micro-ordinateurs en DEUG SSM
Cahier de Didactique des Mathématiques n°35 (IREM PARIS-SUD Janvier 87)

[2] © Software House Licence Microsoft

[3] © The Rand Corporation

INTEGRATION

Il y a eu deux séances sur machine, l'une avec le logiciel "INTEGRAL" de P. Fortin, l'autre avec mon logiciel "INTEGRALE"

Le calcul formel des primitives

Nous avons utilisé - dans une version expérimentale[4] - le logiciel développé à l'IREM de Rouen par P. Fortin[5]. Ce logiciel, très original, utilise les possibilités de calcul formel du langage MU-MATH pour faire une séance de calcul de primitives assisté par ordinateur: on fournit à l'étudiant une liste de fonctions dont il faut calculer une primitive. Pour chaque fonction l'étudiant cherche d'abord sur une feuille de papier une primitive par les moyens qu'il connaît et propose son résultat à la machine qui vérifie s'il est correct. Si la réponse est bonne, on passe à la fonction suivante sinon l'ordinateur propose des indications de méthode (linéarisation, changement de variable intégration par parties, etc..) de plus en plus précises jusqu'à ce que l'étudiant arrive au résultat correct.

L'important est que cette séquence d'aide et de corrigé est entièrement gérée par le logiciel, l'enseignant n'ayant qu'à fournir les énoncés à la machine (et éventuellement des indications de calcul). Les énoncés sont choisis librement dans les limites du programme de la classe et des possibilités du logiciel. Le logiciel conserve en mémoire les réponses de l'étudiant ce qui permet à l'enseignant de surveiller le travail de celui-ci.

Voici la feuille remise aux étudiants:

[4] La version définitive est maintenant sortie et commercialisée par FIL.

[5] P. FORTIN:

Université de Rouen Haute Normandie
IREM 1 rue T. Becket BP 27 76130 MONT SAINT AIGNAN

Séance sur machine

Calcul de primitives avec le logiciel INTEGRAL de P.Fortin

(extraits de la notice de présentation du logiciel)

Principales caractéristiques

Le système SAIT calculer des primitives, il est donc inutile de lui indiquer par avance les réponses attendues. Il peut ainsi traiter des exercices totalement extérieurs au programme.

En cas d'insuccès de la part de l'élève, le programme peut entièrement guider ce dernier, en lui indiquant, quand cela est nécessaire, les méthodes à utiliser.

Si la réponse de l'élève demeure incorrecte, le message d'erreur affiché à l'écran précise la nature de celle-ci (erreur de signe, de coefficient, oubli d'une valeur absolue, etc...).

Il est possible d'obtenir la correction complète d'un exercice. Les méthodes d'intégration ou les résultats utilisés (dérivées usuelles, formules de linéarisation ...) sont alors entièrement détaillés. Rappelons que cette correction est entièrement déterminée par le système.

Mise en route du programme:

Mettre la disquette dans la machine, allumer la machine et après l'arrivée du A\> du DOS entrer la commande:

>LANCE <return>

Le programme est alors lu sur la disquette. On obtient l'affichage d'un texte indiquant les règles d'écriture à utiliser puis d'un menu indiquant les commandes disponibles.

Il est possible à tout instant de revenir à ce menu :

- en appuyant sur la touche <ESC> pendant un calcul,
- en répondant <ESC> <return> ou MENU <return> à une question.

Commandes disponibles :

externe: Cette commande permet de rechercher des primitives de fonctions proposées par des sources externes au programme. L'élève doit alors proposer une réponse.

1/Si cette réponse est correcte on passe à l'exercice suivant.
Répondre fin ; <return>
s'il n'y a pas d'autre exercice à faire.

2/Si l'élève ne sait pas faire l'exercice, il lui suffit de répondre ? <return> pour obtenir de l'aide. Cette possibilité lui est offerte à tous les niveaux du programme. Cette aide sera progressive. Cette possibilité permet également d'obtenir une décomposition du calcul (pour une intégration par parties par exemple).
3/Si la réponse est incorrecte, la question sera reposée en offrant à l'utilisateur la possibilité de demander de l'aide.

<u>serie</u>: Cette commande permet à l'utilisateur de chercher une série d'exercices préalablement enregistrée.
1/L'utilisateur doit indiquer le code de la série désirée.
2/Les exercices sont alors automatiquement proposés. Le mode de réponse est le même qu'avec la fonction précédente;

<u>fin</u>: Sortie du programme et retour à MSDOS.

<u>Entrées au clavier</u>

Toute entrée au clavier doit se terminer par la séquence ; <return>

exemple : sin x + cos x ; <return>
--> sin x + cos x

Le <return> seul fait simplement passer à la ligne suivante .Il est alors possible de poursuivre l'introduction de l'expression

exemple: sin x <return>
+ cos x ; <return>
--> sin x + cos x

Tant que le ; n'a pas été frappé, le système est en attente de la suite de l'expression. Si vous l'oubliez avant de taper sur la touche <return>, le curseur se positionne sur le début de la ligne suivante. Il vous est alors possible de conclure l'introduction par l'entrée de la séquence ; <return>

exemple :
sin x + cos x <return>
; <return>
--> sin x + cos x

<u>Règles de syntaxe</u>

<u>PRODUITS</u>: A x B se note A * B. ex : x*cos x +(x+1)*(x+2);
si A est un nombre, le * est inutile. ex : 5x + 4b;

<u>QUOTIENTS</u>: $\dfrac{A}{B}$ s'écrit A/B. ex : $\dfrac{4}{x}$ --> 4/x;

attention au parenthèsage : $\dfrac{x+2a\ b}{x-3y\ z}$ --> (x+2a b)/(x-3y z);

<u>PUISSANCES</u>: a^b s'écrit a^b; <u>RACINES CARREES</u>: $\sqrt{x}$: rac(x) ou x^(1/2)

syntaxe des autres fonctions :

sinus	sin x , sin (x) , sin (2x) , sin (x+3)
cosinus	cos x , cos (x) , cos (x^2), (cos x)^2 ,
tangente	tg x
cotangente	cotg x

attention : cos 2x --> (cos 2) * x !!!

log népérien	ln x , ln (x y) , ln (x+2y),
log decimal	log x ,
log base a	log (x,a) , log (2x+3,5) ,
valeur absolue	abs x , abs(2x+1) ,

constantes disponibles   pi et e

Aide pendant un calcul:

Si vous ne connaissez pas la réponse à une question, ou si vous souhaitez décomposer un calcul, il suffit de répondre
? <return>

Aide pour l'utilisation de ce programme :
Pour obtenir la documentation complète du programme, il suffit de composer la commande (accessible depuis le menu principal)
doc <return>

Retour au menu :
Pendant l'exécution d'un calcul : appuyer sur la touche <ESC>.
En réponse à une question : menu <return>.

Exercices: faire les séries **C** *et* **E** *et quelques calculs de primitives de votre choix.*

Les deux séries de fonctions dont il fallait calculer les primitives étaient les suivantes:

Série C:

f(x) = ln x

f(x) = x ln x

f(x) = x * rac (x + 2)

f(x) = 2 x cos x

f(x) = (x^2 + 1) sin x

f(x) = 1 / (x^2 - 2 x)

f(x) = (3 x^2 + x - 1) / (x + 2)

f(x) = (x^2 - 1) / (x^3 - 3 x)

Série E:

f(x) = sin (10 x) sin (15 x)

f(x) = (-rac x + x + 1) (rac x + 1)

f(x) = x cos (3 x)

f(x) = (x^2 + 5 x + 6) cos (2 x)

f(x) = x^3/(a^2 - x^2)

f(x) = tg (rac x) / rac x

f(x) = sin x cos x / rac ((cos x)^2- (sin x)^2)

f(x) = 1/(x^4 + x^7)

Il y a **deux limites** à l'utilisation de ce logiciel: d'une part il a été conçu pour des élèves de terminale C et bridé volontairement pour n'utiliser que les fonctions figurant au programme de cette classe (par exemple, il ne "sait" pas intégrer $1/(1+x^2)$ alors que MU-MATH le sait). Il est tout à fait possible de lever cette limitation en modifiant légèrement le programme. Un autre problème est que, faute de **salle de micro-ordinateurs en libre-service**, nous avons dû utiliser le logiciel pendant une séance habituelle de Travaux Dirigés sous la surveillance de l'enseignant habituel qui aurait très bien pu faire lui-même (et aussi bien) le travail de l'ordinateur. Et l'on ne profite pas de la liberté que le travail sur machine sans surveillance de son enseignant pouurait donner à l'étudiant.

Illustration de l'intégrale de Riemann

Le point de vue est tout à fait différent, il s'agit au moyen d'un programme écrit en Pascal d'illustrer la théorie de l'intégrale de Riemann telle qu'elle est enseignée en cours. En 1985-1986, faute de matériel nous nous étions contentés d'une présentation en cours avec manipulation de la machine par l'enseignant. Cette année nous avons laissé les étudiants utiliser la machine eux-mêmes en les guidant avec une feuille de travaux pratiques.

Nous avons utilisé mon programme"INTEGRALE" présenté en détail dans le cahier 27[6]. dans une version pour micro-ordinateur IBM-PC (ou compatible) muni d'un écran graphique (type CGA). La structure générale est la même, l'utilisation voisine. La réalisation en Turbo-Pascal de Borland a permis d'augmenter la vitesse d'éxécution. La définition est meilleure (deux fois plus de points horizontalement) et l'affichage est plus lisible et plus complet; par contre l'entrée directe d'une fonction au clavier comme chaîne de caractères a nécessité l'utilisation d'un analyseur syntaxique dû à R.Rolland (CIRM Marseille) que je tiens à remercier.

Une autre amélioration est l'introduction d'une méthode "probabiliste" à la "Monte-Carlo" qui m'a été suggérée par V.Gautheron: supposons pour simplifier la fonction positive, grace à la fonction RANDOM de la machine on tire des points au hasard sur l'écran et on compte le pourcentage de ceux entre le graphe de la fonction et l'axe y=0, quand le nombre d'essais augmente ce pourcentage tend vers le rapport de la valeur de l'intégrale à l'aire de la surface représentée à l'écran.

Si l'on a pris la précaution de charger le fichier GRAPHICS.COM une recopie d'écran sur papier est possible en utilisant les touches SHIFT + PRINTSCREEN. Des exemples en sont donnés dans les pages suivantes.

On trouvera le programme dans sa version compilée INTEGRAL.COM et le fichier source INTEGRAL.PAS dans la disquette d'accompagnement de ce cahier (voir à la fin).

6 P.JARRAUD
Riemann, Darboux, Basic.Illustration de l'intégrale sur micro-ordinateur
Cahier de Didactique des Mathématiques n° 27 IREM Paris-Sud

Dans le premier cas il suffit, sous DOS après avoir éventuellement allumé l'écran graphique par MODE CO80 et chargé GRAFTABL pour avoir les accents, de frapper INTEGRAL et de valider par ENTER.

Dans le second cas d'éventuelles modifications sont plus faciles mais il faut disposer du Turbo-Pascal dans la version 3.0 (avec les librairies graphiques GRAPH.P et GRAPH.BIN) et de l'analyseur syntaxique FONCTION.LIB de R.Rolland.

Les fonctions reconnues par l'analyseur sont expliquées dans la feuille remise aux étudiants (voir ci-dessous).

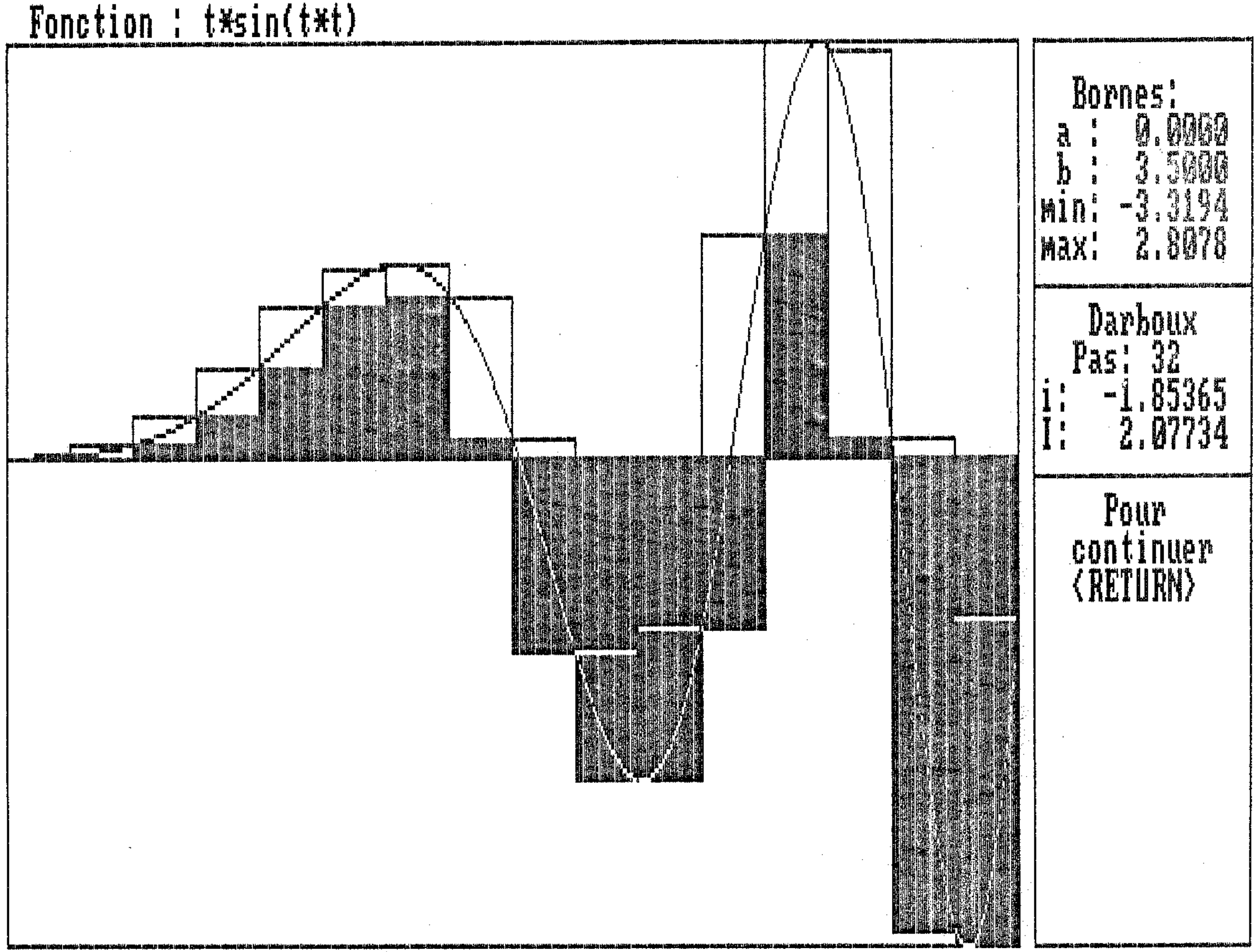
On trouvera dans les pages suivantes des exemples d'écran et le feuilles remises aux étudiants pour les guider pendant la séance d'utilisation du logiciel.

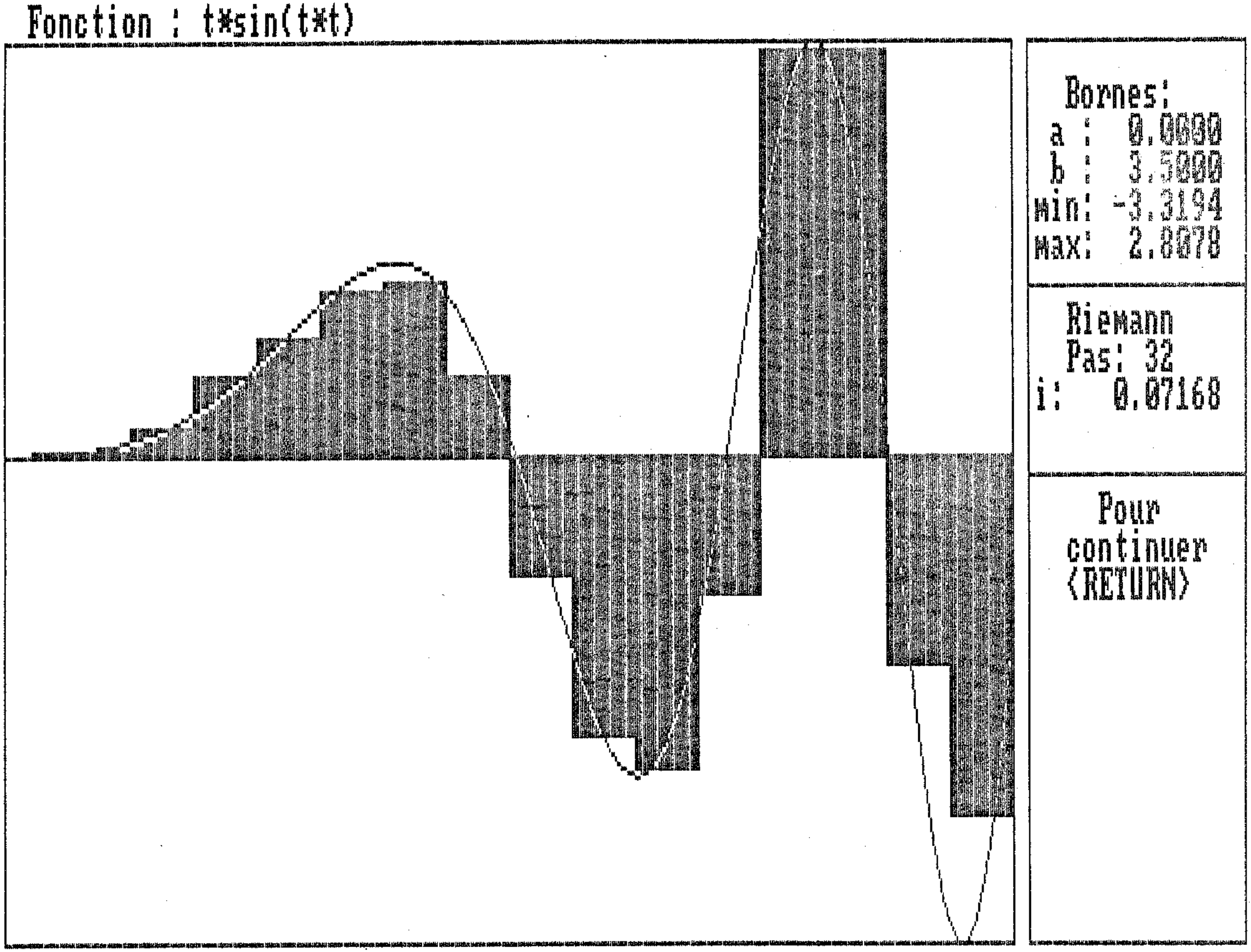
Fonction : t*sin(t*t)

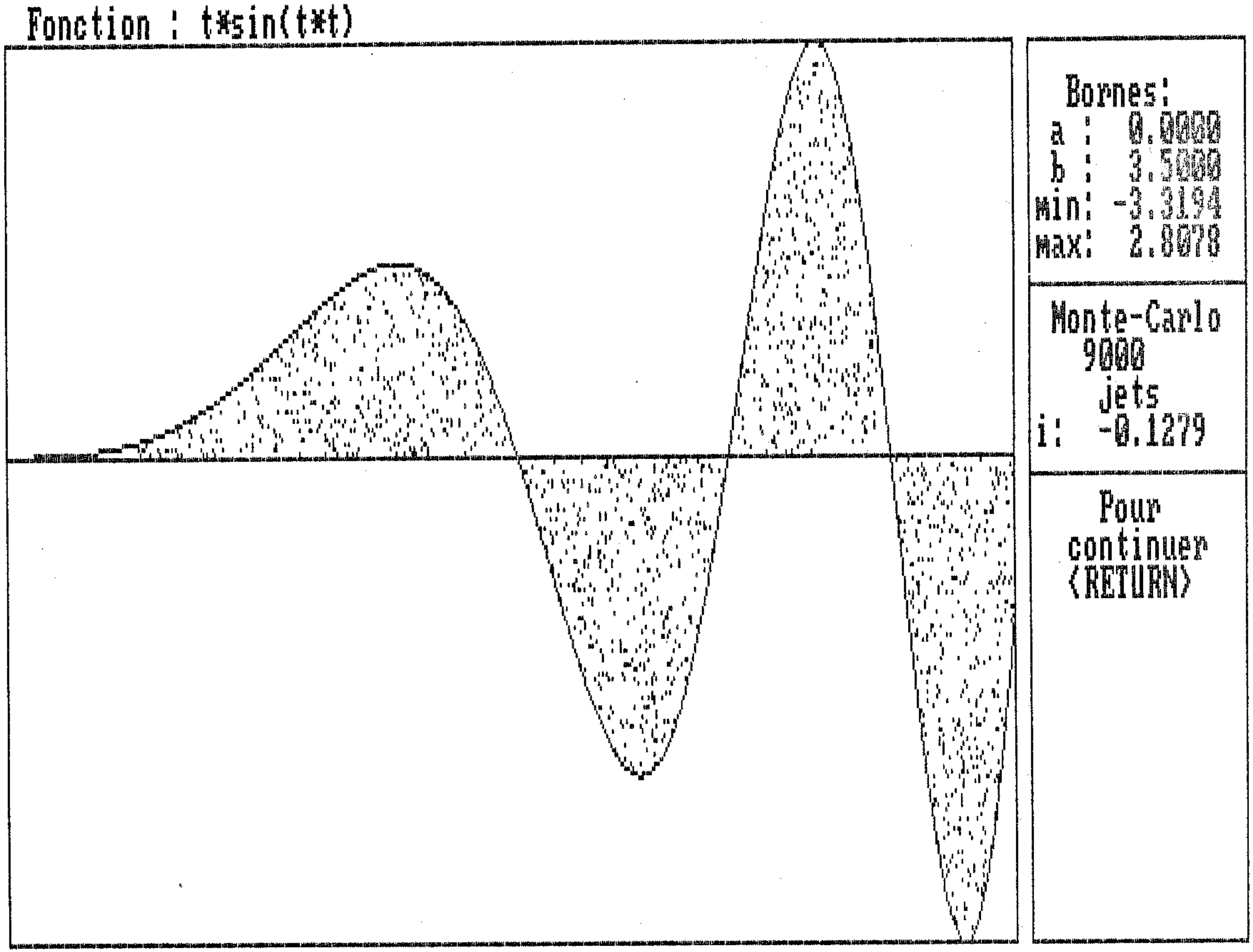
MENU

0 Fin
1 Approximation par des rectangles
2 Sommes de Darboux
3 Sommes de Riemann
4 Methode des trapezes
5 Methode de Simpson
6 Methode de Monte-carlo
7 Changement de bornes
8 Choix d'une nouvelle option
9 Nouvelle fonction

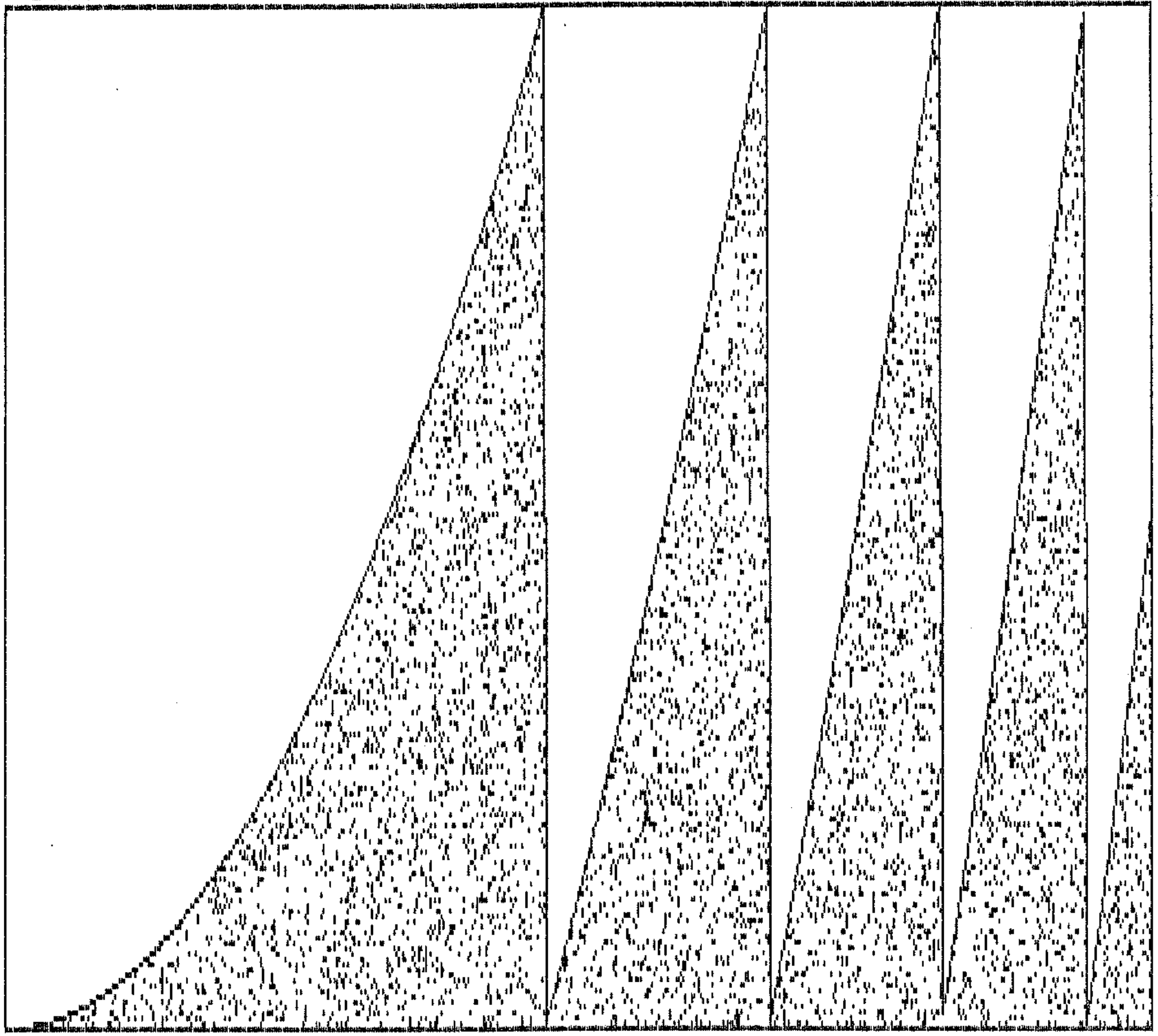
Bornes:
a : 0.0000
b : 3.5000
min: -3.3194
max: 2.8078


Fonction : t*sin(t*t)
Bornes:
a : 0.0000
b : 3.5000
min: -3.3194
max: 2.8078
Darboux
Pas: 32
i: -1.85365
I: 2.07734
Pour
continuer
<RETURN>


Fonction : t*sin(t*t)
Bornes:
a : 0.0000
b : 3.5000
min: -3.3194
max: 2.8078
Riemann
Pas: 32
i: 0.07168
Pour
continuer
<RETURN>


Fonction : t*sin(t*t)
Bornes:
a : 0.0000
b : 3.5000
min: -3.3194
max: 2.8078
Monte-Carlo
9000
jets
i: -0.1279
Pour
continuer
<RETURN>

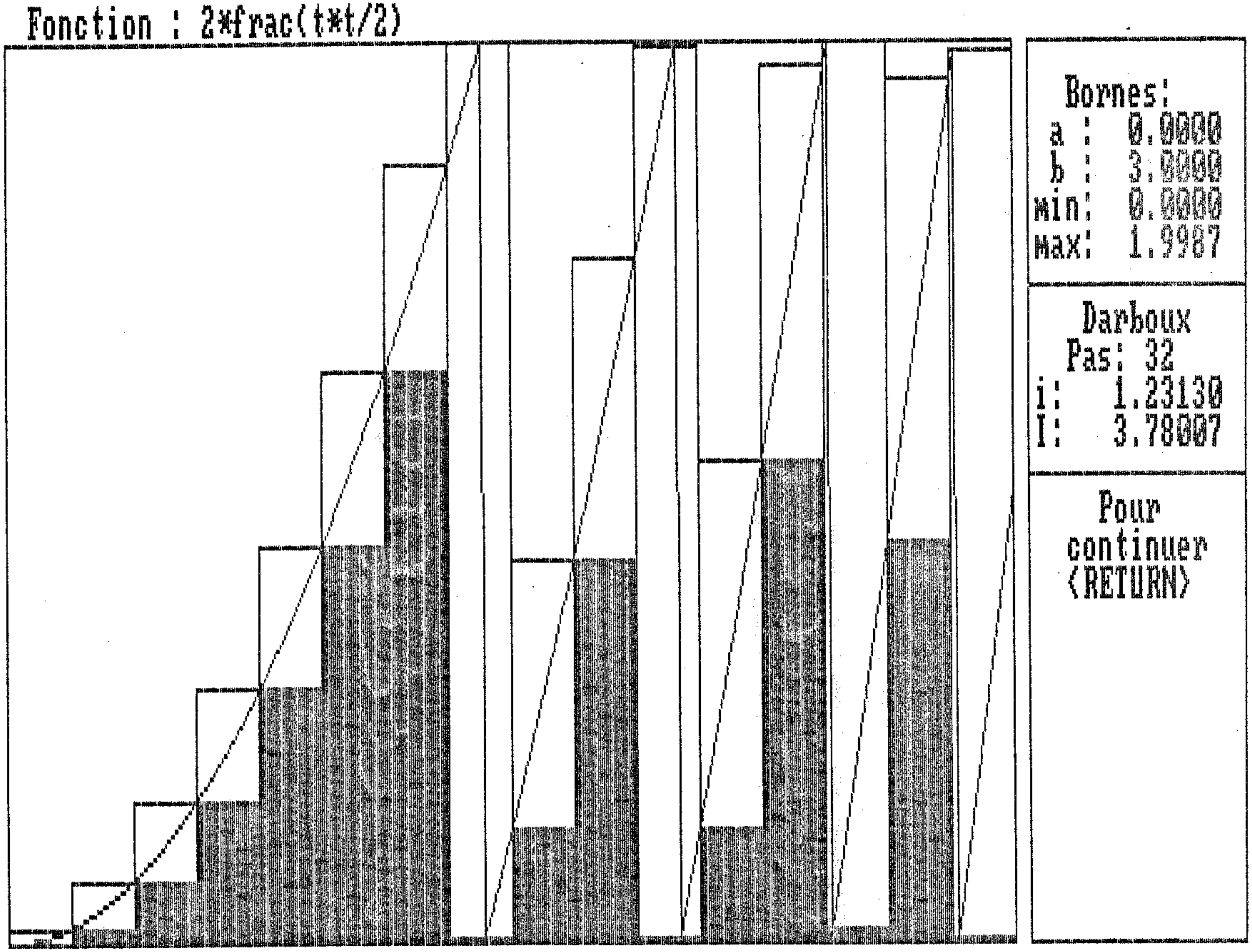
Fonction : 2*frac(t*t/2)



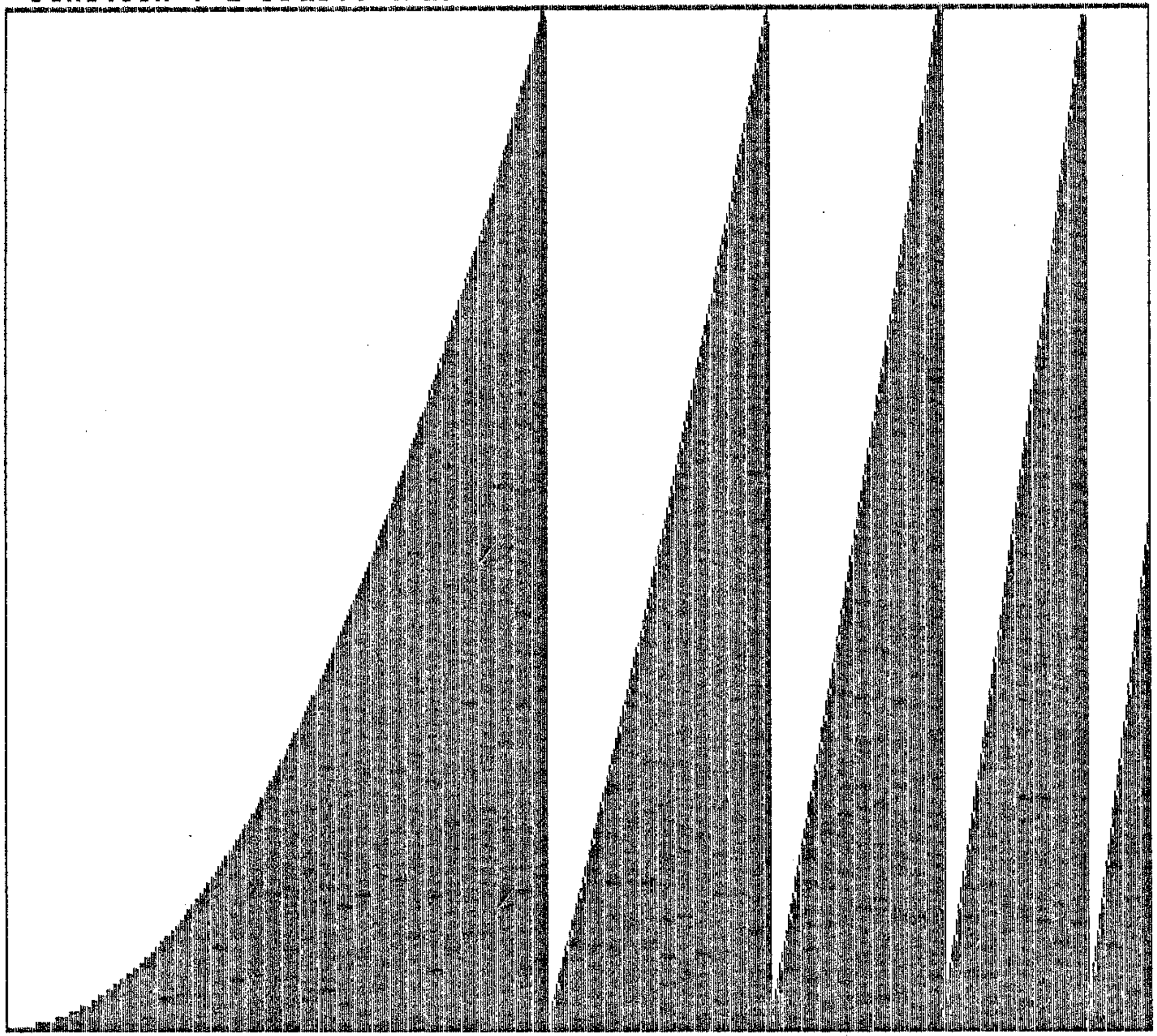
Bornes:
a : 0.0000
b : 3.0000
min: 0.0000
max: 1.9987

Monte-Carlo
16000
jets
i: 2.3774

Pour
continuer
<RETURN>


Fonction : 2*frac(t*t/2)
Bornes:
a : 0.0000
b : 3.0000
min: 0.0000
max: 1.9987
Darboux
Pas: 32
i: 1.23130
I: 3.78007
Pour
continuer
<RETURN>

Fonction : 2*frac(t*t/2)



Bornes:
a : 0.0000
b : 3.0000
min: 0.0000
max: 1.9987

Rectangles
Pas: 2
i: 2.38483

Pour
continuer
<RETURN>

Séance sur machine
Utilisation du logiciel "INTEGRALE"

1) Insérer la disquette "DEVLIM-INTEGRALE" dans le lecteur A et allumer l'appareil. Après l'apparition du **A\>** du MS-DOS taper **"INTEGRALE"** puis ↵. Le logiciel démarre.

2) Introduction de la fonction à intégrer: grace à l'analyseur syntaxique de Modulog, on l'introduit comme une chaîne de caractères dans la fenêtre principale, la variable doit **impérativement** s'appeler **t**. Les chaînes correctes sont les expressions formées avec les opérations +, -, *, /, ^(le signe ^ désignant l'exponentiation), les fonctions:

ABS	valeur absolue
ACH	argument cosinus hyperbolique
ACOS	arc cosinus
ASH	argument sinus hyperbolique
ASIN	arc sinus
ATAN	arc tangente
ATH	argument tangente hyperbolique
CH	cosinus hyperbolique
COS	cosinus
EXP	exponentielle
FRAC	partie fractionnaire
INT	partie entière
LN	logarithme népérien
SH	sinus hyperbolique
SIN	sinus
SQR	carré
SQRT	racine carrée
TG	tangente
TH	tangente hyperbolique

La fonction s'affiche en haut de l'écran.

3) Introduction des bornes d'intégration: dans la fenêtre en bas à droite de l'écran, sous forme de **nombre décimal** (1.414 et non $\sqrt{2}$). Valider par ↵. Les valeurs choisies, puis après un certain temps de calcul, le minimum et le maximum de la fonction sur l'intervalle choisi s'affichent en haut à droite.

4) Choix du cadrage et d'une méthode d'intégration: dans la fenêtre principale: taper un chiffre (sans valider par ↵).

5) Choix du pas: la fenêtre graphique a une largeur de 512 et le pas est la largeur de la subdivision, ce doit être une puissance de 2: 2, 4,.., 128, l'entrée se fait dans la fenêtre du milieu à droite (valider par ↵).

6) Les résultats s'affichent dans la fenêtre du milieu à droite.

7) Pour continuer: ↵.

$$I_1 = \int_0^1 e^t \, dt \qquad \text{(entrer exp(t))}$$

Valeur calculée:

Option de cadrage:2

Rectangles: pas 128: pas 32: pas 4:

Darboux:

pas	128	32	8	4
i				
I				

Riemann (pas 64):

1e essai: 2e essai: 3e essai:

Trapèzes: pas 64: pas 8: pas 4:

Simpson: pas 64 pas 8:

Monte-Carlo (10 000 jets)

1e essai: 2e essai: 3e essai:

Commentaires:

$$I_2 = \int_0^3 \sqrt{1+t^3} \, dt \qquad \text{(entrer sqrt(1+t^3))}$$

Savez-vous calculer exactement I_2?

Option de cadrage:2

Rectangles: pas 64: pas 16: pas 4:

Darboux:

pas	128	32	8	4
i				
I				

Riemann (pas 32):

1e essai: 2e essai: 3e essai:

Trapèzes: pas 64: pas 8: pas 4:

Simpson: pas 16: pas 4:

Monte-Carlo (10 000 jets)

1e essai: 2e essai: 3e essai:

Commentaires:

$$I_3 = \int_0^3 (t^2 \bmod 2)\, dt$$ (entrer 2*frac(t*t/2))

Valeur calculée:

Option de cadrage: 1 ou 2

Rectangles: pas 64: pas 16: pas 4:

Darboux:

pas	64	16	8	2
i				
I				

Riemann (pas 32):

1e essai: 2e essai: 3e essai:

Trapèzes: pas 64: pas 8: pas 4:

Simpson: pas 64: pas 8: pas 2:

Monte-Carlo (10 000 jets)

1e essai: 2e essai: 3e essai:

Commentaires:

$$I_4 = \int_{-0.2}^{0.2} t\cos(1/t)\, dt$$ (entrer t*cos(1/t))

Bornes: -0.2 et 0.20001 pour éviter un problème en 0.

Valeur calculée:

Option de cadrage: 1 puis 5

Rectangles: pas 64: pas 16: pas 4:

Darboux:

pas	128	32	8	4
i				
I				

Riemann (pas 32):

1e essai: 2e essai: 3e essai:

Trapèzes: pas 64: pas 16: pas 4:

Simpson: pas 32 pas 4:

Monte-Carlo (??jets):

1e essai: 2e essai: 3e essai:

Commentaires:

Libre choix:

$$I_5 = \int$$

Valeur calculée:

Option de cadrage:

Rectangles: pas ??: pas ??: pas ??:

Darboux:

pas	??	??	??	??
i				
I				

Riemann (pas ??):

1e essai: 2e essai: 3e essai:

Trapèzes: pas ??: pas ?: pas ?:

Simpson: pas ??: pas ?:

Monte-Carlo (?? jets)

1e essai: 2e essai: 3e essai:

Commentaires:

Commentaires sur la séance:

Après la séance, nous avons ramassé les feuilles, non nominatives, de TD; les cases étaient soigneusement remplies et avec des valeurs correctes; par contre les commentaires étaient peu nombreux, les plus pertinents portant sur l'épaisseur des traits et sur la qualité "surprenante" (aux yeux des étudiants interrogés) de la méthode probabiliste.

Nous n'avons pas pu tester s'il y avait eu une meilleure compréhension de la théorie de l'intégrale que lorsque l'on se contente en cours des illustrations habituelles au tableau, par contre la manipulation de la machine par les étudiants eux-mêmes et non par l'enseignant augmentait leur confiance dans ce qu'il voyaient (pas de trucage possible ou d'exemples préparés pour que "ça marche").

DEVELOPPEMENTS LIMITES

Nous avons là aussi associé le point de vue **numérique et graphique** (utilisation de mon logiciel DEVLIM) et le point de vue **formel** en utilisant MU-MATH.

Utilisation de DEVLIM.

Il s'agit d'un programme en Turbo-Pascal qui utilise les mêmes fenêtres et les mêmes procédures d'entrée que INTEGRALE et qui est essentiellement un traceur de courbes permettant de tracer sur un même écran le graphe de plusieurs fonctions (typiquement: une fonction et ses développements limités à différents ordres au voisinage du point considéré). Des possibilités diverses, notamment de Zoom sont offertes (elles sont détaillées dans la feuille de TP ci-dessous), malheureusement l'absence de couleur gêne la distinction entre les différentes courbes (cette année la mise à notre disposition de machines couleurs permettra de corriger cette lacune). Les étudiants doivent calculer eux-mêmes les développements limités.

On met ainsi en évidence la propriété d'approximation de la fonction par ses développements limités. Mais les exemples comme le b) peuvent être trompeurs: on a l'impression que plus l'ordre est grand plus l'intervalle où l'approximation est bonne est grand. Il n'en est rien et on met aussi en évidence le **caractère local du développement limité.** Dans l'exemple f), on a l'impression que le développement limité à l'ordre 2 est "moins bon" que celui à l'ordre 1.

Voici la feuille de TP fournie aux étudiants et les recopies de quelques écrans obtenus (les chiffres entourés indiquent l'ordre des développements limités):

Séance sur machine: Développements limités
Utilisation du logiciel "DEVLIM"

1)Insérer la disquette "DEVLIM-INTEGRALE" dans le lecteur A et allumer l'appareil.

2)Après l'apparition du **A\>** du MS-DOS taper "**DEVLIM**" puis ↵. Le logiciel démarre.

3)<u>Introduction de la fonction:</u> grace à l'analyseur syntaxique de Modulog, on l'introduit comme une chaîne de caractères dans la fenêtre principale, la variable doit **impérativement** s'appeler **t**. Les chaînes correctes sont les expressions formées avec les opérations +, -, *, /, ^ (le signe ^ désignant l'exponentiation), les fonctions:

ABS	valeur absolue
ACH	argument cosinus hyperbolique
ACOS	arc cosinus
ASH	argument sinus hyperbolique
ASIN	arc sinus
ATAN	arc tangente
ATH	argument tangente hyperbolique
CH	cosinus hyperbolique
COS	cosinus
EXP	exponentielle
FRAC	partie fractionnaire
INT	partie entière
LN	logarithme népérien
SH	sinus hyperbolique
SIN	sinus
SQR	carré
SQRT	racine carrée
TG	tangente
TH	tangente hyperbolique

La fonction s'affiche en haut de l'écran.

4)<u>Cadrage:</u> le programme demande dans la fenêtre au milieu à droite de l'écran, les **valeurs du centre**, de la **hauteur** et de la **largeur de la fenêtre**, il s'agit du x_0 où l'on prend le développement limité, la demi-hauteur ht (-ht < y < ht) et la demi-largeur l ($x_0-l < x < x_0+l$). Il faut les entrer sous forme de **nombre décimal** (1.414 et non $\sqrt{2}$, même +2 au lieu de 2 est refusé). Valider par ↵. Les valeurs choisies s'affichent en haut à droite.

Remarque: a)le programme fait automatiquement une translation pour que $f(x_0)=0$,

b)les "polynômes" peuvent être rentrés sous la forme t^i, il est inutile d'appliquer le schéma de Hörner,

c)on peut utiliser indifféremment minuscules ou majuscules mais la variable t est impérative.

5)Zoom: les rapports horizontal (rl) et vertical (rv) sont les coefficients de multiplication de la fenêtre, exemple

avant:	larg: 1	rl: 2	**après:**	larg: 2
	haut: 1	rv: 0.5		haut: 0.5

6)En cas d'erreur:

a)avant validation: corriger (avec ⟵).

b)après validation: on risque une erreur d'éxécution (Run-time Error) auquel cas on recommence en 2).

Exercices:

Calculer (sur une feuille de papier) **au moins trois termes non nuls des développements limités à l'origine des fonctions suivantes et représenter sur l'écran les graphes correspondants.** On pourra prendre les valeurs initiales de cadrage ht et *l* suggérées ci-dessous.

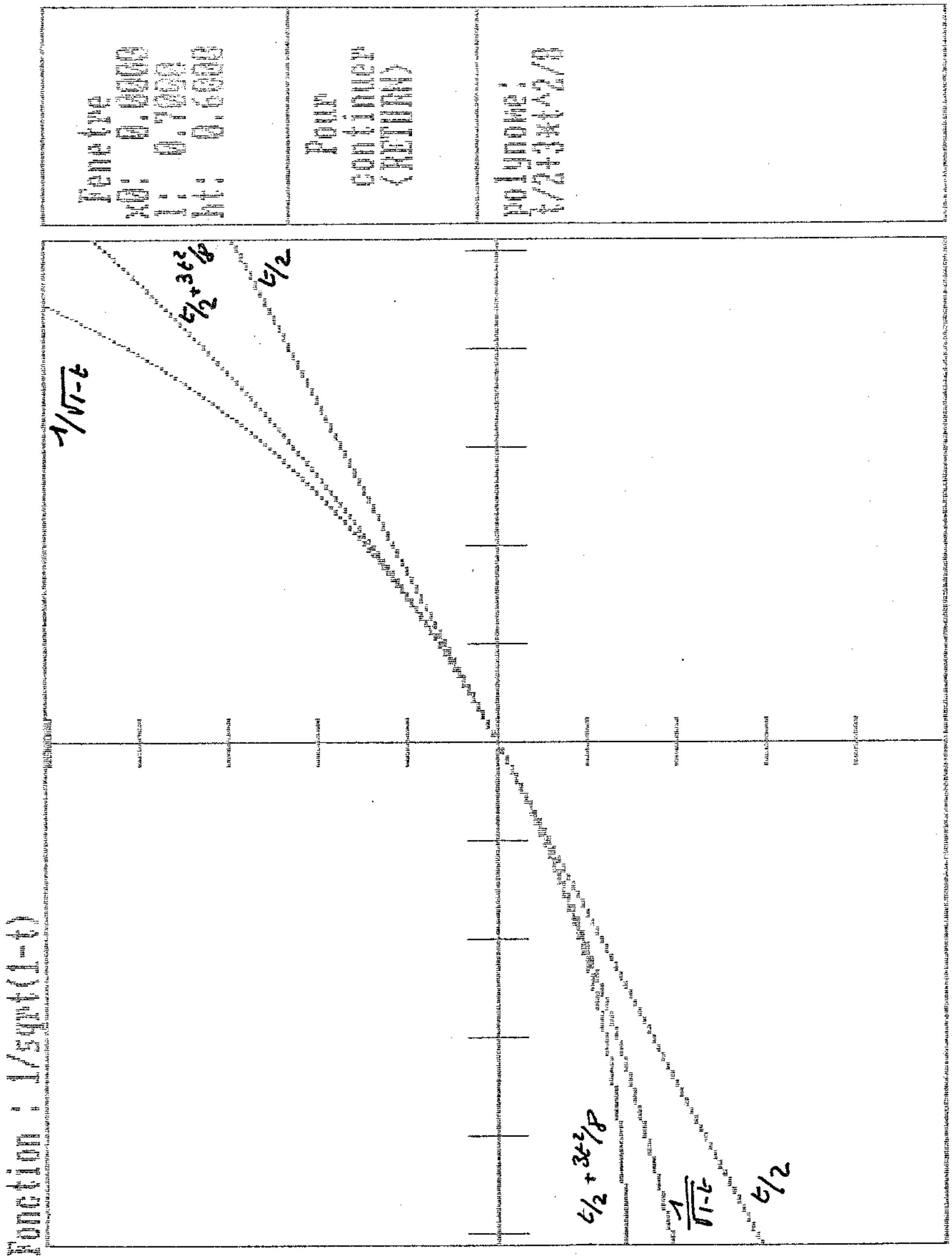
a)	tg(t) ou th(t)	ht = 2, *l* = 1.5
b)	sin(t^2)	ht = 2, *l* = 4
c)	sin(t^2)*exp(t/4)	ht = *l* = 2
d)	t*sin(t*(t+1))	ht = *l* = 4
e)	1 / sqrt(1-t)	ht = 1, *l* = 0.9
f)	ln(1+t)*cos(4*t)	ht = 1.5, *l* = 0.9

Dans le dernier cas, utiliser plusieurs fois le Zoom pour voir de plus près ce qui se passe et expliquer le paradoxe apparent que l'on constate sur l'écran.

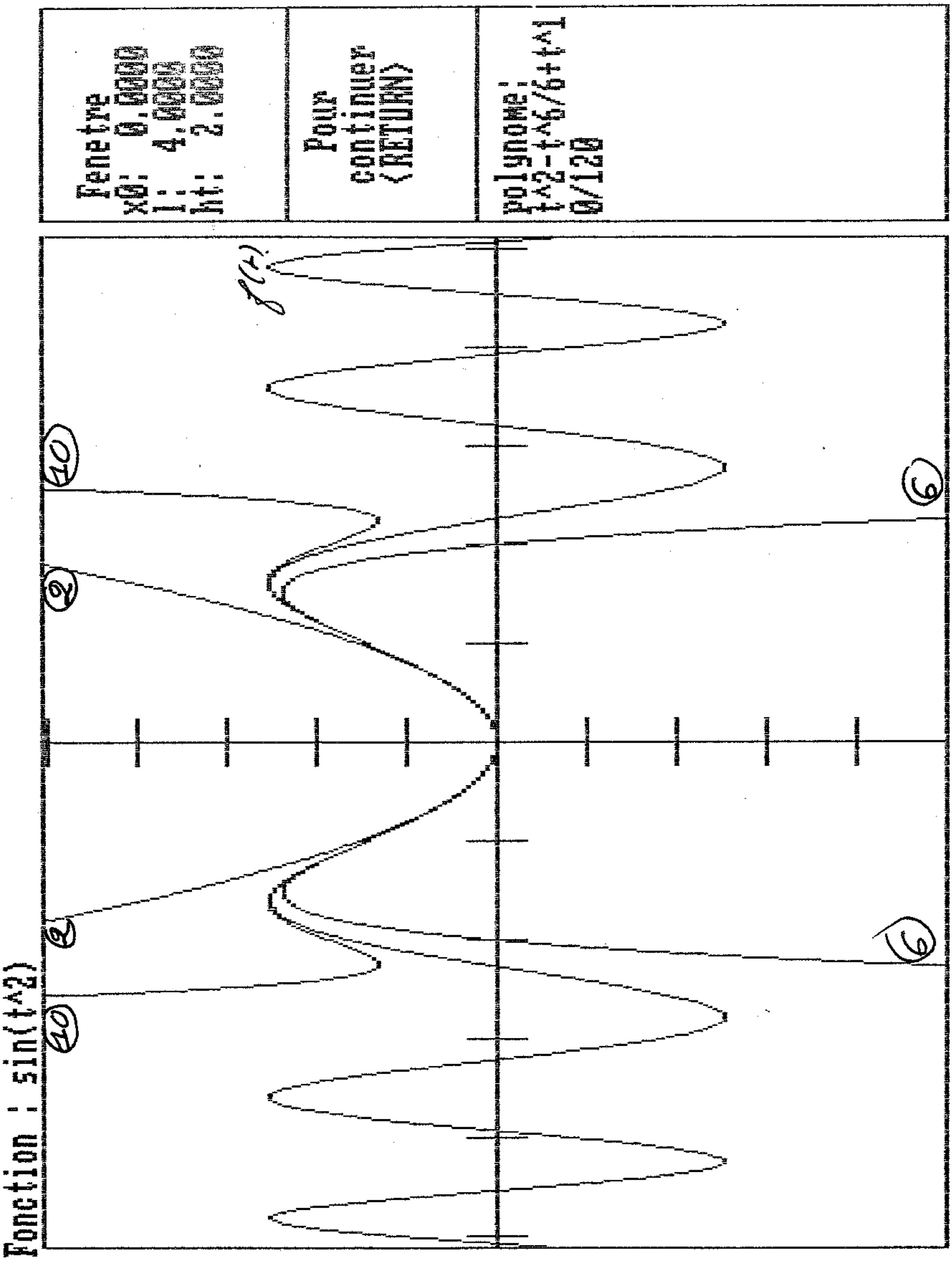
```
Fonction : 1/sqrt(1-t)

  MENU

     0  Fin
     1  Changer de fonction principale
     2  Changer la fenetre
     3  Changer de fonction et de fenetre
     4  Changer de polynome
     5  Changer de polynome (nouvel ecran)
     6  Zoom
```

```
 Fenetre
x0:  0.0000
l:  0.7000
ht:  0.6000
```

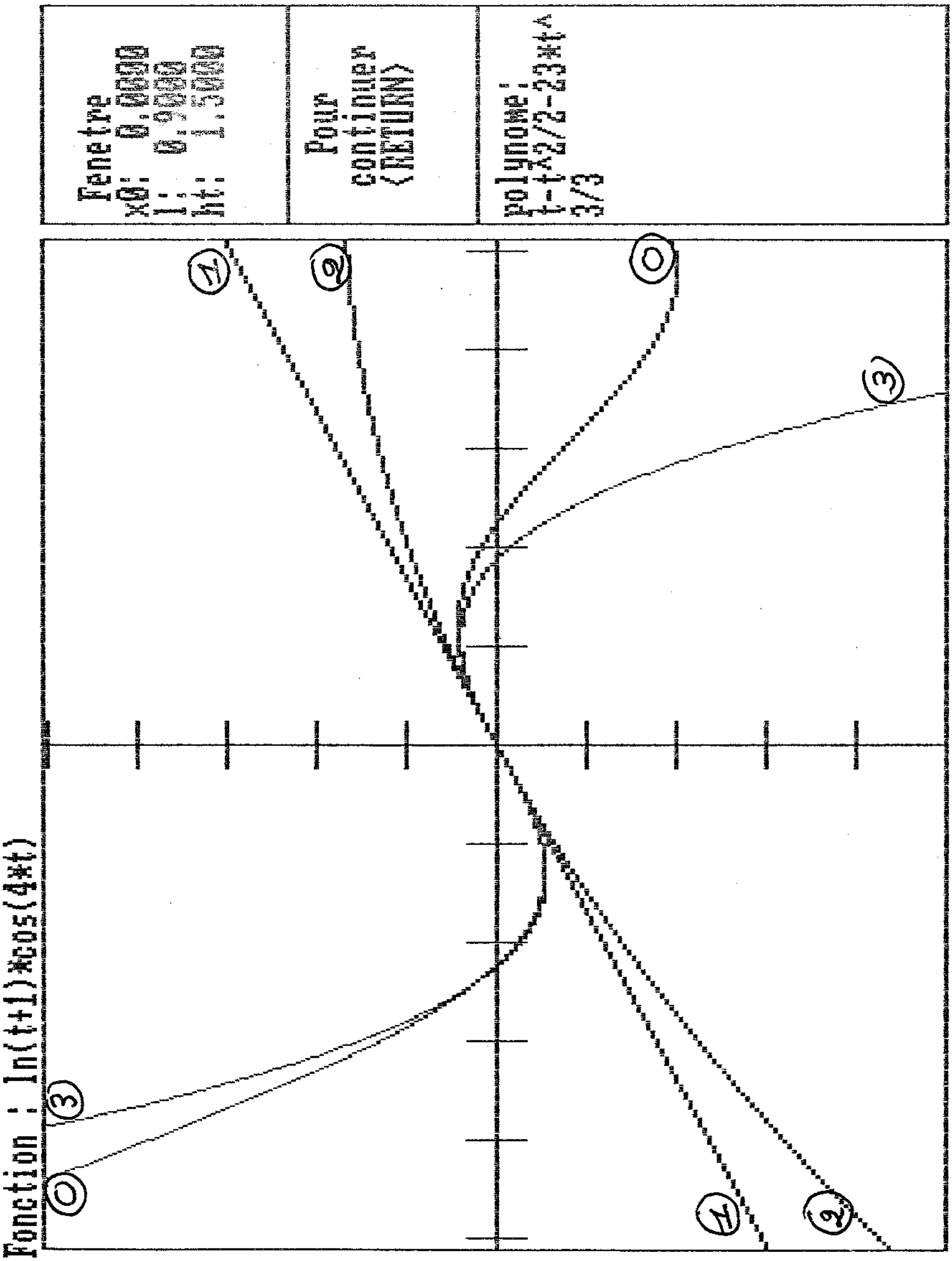
Exemple e)

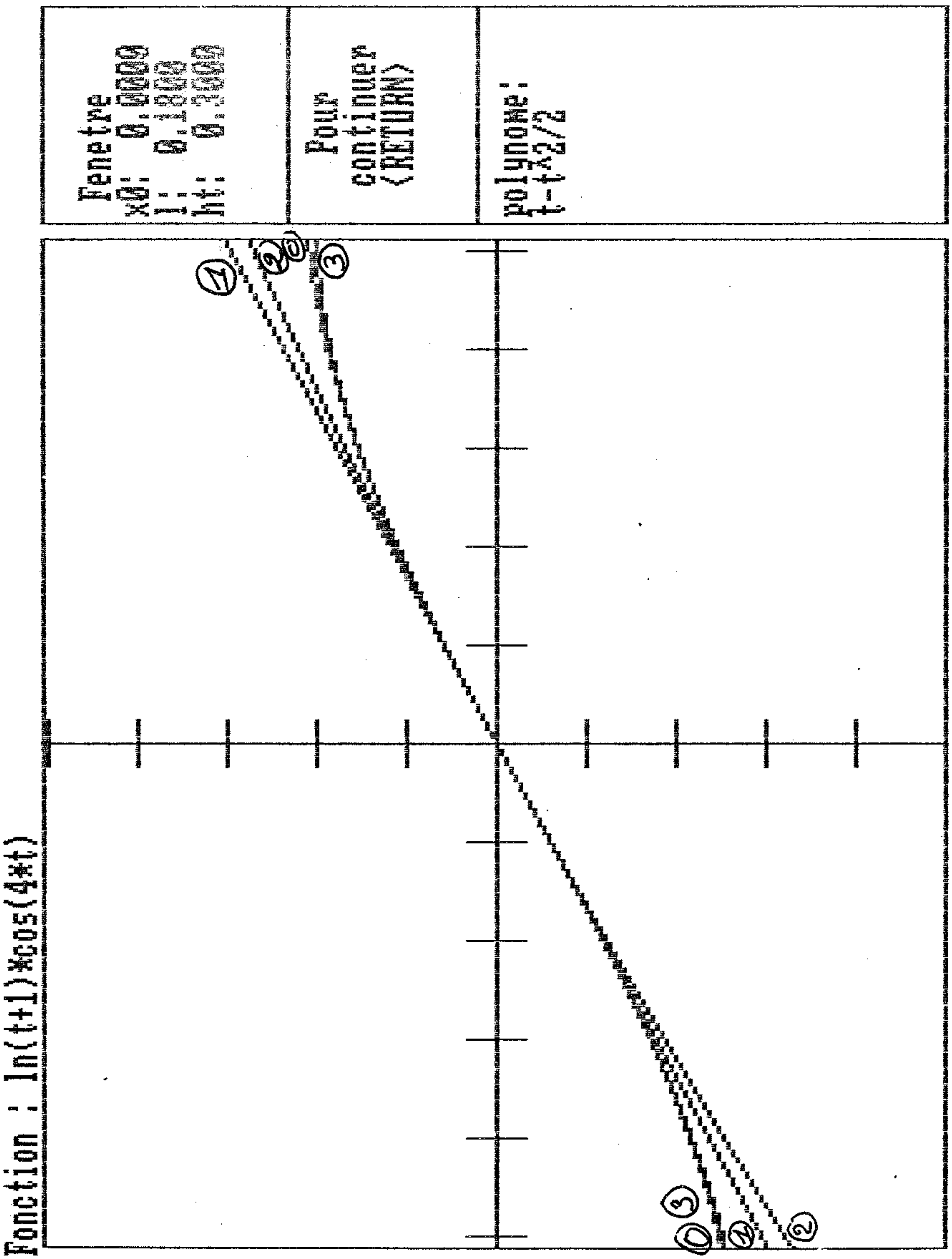

Fonction : 1/sqrt(1-t)
Pour continuer (RETURN)
Polynome : t/2+3*t^2/8
1/√(1-t)
t/2 + 3t²/8
t/2
t/2 + 3t²/8
1/√(1-t)
t/2

Exemple b)

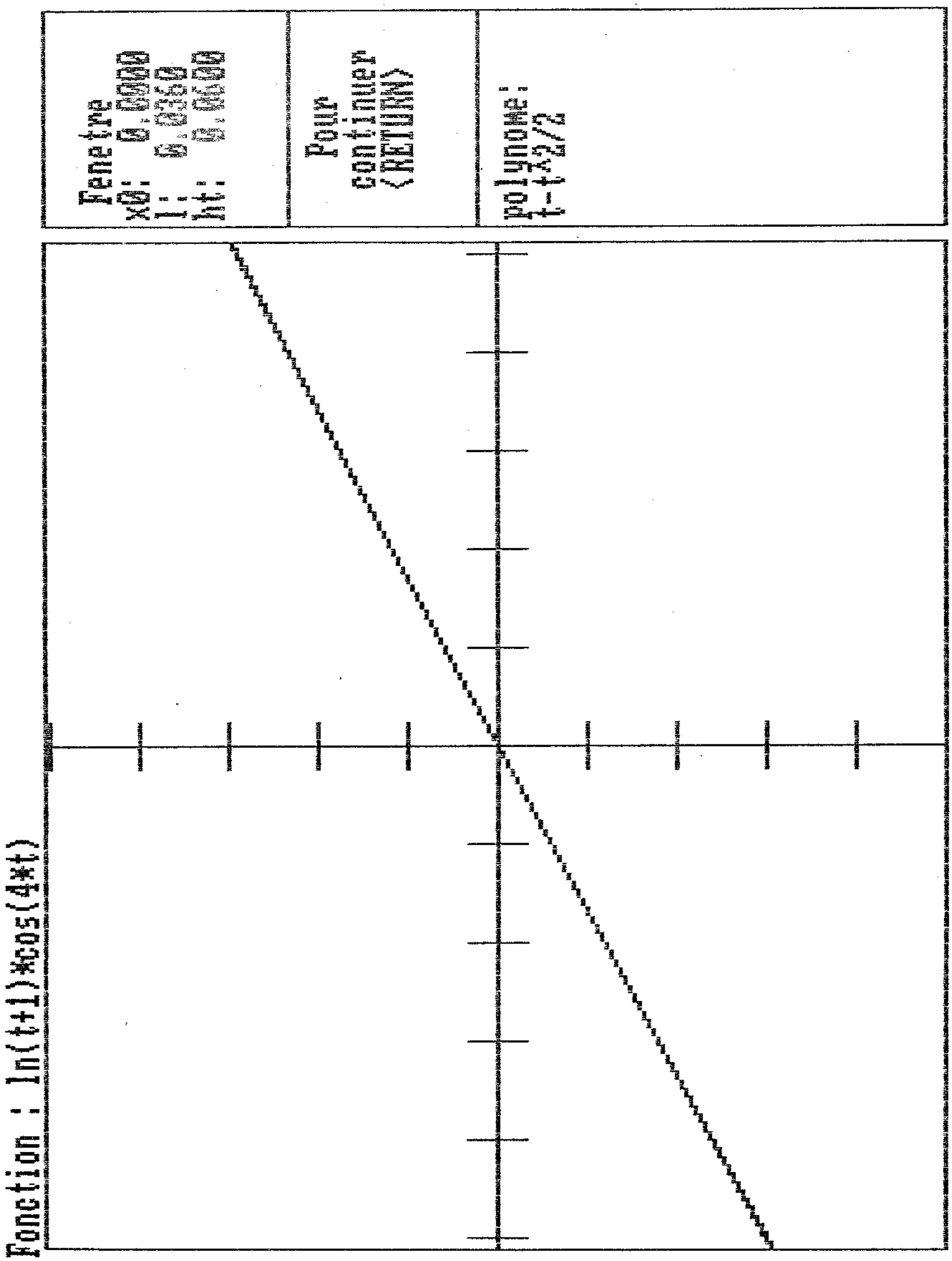


Fenetre
x0: 0.0000
l: 4.0000
ht: 2.0000
Pour
continuer
<RETURN>
polynome:
t^2-t^6/6+t^10/120
f(t)
10
2
6
Fonction : sin(t^2)


Fenetre
x0: 0.0000
l: 0.9000
ht: 1.5000
Pour
continuer
<RETURN>
polynome:
t-t^2/2-23*t^
3/3
Fonction : ln(t+1)*cos(4*t)
0
1
2
3


Fenetre
x0: 0.0000
l: 0.1800
ht: 0.3600
Pour
continuer
<RETURN>
polynome:
t-t^2/2
Fonction : ln(t+1)*cos(4*t)
1
2
0
3
3
0
1
2

après un second agrandissement



Les étudiants ont eu un peu de mal à expliquer le "paradoxe apparent" de l'exemple f). Ils étaient déçus de l' "inefficacité du Zoom", il a fallu expliquer que cela montrait que l'approximation par le premier ordre était au voisinage du point déja très bonne et nous l'avons justifié par un petit calcul numérique sous Multiplan par exemple donnant les valeurs de la différence entre la fonction et son développement limité à l'ordre 1,2,3 successsivement. Sur le tableau numérique on voit que la hiérarchie des développements limités est respectée.

t=dl1	f(t)	f(t)-dl1	f(t)-dl2	f(t)-dl3
-0.1	-0,097043	0,0029565	0,0079565	0.0002899
-0,09	-0,088265	0,0017349	0,0057849	0,0001959
-0,08	-0,079149	0,0008512	0,0040512	0,0001259
-0,07	-0,069744	0,0002555	0,0027055	7,587E-05
-0,069	-0,06879	0,0002099	0,0025904	7,183E-05
-0,068	-0,067833	0,0001666	0,0024786	6,794E-05
-0,067	-0,066874	0,0001256	0,0023701	6,42E-05
-0,066	-0,065913	8,675E-05	0,0022648	6,062E-05
-0,065	-0,06495	5,014E-05	0,0021626	5,718E-05
-0,064	-0,063984	1,566E-05	0,0020637	5.389E-05
-0,063	-0,063017	-1,67E-05	0,0019678	5,073E-05
-0,062	-0,062047	-4,71E-05	0,0018749	4,771E-05
-0,061	-0,061075	-7,55E-05	0,001785	4,483E-05
-0,06	-0,060102	-0,000102	0,0016981	4,207E-05
-0,05	-0,050271	-0,000271	0,0009792	2,082E-05
-0,04	-0,040301	-0,000301	0,0004994	8,747E-06
-0,03	-0,03024	-0,00024	0,0002098	2,836E-06
-0,02	-0,020138	-0,000138	6,191E-05	5,735E-07
-0,01	-0,010042	-4,23E-05	7,703E-06	3,668E-08
0	0	0	0	0
0,01	0,0099424	-5,76E-05	-7,63E-06	3,831E-08

Le temps nous étant compté et le programme de première année ne le prévoyant pas nous n'avons pas abordé, bien qu'ils se prêtent très bien à des séances sur machine, **les problèmes d'approximation ou d'interpolation de fonctions.**

Utilisation de MU-MATH:

Mu-Math, logiciel anglo-saxon, comme les mathématiciens de son pays d'origine ne calcule pas de développements limités, mais calcule des développements de Taylor d'une fonction en un point à un ordre donné ce qui pour les fonctions généralement développables en série entière que l'on considère sur machine revient au même.. ou presque, car ce procédé est beaucoup moins efficace.

On peut alors envisager deux utilisations:

des calculs effectifs de développement de Taylor d'une fonction, à des ordres élevés que l'on ne calculerait pas à la main mais l'on risque de démotiver les étudiants: quel intérêt y a-t-il à faire à la main des calculs que la machine peut faire?

ou, en choisissant bien les exemples, montrer que la méthode des développements de Taylor (on calcule les dérivées jusqu'à l'ordre n pour avoir le développement à l'ordre n) est lourde (voire inapplicable si la fonction ne vérifie pas les bonnes hypothèses de dérivabilité) et que la méthode des développements limités est beaucoup plus efficace, en demandant à l'étudiant d'employer la seconde méthode à la main pendant que la machine utilise la première (utiliser de préférence un IBM-PC qu'un Goupil..).

Exemples:

$f(x) = x^5 / \cos(x)$ à l'ordre 7 en 0 et tous les exemples de ce type où la machine est battue sans discussion,

$f(x) = 1 / (1 + x^2)$ à l'ordre 8 en 0,

$f(x) = \mathrm{tg}\, x$ en 0: la division marche bien,

un autre exemple intéressant est (qu'on trouve dans tous les bons ouvrages) de la recherche d'un équivalent de sin tg x - tg sin x en 0, la machine fournit le résultat si on lui demande un développement à l'ordre 7 mais il faut plusieurs minutes et, si le drapeau TRGEXPD est à sa valeur par défaut 0, le résultat prend plus d'un écran (25x80 caractères): la machine ne fait pas les simplifications du type tg(0) = 0 et en fait on voit apparaître le développement limité à l'ordre 7 en un point Q (on voit donc le nombre de termes qu'il a fallu calculer). Il suffit de donner la valeur 2 à ce drapeau pour retrouver le résultat classique.

A titre d'exemple, sur un Goupil G4 inutilisé nous avons lancé le calcul du développement limité de

$$\sin \operatorname{tg} x - \operatorname{tg} \sin x \text{ en } 0 \text{ à l'ordre } 11$$

et nous avons lu sur l'écran (après plus de 20 mn et en ayant mis le drapeau TRGEXPD à 2):

$$-x^7/ 30 - 29\, x^9/ 756 - 1913\, x^{11}/ 75600 + x^{11}\varepsilon(x)$$

(résultat que l'on ne trouve pas dans tous les bons ouvrages).

Nous avons profité de cette séance pour faire quelques calculs sur l'algorithme de Sturm.

ALGORITHME DE STURM

Soit P un polynôme de degré m à coeeficients réels, on définit par récurrence des polynômes P_0, P_1,.. ,P_m par division euclidienne:

$$P_0 = P$$
$$P_1 = P'$$
$$P_0 = P_1\,Q_1 - P_2$$
$$P_1 = P_2\,Q_2 - P_3$$
....
$$P_{m-2} = P_{m-1}\,Q_{m-1} - P_m \qquad \text{avec} \qquad P_m = 0$$

et pour tout réel x, on appelle "**suite de Sturm de P en x**" la suite

$$(P_0(x),\ P_1(x),..\ ,P_m(x))$$

On appelle $\Delta(x)$ le nombre de changements de signes de la suite précédente (les zéros ne comptent pas). On a alors le

<u>Théorème de Sturm:</u>

Soit P un polynôme à coefficients réels, étant donnés deux réels a *et* b (a < b) *tels que* $P(a).P(b) \neq 0$, *le nombre de zéros distincts du polynôme P dans l'intervalle* [a b] *est* $\Delta(a) - \Delta(b)$.
(on ne compte pas les multiplicités).

Ce théorème a été démontré en cours, il est utilisé effectivement sur machine pour séparer les zéros d'un polynôme, son application à la main est envisageable mais, même en partant de coefficients très simples, se révèle rapidement laborieuse (la croissance des dénominateurs est grosso modo exponentielle). Il est donc très intéressant de pouvoir l'appliquer avec une machine qui calcule exactement sur les rationnels.

Nous avons donc demandé dans un premier tempsaux étudiants d'utiliser la machine sous Mu-Math comme une calculatrice polynômiale pour dérouler les calculs, puis ensuite nous leur avons fourni la fonction STURM qui donne directement la suite de Sturm, il ne reste plus qu'à compter les changements de signes.. et à observer la taille des dénominateurs.

Séance sur machine: Calcul Différentiel
Utilisation du logiciel "Mu-Math"

Insérer la disquette "DEUG-Mu-Math" dans le lecteur de gauche et allumer la machine. Lancer le programme en tapant "CALCULDIF" puis [↵]

En cas d'erreur [CONTROL|ARRET] (ou BREAK) arrête les calculs, répondre A (pour Abort).

Pour modifier une entrée sur une ligne non validée utiliser la touche de retour arrière [←], pour renouveler une entrée on peut utiliser les touches de fonction du DOS [F1] et [F3].

1)Polynômes (critère de Sturm):

a)(méthode décomposée)

```
rentrer        p0:x^3-3x+1;           [↵]
               p1:DIF(p0,x);          [↵]
               p2:-PREM(p1,p0,x);     [↵]
               p3:-PREM(p2,p1,x);     [↵]
               .....
```

s'arrêter quand pr = 0.

pour le nombre de zéros dans]a b], noter les résultats de

```
EVSUB(pi,x,a); [↵]      pour i = 0,..  ,r - 1
EVSUB(pi,x,b); [↵]      pour i = 0,..  ,r - 1
```

et appliquer le critére.

```
b)entrer   p:....;            [↵]
           q:sturm(p);        [↵]
           evsub(q,x,a);      [↵]
           evsub(q,x,b);      [↵]
```

et compter les changements de signes.

Polynômes suggérés:

a)x^3 - 3 x + 1 entre -2 et 2

b)4 x^5- 3 x^4 + 2 x^3 - 2 x^2 - 5 x - 3 entre 0 et 2

c)90 x^5 - 60 x^4 - 161 x^3 + 204 x^2 -79 x + 10 entre 0 et 2

d) $x^7 - 11x^5 + 12x^4 - 25x^3 + 24x^2 - 13x + 12$ entre -2 et 4

e) $x^7 - 3x^5 + x^4 + 2x^3 - 6x + 1$ entre -2 et 2

2) Développements de Taylor:

La syntaxe est

```
TAYLOR (f,     x,   a,     n );      [↵]
        |      |    |      |
        └───── la fonction de la variable x
               |    |      |
               └─── la variable: x
                    |      |
                    └───── le point où l'on prend le développement
                           └──── l'ordre du développement
```

Calculer, à l'aide de la machine et en parallèle sur une feuille de papier les **développements limités** de

$f(x) = x^5 / \cos(x)$ à l'ordre 7 en 0

$f(x) = 1 / (1 + x^2)$ à l'ordre 8 en 0

$f(x) = \text{tg } x$ en 0 à l'ordre 7

$f(x) = \sin \text{tg } x - \text{tg} \sin x$ en 0 à l'ordre 7

la ou les fonctions de votre choix.

Utilisation de Mu-Math:

Entrées au clavier

Toute entrée au clavier doit se terminer par la séquence ; <return>

exemple :
```
sin x + cos x ; <return>
--> sin x + cos x
```

Le <return> seul fait simplement passer à la ligne suivante . Il est alors possible de poursuivre l'introduction de l'expression

exemple:
```
sin x       <return>
+ cos x ;   <return>
--> sin x + cos x
```

Tant que le ; n'a pas été frappé, le système est en attente de la suite de l'expression. Si vous l'oubliez avant de taper sur la touche <return>, le curseur se positionne sur le début de la ligne suivante. Il vous est alors possible de conclure l'introduction par l'entrée de la séquence ; <return>

exemple :
```
sin x + cos x <return>
;             <return>
--> sin x + cos x
```

Règles de syntaxe:

PRODUITS: A x B se note A * B. ex : x*cos x +(x+1)*(x+2);
si A est un nombre, le * est inutile. ex : 5x + 4b;

QUOTIENTS: $\dfrac{A}{B}$ s'écrit A/B. ex : $\dfrac{4}{x}$ --> 4/x;

attention au parenthèsage : $\dfrac{x+2a\ b}{x-3y\ z}$ --> (x+2a b)/(x-3y z);

PUISSANCES: a^{b} s'écrit a^b; RACINES CARREES: $\sqrt{x}$: rac(x) ou x^(1/2)

Syntaxe des autres fonctions :

	sinus	sin x , sin (x) , sin (2x) , sin (x+3)
	cosinus	cos x , cos (x) , cos (x^2), (cos x)^2 ,
	tangente	tan x
attention	: cos 2x -->	(cos 2) * x !!!
	log népérien	ln x , ln (x y) , ln (x+2y),
	log decimal	log x ,
	log base a	log (x,a) , log (2x+3,5) ,
	valeur absolue	abs x , abs(2x+1) ,
	fonctions hyperboliques	sinh x, cosh x, tanh x
Constantes disponibles		#pi et #e

Dérivée de la fonction f par rapport à x: DIF(f,x); ↵

Donner à a la valeur 35/4: a:35/4; ↵

Définir la fonction f par f : $2x^3+1$ f:2 x^3+1; ↵

Division euclidienne du polynôme P par le polynôme Q (variable: X):

Quotient: PQUOT(P,Q,X); ↵

Reste: PREM(P,Q,X); ↵

ALGEBRE LINEAIRE

Ici aussi nous retrouvons la double approche: numérique et formelle.

Approche "numérique":

Le point d'appui de nos travaux pratiques a été la **méthode du pivot** implantée en Turbo-Pascal dans deux programmes de R.Pallu de la Barrière (qu'il soit remercié ici): DETER1 et JORDANØ.

Le premier programme calcule le déterminant d'une matrice numérique par pivotement.

Le second programme réalise une procédure de Jordan sur une matrice (nxm), l'utilisateur choisit librement à chaque itération le pivot en indiquant la ligne et la colonne. On peut l'utiliser

pour chercher le rang d'un système de vecteurs, d'une matrice (quand on ne peut plus pivoter le rang est évident),

chercher l'inverse d'une matrice (le programme ne permutant pas les lignes et les colonnes si on ne prend pas les pivots sur la diagonale, on peut avoir à multiplier par une matrice de permutation),

calculer un déterminant (en faisant le produit des pivots, attention au signe),

résoudre des systèmes linéaires (avec plusieurs seconds membres éventuellement).

Le Turbo-Pascal affiche tous les chiffres qu'il calcule, aussi voit-on très bien les problèmes d'erreurs d'arrondi (qui sont partiellement masqués sur les calculateurs de poche qui calculent avec 2 chiffres de plus que l'affichage).

Nous avons rencontré un problème sérieux: un des exercices de l'examen de fin d'année était la résolution et la discussion d'un système linéaire 3x3 dépendant d'un paramètre, exercice pour lequel la méthode de Cramer est nettement préférable à celle du pivot ce qui nous a conduits à modérer nos efforts sur la méthode du pivot.

Voici, réalisée avec l'aide de F.Blanchard, la feuille remise aux étudiants.

Séance sur machine: Algèbre Linéaire

Insérer la disquette "DEVLIM-INTEGRAL-DETER1-JORDANØ" dans la machine et mettre en marche.

DETER1:

On lance ce programme en tapant: DETER1 ↵

Le programme calcule le déterminant d'une matrice M, il demande:

n = dimension de la matrice (un entier bien sûr),

les coefficients (réels) que l'on doit indiquer colonne par colonne,

exemple: pour $\begin{pmatrix} 3 & 1 \\ 0 & 1 \end{pmatrix}$ on entrera la séquence:

2 ↵
3 ↵
0 ↵
1 ↵
1 ↵

et le programme affiche la valeur du déterminant (ici 3 comme on s'y attend).

Exercices: 1) Calculer le déterminant de quelques matrices de votre choix.

2) Calculer le déterminant de la matrice:

$$\mathbf{M} = \begin{bmatrix} 1.23 & 1.02 & 3.27 \\ 1.22 & 1.01 & 3.24 \\ 1.21 & 1.03 & 3.27 \end{bmatrix}$$

a)sur la machine b)à la main

Comment expliquez-vous la différence?

JORDANØ:

On lance en entrant JORDANØ ↵

Le programme fait la méthode Gaus-Jordan comme en cours, il demande d'entrer:

n = nombre de lignes,

m = nombre de colonnes (on n'a pas nécessairement n = m),

les coefficients colonne par colonne (en validant par ↵ à chaque fois)

Le programme affiche ce qu'on vient d'entrer sous la forme:

$$\begin{matrix} A & B \\ I & \end{matrix}$$

où A et B sont des matrices et I la matrice identité ayant la même largeur que A

La machine demande alors un **pivot** (indiqué par ses indices de ligne et de colonne), affiche:

$$\begin{matrix} K_1A & K_1B \\ K_1I & \end{matrix}$$

où K_1 est la matrice de pivotement et demande un nouveau pivot.

Après n pivotements on a:

$$\begin{matrix} K_nA & K_nB \\ K_nI & \end{matrix}$$

On s'arrête quand la matrice K_nA est une matrice de permutation. Si $K_nA = I$ alors $K_n = A^{-1}$.

Exercices: 1)Trouver le rang de la matrice $A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 3 \end{pmatrix}$ et, si elle est de rang 3, calculer son inverse.

2)Trouver le rang de la matrice $B = \begin{bmatrix} 4 & 2 & 3 & 4 & 1 \\ 0 & 6 & -1 & 1 & 5 \\ 5 & -3 & 4 & 3 & -4 \\ 8 & -2 & 7 & 7 & -3 \end{bmatrix}$

3)Soit $C = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$. Résoudre C X = V et C X = W où

$$V = \begin{bmatrix} 2 \\ 4 \\ 1 \\ 1 \end{bmatrix} \text{ et } W = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

4)Comment faut-il choisir les pivots pour obtenir les "meilleurs" résultats?

La méthode formelle:

Nous avons utilisé les possibilités de calcul formel de Mu-Math pour calculer exactement sur des matrices ou des systèmes linéaires à cefficients entiers ou même dépendant d'un paramètre. Les possibilités sont grandes: opérations sur les matrices (somme, produit, puissance, **inverse**), résolution de systèmes linéaires. Les choses sont à la fois plus simples et plus compliquées: les calculs exacts sont bien agréables, on peut manipuler des nombres complexes ou des nombres algébriques comme $\sqrt{2}$, par contre il y a plusieurs écueils:

d'abord l'interface utilisateur n'est pas très agréable, on doit travailler sous l'éditeur ligne du DOS (on ne peut rappeler une commande déja validée, et avant validation on ne peut circuler comme on veut sur la ligne courante),

la syntaxe d'entrée des matrices est lourde (en outre il faut avoir recours aux { et } qui ne sont pas sur le clavier français),

dans le cas de systèmes dépendant polynômialement d'un paramètre formel m, le résultat qui est une fraction rationnelle en m n'est pas toujours simplifié comme il faut (et est donc difficile à reconnaître) et le calcul peut prendre beaucoup de temps.

Ces derniers défauts peuvent être surmontés en utilisant Reduce qui, sur ce point, est beaucoup plus performant. Par exemple, l'examen de septembre comportait la question suivante (notée sur 4 points donc correspondant à un temps approximatif de 50 minutes):

Résoudre et discuter, suivant les valeurs du paramètre réel m, *le système suivant:*

$$\begin{cases} mx - 2y + z = 1 \\ x - 2my + z = -2 \\ x - 2y + mz = 1 \end{cases}$$

or si, sous Reduce, l'on entre les commandes:

```
solve(lst(m*x-2*y+z-1,x-2*m*y+z+2,x-2*y+m*z-1),x,y,z);
```

on obtient une minute plus tard:

```
SOLN(1,1) := 1/(M-1)
SOLN(1,2) := 1/(M-1)
SOLN(1,3) := 1/(M-1)
```

et après:

det mat ((m,-2,1),(1,-2*m,1),(1,-2,m));

on obtient:

$$2*(M+2)*(M-1)^2$$

et il n'est pas difficile d'en déduire la réponse à l'exercice.

Il faut donc faire très attention d'une part à ne pas démobiliser les étudiants d'autre part les applications linéaires sont occultées au profit des matrices ce qui est dommage, il faudrait des logiciels spécifiques[7].

Nous avons aussi montré aux étudiants l'exemple classique (permettant de mettre en valeur l'universalité de l'algèbre linéaire et son intérêt en programmation) de la suite de Fibonacci:

si on traduit directement la définition on obtient (en LOGO par exemple)

```
Pour fib :x
si :x = 0 [retourne 1]
si :x = 1 [retourne 1]
retourne fib :x - 1 fib :x - 2
fin
```

procédure très inefficace car pour obtenir fib(n) il faut environ 2^n appels de fib et la machine sature rapidement. (il est long d'obtenir fib(n) pour n supérieur à 20).

si on utilise un vecteur (ou une liste) à deux composantes, on a la procédure:

```
Pour fib :x
si :x = 0 [retourne list (1 1)]
si :x = 1 [retourne list (2 1)]
donne "v fib :x - 1
retourne list (dernier :v somme dernier :v premier :v)
fin
```

et pour obtenir le n^e terme le nombre d'appels n'est plus que linéaire en n, le gain obtenu est donc énorme.

[7] Un logiciel de l'Atelier Logiciel de l'Ecole Centrale devrait être disponible l'année prochaine.

EQUATIONS DIFFERENTIELLES

Ce chapître laisse beaucoup à désirer et pour une bonne référence nous renvoyons aux réalisations de M. Artigue et V. Gautheron à Paris 7 et de M. Rogalski et C. Sacré à Lille[8]. Notre volonté de rénovation de l'enseignement des équations différentielles s'est heurtée à trois obstacles:

le programme de la première année qui prévoit: "équations différentielles linéaires du premier ordre et équations linéaires à coeeficients constants du second ordre", les sytèmes étant renvoyés à la seconde année,

le sujet (commun aux quatre sections de DEUG SSM-MP) qui comme tous les ans comportait un exercice "à la Dixmier" sur les équations différentielles: on prend une équation linéaire du premier ordre, on cherche (exercice de révision sur le calcul des primitives) les solutions de l'équation homogène puis de l'équation complète sur les intervalles ad hoc de $\mathbb{R}$, (exercice de révision sur les développements limités) on cherche quelles sont les solutions qui se prolongent à des intervalles plus grands, (exercice sur les études de fonctions), on étudie les courbes solutions.

le manque de temps à la fin de l'année n'incitait pas à faire du supplément par rapport au programme.

Dans ces conditions (et dans certains groupes de Travaux dirigés seulement) nous nous sommes limités (outre des séances de préparation au type d'exercice ci-dessus) à deux séances:

une "normale" où nous avons étudié "à la main" et en détail l'allure des champs de vecteurs, des courbes intégrales des équations $y' = x^m$ pour différentes valeurs de m (on y rencontre les principaux phénomènes).

[8] voir par exemple le preprint pour l'ECM de Rome Juin 87:
M. ARTIGUE et V. GAUTHERON:
L'outil informatique et les équations différentielles

une séance sur machine où nous avons utilisé le logiciel MERLIN[9] qui présente l'avantage d'être très simple à utiliser (menus faciles d'emplois, valeurs initiales de cadrage et de paramètres proposées) pour étudier quelques équations différentielles en en traçant les champs de vecteurs, les courbes intégrales par différentes méthodes (Euler, Runge-Kutta).

Il est à remarquer que dans certains cas, on a l'équation des solutions mais que le tracé des courbes intégrales est plus facile à partir de l'équation différentielle par une méthode type Euler ou Runge-Kutta que par tracé direct.

Le logiciel proposait entre autres les équations différentielles:

$y' = x - x^2$

$y' = -y - x^2$

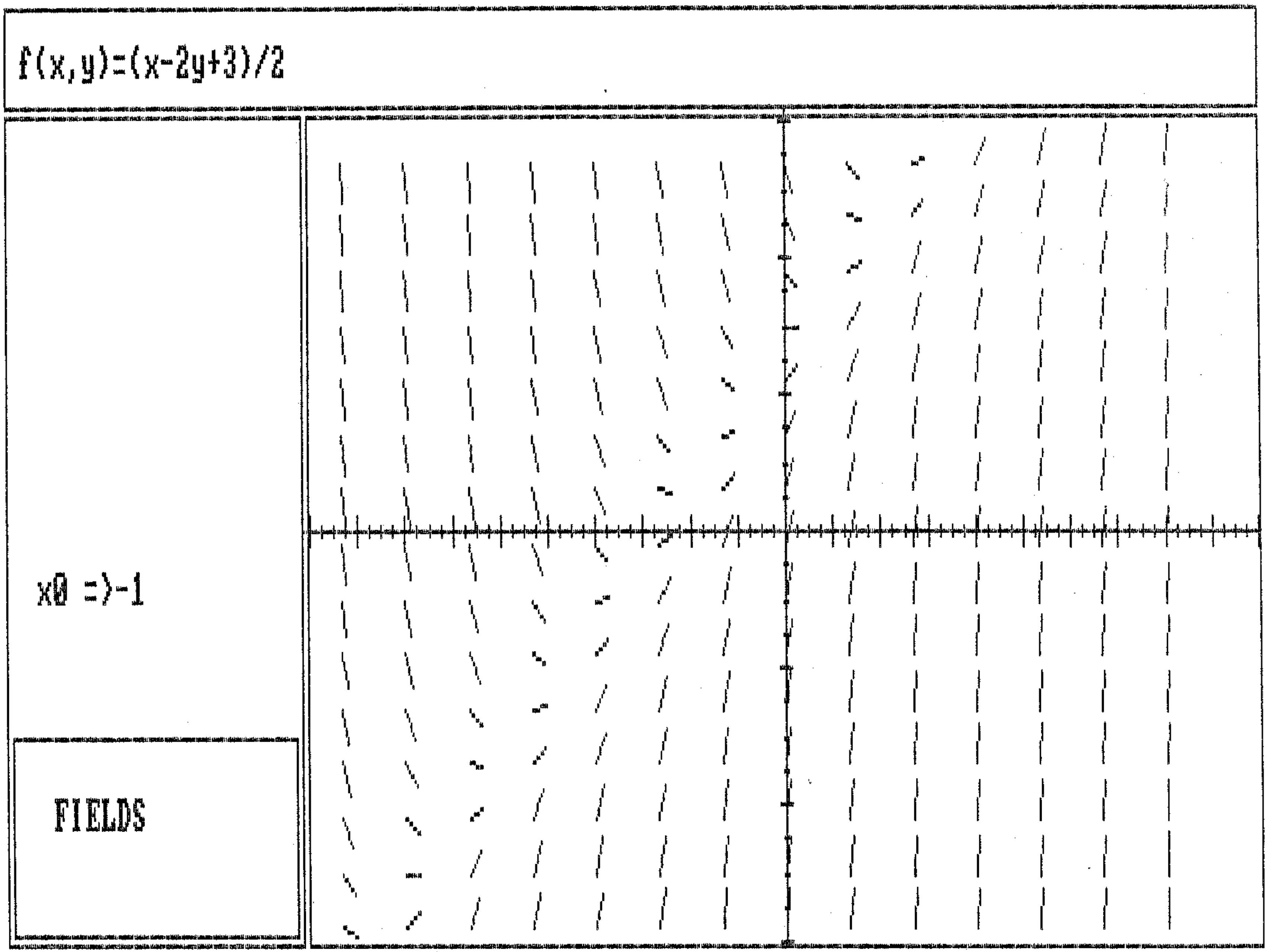
$y' = 3y/x$

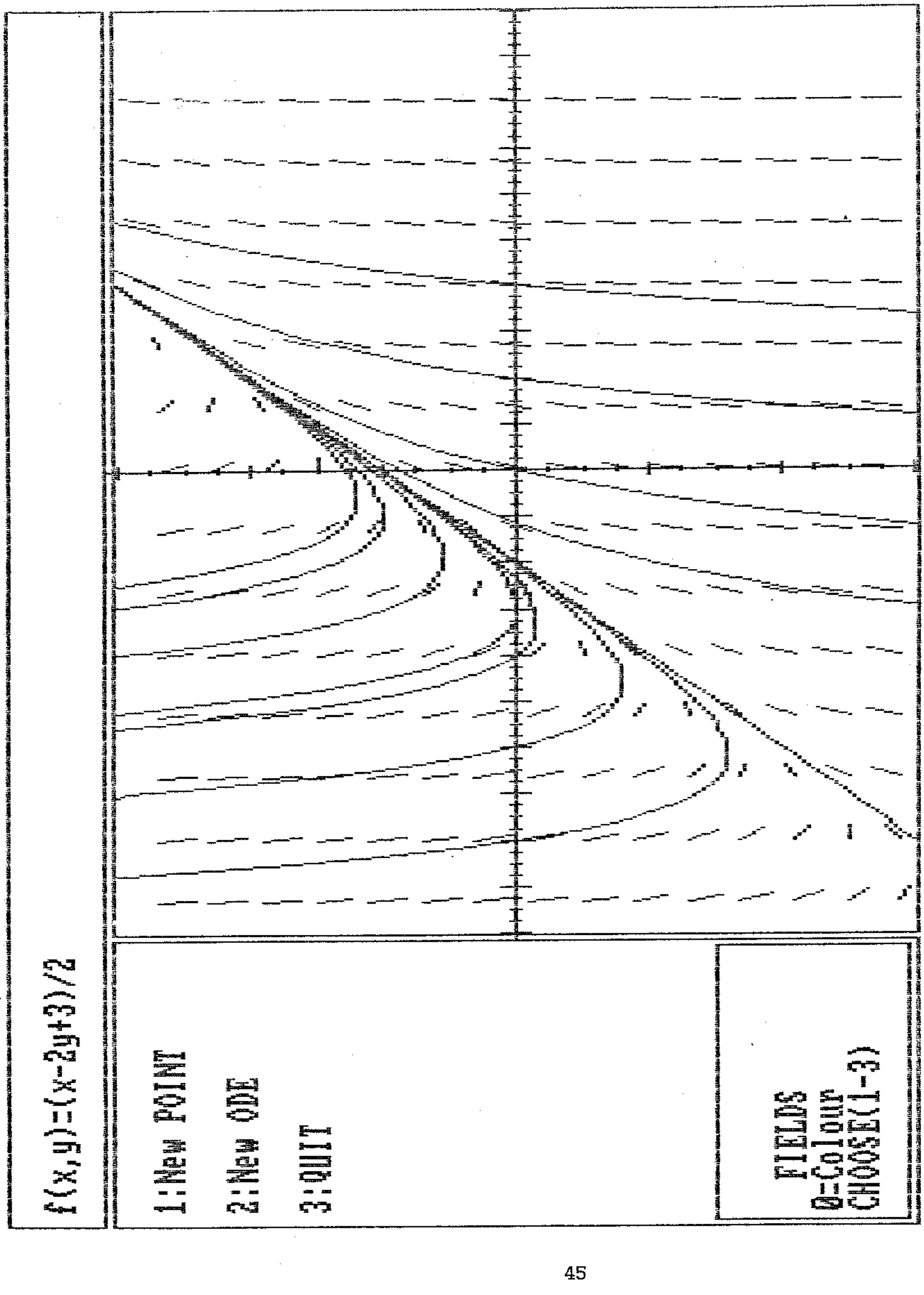
$y' = (x-2y+3)/2$

$y' = x+y/x$

Et le type d'écran obtenu était le suivant:

9 de P.J.PONZO Université de Waterloo Ontario (Canada)


f(x,y)=(x-2y+3)/2
x0 =)-1
FIELDS


f(x,y)=(x-2y+3)/2
1:New POINT
2:New ODE
3:QUIT
FIELDS
Ø=Colour
CHOOSE(1-3)

ET AUSSI...

Les machines ont aussi une influence sur les sujets de devoirs et d'examens et réciproquement. Nous en avons parlé dans les chapîtres consacrés aux équations différentielles et à l'algèbre linéaire. Mais il y a d'autres influences: les calculatrices graphiques de poche (comme la CASIO 7000) qu'ont certains étudiants ont conduit à écarter des sujets d'examens les tracés classiques de courbes (le fait d'obtenir quasi-instantanément l'allure générale du graphe sur un écran étant un avantage trop important pour les étudiants bien "équipés"), ne voulant pas adopter l'attitude passéiste d'interdiction des machines à l'examen, il a fallu restreindre l'étude des courbes à l'étude de points singuliers ou d'asymptotes, domaine où la théorie reste nécessaire.

De même il y a eu une retombée des machines sur des sujets de devoir.

-dans deux sections parallèles a été donné le "même" sujet dû à F.Digne:

1e version:

I. On pose $F(x) = \int_x^{3x} \frac{\cos t}{t}\, dt$ pour $x > 0$.

1. Montrer que F est dérivable.
2. Montrer que $|F(x)| \leq \ln 3$ pour tout $x > 0$.
3. Montrer que $\int_x^{3x} \sin t \ln t dt$ tend vers 0 si $x \to 0_+$. En déduire que $F(x)$ a une limite si $x \to 0_+$.
4. On note $G(x)$ la fonction définie sur $[0, \infty[$ par

$$G(x) = \begin{cases} F(x), & \text{si } x > 0; \\ \ln 3, & \text{sinon.} \end{cases}$$

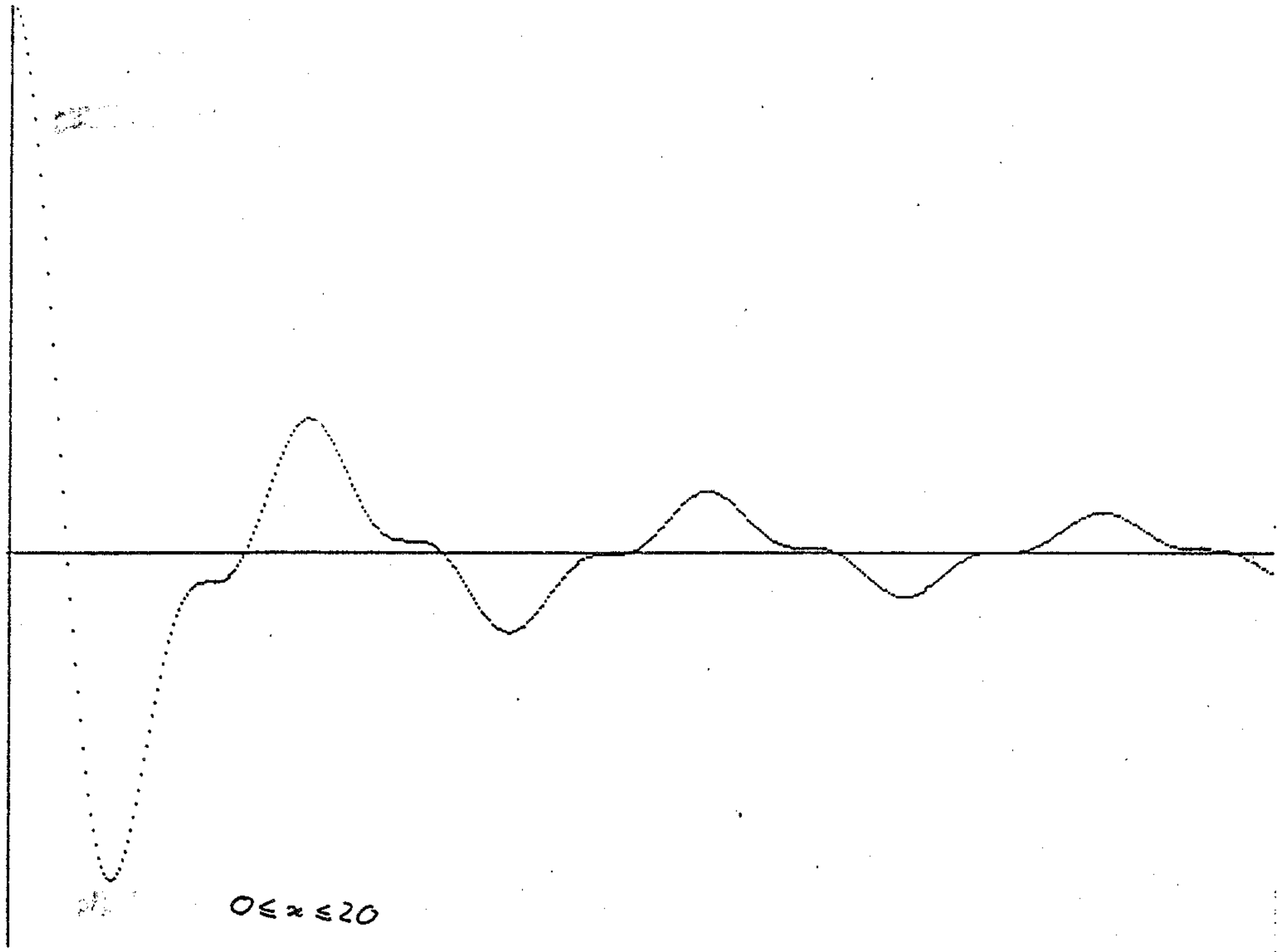
a) Montrer que G est dérivable à droite en 0, et calculer sa dérivée.

b) Majorer à l'aide d'une intégration par parties la valeur absolue

$$|G(x) - \frac{\sin(3x) - 3\sin(x)}{3x}|$$

pour $x > 0$. En déduire la limite de $G(x)$ quand $x \to \infty$.

c) Le graphe ci-dessous est celui de la fonction G. Préciser les abcisses des maxima et des minima de G. *Facultatif* Montrer que les ordonnées des maxima et minima sont bien placées comme l'indique la figure.



<u>2e version:</u> l'énoncé était sensiblement le même sauf la question 4c) où l'on demandait: "tracer la courbe", la courbe (tracée sur machine par C. Laurent en évaluant numériquement l'intégrale par la méthode des trapèzes à pas variable) n'étant donnée que dans le corrigé.

La principale différence à la correction était:

dans le deuxième cas de nombreux étudiants ont commis l'erreur suivante: la dérivée $G'(x) = -\dfrac{4 \sin^2 x \cos x}{x}$ s'annule pour $x = k\pi$

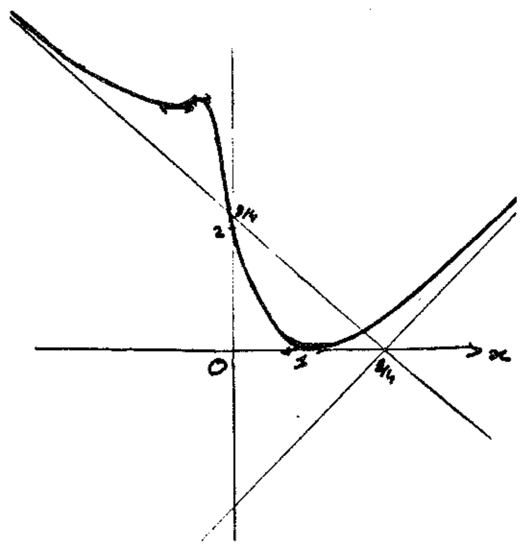
(k entier positif) donc ces points correspondent à des extrema.

dans le premier cas il y eu la même tentation d'erreur (des étudiants nous ont fait part de leur perplexité pendant des travaux dirigés avant la remise du devoir) mais dans les copies on trouvait moins cette erreur, donc la présence de la courbe dans l'énoncé avait eu une influence sur le raisonnement; nous n'avons pas fait d'étude pour savoir quel groupe d'étudiants aurait le meilleur résultat ensuite face à ce type de problèmes.

Un autre exemple est la courbe d'équation:

$$y = \frac{2\ (x-1)^2}{\sqrt{4x^2 + 2x + 1}}$$

Le corrigé donnait la courbe suivante, "tracée à la main" mettant en évidence les extrema en $x = -1$, $y = 8/\sqrt{3} = 4.618..$ et $x = -3/4$, $y = 7\sqrt{7}/4 = 4.63...$



alors que la présence de ces extrema est difficile à distinguer de l'existence d'un point d'inflexion à tangente horizontale sur l'image fournie par un traceur de courbe.

EN GUISE DE BILAN

Tout d'abord, regrettons qu'il ne soit pas prévu de "projets" pour les étudiants de premier cycle de notre université. Pour ceux qui choisiraient un projet d'application de l'informatique aux mathématiques on pourrait avoir là l'occasion de les faire travailler sur des thèmes où la nécessité de la programmation motiverait les étudiants pour des études théoriques (les problèmes de codage par exemple peuvent être une motivation pour l'étude des congruences ou des corps finis, voir aussi mon papier sur la génération tangentielle des courbes et l'enseignement de la géométrie[10]) qui a priori les rebutent.

De même l'option, dont nous ne sommes que partiellement responsables, de ne pas faire programmer les étudiants et de leur fournir des programmes tout faits, exclut un certain nombre d'applications où l'intérêt est dans la réflexion nécessaire à la mise en oeuvre du programme plutôt que dans le résultat du programme: par exemple les notions de **récursivité** et de **récurrence** sont liées et l'utilisation de l'une peut aider à la compréhension de l'autre, un bel exemple, en arithmétique sous LOGO, en est fourni par la brochure de R. Cuppens[11].

Il est difficile de savoir ce que l'introduction de travaux dirigés de mathématiques a réellement apporté aux étudiants. Ces derniers, interrogés ont des avis très variés sur la question: cela va de l'enthousiasme (souvent de la part de mordus de la micro-informatique) à l'hostilité ("cela ne sert pas directement à l'examen" ou "ce ne sont pas des vraies mathématiques"). Il est frappant de constater que l'introduction de la micro-informatique dans les lycées est loin d'être complète et que beaucoup de nos étudiants sont des débutants en la matière, ce qui complique notre tâche mais renforce l'intérêt d'une initiation à l'usage de micro-ordinateurs.

10 P. JARRAUD:

Enseignement de la Géométrie et Génération tangentielle
des courbes avec la Géométrie de la Tortue
Séville septembre 1987 Preprint

11 R. CUPPENS:

Apports de l'Informatique en Arithmétique
IREM de Toulouse (1986)

Au niveau de l'examen, les résultats sont difficiles à interpréter, les quatre sections parallèles n'étant pas forcément de niveau équivalent au départ et les critères de notation n'étant pas forcément homogènes. De plus le programme de premier cycle n'est peut-être pas idéal, et l'introduction de nouvelles techniques peut déplacer les centres d'intérêt. Qui oserait encore faire du calcul numérique comme on le pratiquait à grand renfort de tables de logarithmes il y a 20 ans dans les classes préparatoires? Ce serait dommage de limiter notre effort à une meilleure compréhension d'un programme dont certains points datent (penser notamment au problème de l'étude des équations différentielles).

Nous referons des séances sur machine cette année. Les conditions techniques seront meilleures: nous disposerons d'une salle de 15 micro-ordinateurs tous graphiques (avec l'apport bien agréable de la couleur) et munis de disques de durs ce qui nous permettra d'utiliser de nouveaux logiciels. Nous essaierons aussi de profiter de l'expérience acquise cette année (notamment au niveau des feuilles de travaux dirigés).

La tâche la plus importante à accomplir serait de se livrer à une **évaluation** de l'influence de ces séances sur machines.

La disquette (contenant les programmes cités dans ce cahier et le précédent):

C.Laurent et P.Jarraud:

Travaux dirigés de Mathématiques sur micro-ordinateurs

peut être obtenue auprès de:

IREM Paris-Sud Université Paris VII
55-56 3^e Etage 2 Place Jussieu
75251 PARIS CEDEX 05

des auteurs: Christine Laurent ou Pierre Jarraud
Université Pierre et Marie Curie
Institut de Mathématiques Pures et Appliquées
45-46 5^e Etage 2 Place Jussieu
75252 PARIS CEDEX 05

pour les sommes de 4 Francs sur place et de 20 francs par correspondance.