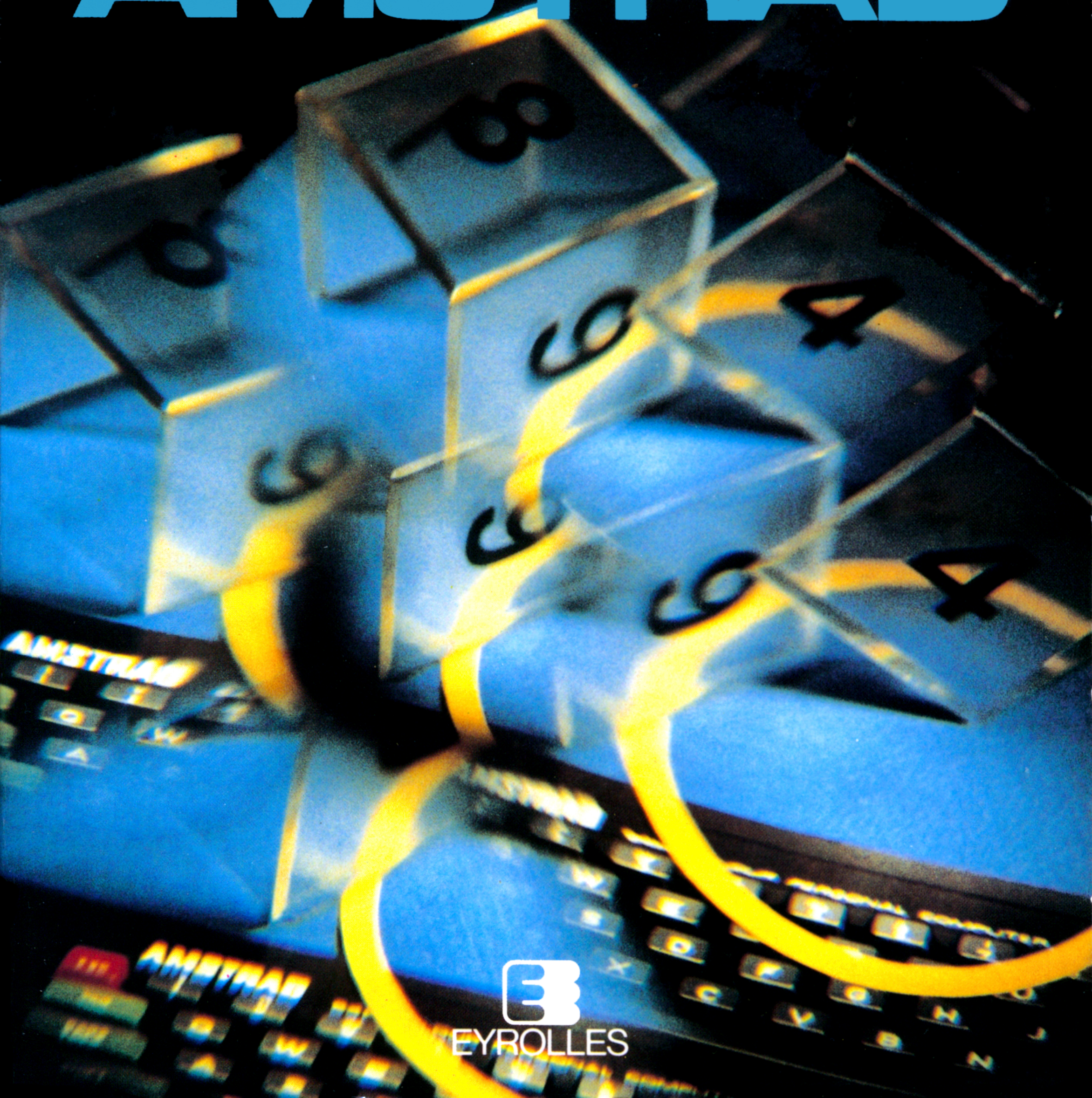
M. ROUSSELET

çalcul numérique sur AMSTRAD



EYROLLES

CALCUL NUMÉRIQUE

SUR AMSTRAD

CHEZ LE MEME EDITEUR

C. DELANNOY — *Faites vos jeux avec AMSTRAD.*

C. DELANNOY — *Je débute en BASIC AMSTRAD.*

P. BIHAN — *Programmation sur AMSTRAD PCW 8256 et 8512. BASIC et fichiers.*

J. MILSANT — *Lexique d'informatique et de micro-informatique avec index alphabétique anglais-français.*

P. BEAUFILS, M. LAMARCHE, Y. MUGGIANU
— *Programmes de physique sur AMSTRAD.*
— *Programmes de mathématiques sur AMSTRAD.*

C. DELANNOY — *Initiation à la programmation.*

O. LEPAPE — *L'assembleur facile du Z80.*

J.-P. DELAHAYE — *Dessins géométriques et artistiques avec votre micro-ordinateur.*
— *Nouveaux dessins géométriques et artistiques avec votre micro-ordinateur.*

Ph. DAX — *CP/M et sa famille. Guide d'utilisation.*

P. STUMM — *Jeux de robotique en BASIC.*

M.-G. MONTEIL, R. SCHOMBERG
— *Programmes d'intelligence artificielle en BASIC.*

M. JAMES — *Introduction à l'intelligence artificielle sur micro-ordinateur.*

CALCUL NUMÉRIQUE

SUR AMSTRAD

Michel ROUSSELET

EYROLLES

61, boulevard Saint-Germain — 75005 PARIS

1986

Si vous désirez être tenu au courant de nos publications, il vous suffit d'adresser votre carte de visite au :

Service « Presse », Editions EYROLLES,
61, Boulevard Saint-Germain
75240 PARIS CEDEX 05,

en précisant les domaines qui vous intéressent. Vous recevrez régulièrement un avis de parution des nouveautés en vente chez votre libraire habituel.

« La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause, est illicite » (alinéa 1er de l'article 40). »

« Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait une contrefaçon sanctionnée par les articles 425 et suivants du Code pénal. ».

© Editions EYROLLES, 1986

à Ph. ROUX et à C. DUBOIS,
membres éminents de
l'Académie Rhétaise.

Du même auteur

- *Mathématiques sur ZX-81 et ZX-Spectrum, E.T.S.F. Paris 1983.*
- *Graphismes en kits, E.T.S.F. Paris 1984.*
- *Graphisme 3D, E.T.S.F. Paris 1985.*

Avant-Propos

Après avoir longtemps utilisé les doigts des mains et des pieds, les hommes ont peu à peu développé des méthodes et des instruments de calcul.

Les Grecs utilisaient des abaques en marbre qu'ils transmirent aux Romains.

Ceux-ci remplacèrent les jetons par des petits cailloux (calculus en latin).

En Europe, abaques et bouliers demeurèrent en usage jusqu'au Moyen-Age.

Les calculs numériques sont cependant restés très longtemps laborieux et difficiles.

Par exemple, la résolution manuelle d'un simple système de cinq équations à cinq inconnues peut exiger jusqu'à 2 880 multiplications et 720 additions ou soustractions.

Il ne faut donc pas s'étonner si douze années de travail ont été nécessaires à l'astronome allemand Rhéticus pour constituer, au seizième siècle, une table de sinus à quinze décimales, avec un pas de dix secondes.

Tout a changé avec l'apparition des premiers calculateurs électroniques dans les années de l'après-guerre. Ces calculateurs n'existaient alors que dans les centres de calcul importants, surtout militaires.

Aujourd'hui les micro-ordinateurs personnels, portables ou de poche, permettent à chacun d'effectuer facilement des calculs complexes autrefois réservés aux seuls spécialistes : résolution d'équations de degré quelconque, calculs statistiques, résolution d'équations différentielles, ... etc.

Ainsi s'est presque réalisé ce que le mathématicien Jean Kuntzmann annonçait en 1967 : « il n'est pas utopique de penser que, dans dix ans, chaque utilisateur éventuel aura à sa disposition une « prise de calcul » utilisable dans les mêmes conditions qu'une prise de courant ou de téléphone ».

sommaire

Chapitre 1

Aperçu sur les erreurs de calcul

Le résultat d'un calcul est rarement exact mais l'erreur commise est souvent très difficile à évaluer. Elle dépend tout autant du matériel utilisé que de la méthode employée. Habituellement, on distingue trois catégories d'erreurs :

1) **Les erreurs de calcul proprement dites** proviennent de la propagation des erreurs d'arrondis. Celles-ci se produisent à chacune des étapes d'un calcul. Elles dépendent essentiellement de de la manière dont sont représentés les nombres utilisés et de la précision des algorithmes qui permettent de définir les habituelles fonctions du BASIC : racine carrée, sinus et cosinus, logarithmes, etc...

2) **Les erreurs de méthode** sont dûes à l'emploi répété de simplifications et d'approximations.

3) **L'imprécision** qui peut exister sur les données initiales est souvent la cause d'une erreur supplémentaire sur le résultat final d'un calcul.

1. La représentation des nombres

Nous ne traiterons que le cas de la **simple précision,** la multiprécision n'existant pas de manière usuelle sur les micro-ordinateurs.

1.1. Les deux types de représentation

Un nombre peut être conservé en RAM sous deux façons :

1) **Sous la forme décimale** habituelle, on dispose de 9 chiffres significatifs au total.

Exemples : 1,453 697 32; 101,457 687; 0,347 688 152.

2) **Sous la forme $\boldsymbol{a \times 10^b}$**, on dispose de 6 chiffres pour le nombre a et de deux chiffres pour le nombre b.

La valeur absolue de a est toujours comprise entre 1 et 10.

Selon le cas, b est compris entre -39 et -10 ou bien entre 10 et 39.

Exemples : 2,417 32 $E + 12$; $-$ 4,577 21 $E + 17$; 8,373 54 $E - 05$.

1.2. Étendue des registres

Dans le cas des nombres décimaux positifs, on peut utiliser tout nombre compris entre 3×10^{-39} et 10^{38} compte tenu des limitations qui existent sur le nombre des décimales.

1.3. L'arrondissement des nombres

Sauf exception, on ne travaille qu'avec des valeurs approchées.

Celles-ci peuvent être des **valeurs approchées par excès** ou bien des **valeurs approchées par défaut.**

Exemples : $\frac{2}{3}$ est représenté par 0,666 666 667 valeur approchée par excès; $\frac{1}{3}$ est représenté par 0,333 333 333, valeur approchée par défaut.

2. *La propagation des erreurs d'arrondi*

Les résultats intermédiaires d'un calcul étant arrondis, il en résulte que, de proche en proche, les erreurs d'arrondis vont s'accumuler.

Cette accumulation peut prendre deux formes extrêmes : les erreurs peuvent produire un effet de « boule de neige » mais elles peuvent aussi se compenser et donner une erreur finale très faible.

Ajoutons à cela le phénomène de la « différence évanescente », phénomène qui peut modifier considérablement le déroulement d'un calcul.

2.1. Effet boule de neige

Calculons $S_1 = \frac{2}{3} + \frac{4}{3} + ... + \frac{19\,998}{3} + \frac{20\,000}{3}$ à l'aide du programme suivant :

```
10 s=2/3
20 FOR n=2 TO 20000 STEP 2
30 s=s+n/3
40 NEXT
50 PRINT"s=";s
```

On obtient $S_1 = 33\,336\,667{,}3$ au lieu de $33\,333\,666{,}66...$

Les erreurs d'arrondi ont toutes **le même signe** et s'accumulent en s'ajoutant les unes aux autres.

Dans certains cas, cet effet « boule de neige » peut rendre le résultat final d'un calcul sans intérêt.

2.2. Compensation des erreurs

Calculons maintenant $S_2 = \frac{1}{3} + \frac{2}{3} + ... + \frac{10\,000}{3}$:

```
10 s=1/3
20 FOR n=2 TO 10000
30 s=s+n/3
40 NEXT
50 PRINT"s=";s
```

On obtient $S_2 = 16\,668\,333{,}3$: les neuf chiffres sont exacts.

Les erreurs de calcul se compensent mutuellement puisqu'elles sont **alternativement** positives et négatives. Il s'agit là, évidemment, d'un cas très favorable.

2.3. Les différences évanescentes

Si deux nombres x et y sont très voisins, le nombre $x - y$ peut être arrondi à 0 par l'ordinateur : la différence « s'évanouit ». Ce phénomène peut fausser complètement un calcul.

Exemple : soit $x = 10^{-9}$ et $y = \frac{1}{x} - \frac{1}{x+1}$.

Le calcul direct donne $y = 999\,999\,999$. En fait, $y = 10^{18}$: il suffit de remarquer que $y = \frac{(x+1) - x}{x(x+1)} = \frac{1}{x(x+1)}$.

3. Rôle des fonctions du BASIC

Les fonctions usuelles du BASIC (racine carrée, fonctions trigonométriques, fonctions logarithmiques, exponentielles) peuvent elles aussi donner lieu à des erreurs de calculs.

Les **tests**, souvent pratiqués dans les programmes, peuvent y être très sensibles. Nous le verrons sur un exemple.

3.1. Exemple

Déterminons, parmi les entiers compris entre 1 et 100, ceux qui sont des carrés parfaits.

La méthode est très simple : soit n un entier et x la racine carrée de n; on reconnait que x est un nombre entier si on a $x = \text{int}(x)$.

```
10 FOR n=1 TO 100
20 x=SQR(n)
30 y=INT(x)
40 IF y=x THEN PRINT n
50 NEXT
```

On obtient la liste suivante : 1, 4, 9, 16, 64 et 81.

L'ordinateur a « oublié » 25, 36, 49 et 100!

Ceci tient au fait que l'algorithme de calcul d'une racine carrée donne par exemple $\sqrt{25} = 5{,}000\,000\,001$ au lieu de 5, $\sqrt{100} = 10{,}000\,000\,003$ au lieu de 10.

Bien que 9 chiffres significatifs seulement soient affichés, il peut y en avoir deux de plus en mémoire dans certains cas. On peut vérifier ce fait en faisant afficher $x - \text{int}(x)$: pour $x = \sqrt{25}$, on obtient par exemple $1{,}86\,265 \times 10^{-9}$.

3.2. Quelles solutions ?

Dans l'exemple traité, on a deux possibilités :

- soit on change le test utilisé
- soit on élabore un algorithme de calcul plus précis pour les racines carrées.

Envisageons d'abord la première solution :

1) Le nouveau programme est le suivant :

```
10 FOR n=1 TO 100
20 x=SQR(n)
30 y=INT(x)
40 IF y*y=n THEN PRINT n
50 NEXT
```

Nous obtenons bien cette fois-ci 1, 4, 9, 16, 25, 36, 49, 64, 81 et 100.

L'élévation au carré semble avoir « effacé » l'erreur commise sur la racine carrée.

Malheureusement, cela ne dure pas. Par exemple, le programme ne donne pas 1 369 comme un carré parfait. On obtient en effet $\sqrt{1\,369} = 36{,}999\,999\,99$ d'où $y^2 = \text{int}(\sqrt{1\,369})^2 = 36^2 = 1\,296$. On a donc $y^2 \neq n$. Faisons une dernière tentative.

```
10 FOR n=1 TO 100
20 x=SQR(n)
30 y=INT(x+0.000005)
40 IF y*y=n THEN PRINT n
50 NEXT
```

Pour $n \leqslant 10\,000$, ce programme fonctionne correctement. Pour des valeurs supérieures, il n'a pas été testé.

2) Il existe un algorithme très simple et très précis pour calculer une racine carrée. A partir d'une approximation initiale x_1, on détermine une suite (x_i) définie par $x_{i+1} = \frac{1}{2}\left(x_i + \frac{n}{x_i}\right)$, n étant le nombre dont on veut calculer la racine.

Quel que soit le choix de $x_1 \neq 0$, on démontre que la suite (x_i) converge vers $\sqrt{n}$.

Prenons donc pour x_1 la valeur de $\sqrt{n}$ donnée par la fonction BASIC SQR(n) et reprenons le premier test. Nous obtenons le programme suivant :

```
10 FOR n=1 TO 1000
20 GOSUB 100
30 y=INT(x)
40 IF y=x THEN PRINT n
50 NEXT n
60 END
100 x=SQR(n)
110 FOR i=1 TO 6
120 a=n/x
130 x=0.5*(x+a)
140 NEXT
150 RETURN
```

Il fonctionne parfaitement mais est plus lent.

4. *Estimation de l'erreur*

Par définition, cette erreur est inconnue mais on peut essayer d'en trouver un majorant. Il n'existe pas et ne peut pas exister de méthode générale : chaque calcul est un cas particulier.

A titre d'exemple, nous montrerons un tel calcul à propos du calcul approché d'une dérivée.

Ce calcul est simple mais a une portée générale.

4.1. Méthode de calcul d'une dérivée

Soit f une fonction numérique définie et dérivable sur un intervalle I de $\mathbb{R}$.

Soient x et x' deux valeurs voisines de cet intervalle I.

Posons $x' = x + h$ et prenons $\dfrac{f(x+h) - f(x)}{h}$ comme valeur approchée de $f'(x)$.

Ceci donne lieu au programme suivant :

```
10 DEF FN f(x)=...
20 h=0.001
30 INPUT"x=";x
40 y1=FN f(x)
50 y2=FN f(x+h)
60 d=(y2-y1)/h
70 PRINT "f'(";x;")=";d
```

4.2. Calcul de l'erreur de méthode

Nous supposerons tous les calculs exacts. Considérons un développement limité à l'ordre 1 de $f(x + h)$ au voisinage de x. On peut écrire $f(x + h) = f(x) + hf'(x) + r$, r étant un nombre dépendant de x et de h qui tend vers 0 quand h tend lui-même vers 0.

Il en résulte $f'(x) = \dfrac{f(x+h) - f(x)}{h} - \dfrac{r}{h}$.

Supposons maintenant f'' bornée sur I : il existe alors un nombre positif M tel que $|f''(x)| \leqslant M$ pour tout nombre x de I. Dans ces conditions, on a $|r| \leqslant M \dfrac{h^2}{2}$ (voir chapitre 6).

L'erreur de méthode est donc inférieure à $M \dfrac{|h|}{2}$.

4.3. Calcul de l'erreur d'arrondi

Soient e et e' les erreurs d'arrondi commises respectivement sur $f(x + h)$ et $f(x)$. La valeur effectivement calculée n'est donc pas $\dfrac{f(x+h) - f(x)}{h} - \dfrac{r}{h}$ mais $\dfrac{f(x+h) + e - (f(x) + e')}{h} - \dfrac{r}{h}$ soit $\dfrac{f(x+h) - f(x)}{h} + \dfrac{e - e' - r}{h}$.

Comme $f(x + h)$ et $f(x)$ sont très voisins, on peut supposer $e = e'$.

L'erreur d'arrondi est donc majorée par $\dfrac{2|e|}{|h|}$.

4.4. Résultat final

L'erreur commise en remplaçant $f'(x)$ par $\dfrac{f(x+h) - f(x)}{h}$ est donc majorée par $M \dfrac{|h|}{2} + \dfrac{2|e|}{|h|}$.

Le calcul montre que cette erreur est **minimum** pour une certaine valeur $h_{\min}$ de h : $h_{\min} = 2\sqrt{\frac{|e|}{M}}$.

Ceci est confirmé par l'emploi du programme dans des cas connus.

Il s'agit là d'un phénomène très général : pour tout calcul numérique existent des conditions optimales.

5. Conclusion

Un calcul qui n'est pas accompagné d'une estimation de l'erreur commise ne peut avoir qu'une valeur indicative. On ne doit pas oublier, lorsque l'évaluation de l'erreur s'avère trop difficile, qu'il est toujours possible de procéder à une double vérification :

- vérification du calcul proprement dit
- vérification du domaine de validité du programme lui-même.

Rappelons également deux constats entrevus dans ce chapitre :

- au delà d'un certain volume de calcul, les erreurs d'arrondi finissent par l'emporter. Il vaut mieux envisager de changer de programme plutôt que de multiplier inutilement les calculs.
- l'évaluation de l'erreur ralentit toujours considérablement un programme de calcul.

Chapitre 2
Polynômes

Les polynômes font l'objet de calculs nombreux et variés. Dans ce chapitre nous aborderons principalement :

- la multiplication et la division des polynômes
- la construction des polynômes dits de Lagrange
- la résolution des équations du type $P(x) = 0$, P désignant un polynôme de degré quelconque.

Les polynômes utilisés dans l'ensemble du chapitre sont des **polynômes à coefficients réels**.

1. Calcul des valeurs d'un polynôme

Soit P un polynôme de degré n défini par $P(x) = a_0 x^n + a_1 x^{n-1} + \ldots + a_n$.
Le calcul de $P(x)$ peut être effectué directement ou grace à la **méthode de Horner.**

1.1. Emploi de l'instruction ↑

Soit, à titre d'exemple, $P(x) = 5x^4 - 4x^3 + 3x^2 - 2x + 1$. Avec les instructions ↑, DEF FN et FN, le programme de calcul est particulièrement simple à utiliser :

```
10 DEF FN p(x)=5*x↑4-4*x↑3+3*x↑2-2*x+1
20 INPUT "x=";x
30 y=FN p(x)
40 PRINT "p(";x;")=";y
```

Si on ne dispose pas des instructions DEF FN et FN on peut placer le calcul de $P(x)$ dans un sous-programme :

```
10 INPUT "x=";x
20 GOSUB 100
30 PRINT "p(";x;")=";y
40 END
100 y=5*x↑4-4*x↑3+3*x↑2-2*x+1
110 RETURN
```

1.2. Méthode de Horner

Cette méthode de calcul attribuée à Horner est due, en fait, à Newton.

Reprenons l'exemple de $P(x) = 5x^4 - 4x^3 + 3x^2 - 2x + 1$. On peut écrire

$$P(x) = 1 - 2x + 3x^2 - 4x^3 + 5x^4$$

d'où
$$P(x) = 1 + x\big(-2 + x(3 + x(-4 + 5x))\big)$$

puis
$$P(x) = \big(((5x - 4)x + 3)x - 2\big)x + 1.$$

Posons $c_4 = 5$, $c_3 = xc_4 - 4$, $c_2 = xc_3 + 3$, $c_1 = xc_2 - 2$ et $c_0 = xc_1 + 1$.

On obtient $P(x) = c_0$.

Cette méthode se généralise aisément à un polynôme P de degré quelconque :

```
10 INPUT "Degre du polynome=...";n
20 DIM a(n+1)
30 FOR i=0 TO n
40 PRINT "coeff.du terme de degre ";n-i;
50 INPUT  a(i)
60 NEXT i
70 INPUT "valeur de x";x
80 c=a(0)
90 FOR i=1 TO n
100 c=c*x+a(i)
110 NEXT i
120 PRINT "valeur de p(";x;")=";c
130 PRINT "un autre calcul ? (o/n)"
140 INPUT r$
150 IF r$="n" THEN END ELSE GOTO 70
```

1.3. Comparaison des différentes méthodes

En prenant encore $P(x) = 5x^4 - 4x^3 + 3x^2 - 2x + 1$, calculons $P(x)$ pour 1 000 valeurs de x choisies au hasard entre 0 et 1.

Les temps de calcul sont les suivants : 76 secondes pour le 1^er^ programme, 70 secondes pour le 2^e^ et 80 secondes pour le 3^e^.

Comme on le voit, la méthode de Horner n'est pas réellement plus rapide que les autres méthodes. Elle offre néanmoins deux avantages : elle permet tout d'abord de calculer facilement les polynômes dérivés; elle évite ensuite d'avoir à introduire $P(x)$ dans une ligne de programme.

2. Dérivation d'un polynôme

Soit P un polynôme défini par $P(x) = a_0 x^n + ... + a_n$. Si P' désigne le **polynôme dérivé** de P, P' est de degré $n - 1$. On a alors

$$P'(x) = b_0 x^{n-1} + b_1 x^{n-2} + ... + b_{n-1}$$

avec $b_i = (n - i)a_i$, $0 \leqslant i \leqslant n - 1$.

Dans le programme qui suit, on a utilisé la méthode de Horner pour calculer simultanément $P(x)$ et $P'(x)$:

```
10 INPUT "Degre du polynome p(x)";n
20 DIM a(n+1):DIM b(n)
30 FOR i=0 TO n
40 PRINT "coeff.du terme de degre ";n-i;
50 INPUT  a(i)
60 IF i<n THEN b(i)=(n-i)*a(i)
70 NEXT i
80 INPUT "valeur de x";x
90 c=a(0):d=b(0)
100 FOR i=1 TO n
110 c=c*x+a(i)
120 NEXT i
130 FOR i=1 TO n-1
140 d=d*x+b(i)
150 NEXT i
160 PRINT "p(";x;")=";c
170 PRINT "p'(";x;")=";d
180 PRINT "un autre calcul ? (o/n)"
190 INPUT r$
200 IF r$="n" THEN END ELSE GOTO 80
```

Exemple : soit $P(x) = x^4 - 3x^3 + 5x^2 - 2x + 1$.

Pour $x = 2$, $P(x) = 9$ et $P'(x) = 14$.

Remarque : la même méthode permettrait de calculer facilement $P''(x)$, $P^{(3)}(x)$,... etc.

3. Polynômes de Lagrange

Considérons n points distincts de coordonnées respectives x_1 et y_1, x_2 et y_2,..., x_n et y_n. Il existe un polynôme P unique, de degré $n - 1$ au plus, tel que $y_i = P(x_i)$, $1 \leqslant i \leqslant n$.

Ce polynôme est le **polynôme de Lagrange** associé aux points $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$.

Nous distinguerons deux problèmes :

- le calcul explicite du polynôme P
- le calcul de $P(x)$ pour x quelconque.

3.1. Calcul explicite de *P*

A titre d'exemple prenons $n = 3$ et posons $P(x) = a_0 x^2 + a_1 x + a_2$.

Pour trouver les coefficients a_0, a_1 et a_2, il nous faut résoudre le système

$$\begin{cases} x_1^2 a_0 + x_1 a_1 + a_2 = y_1 \\ x_2^2 a_1 + x_2 a_1 + a_2 = y_2 \\ x_3^2 a_2 + x_3 a_1 + a_2 = y_3 \end{cases}$$

Ce système est linéaire : on peut donc employer la méthode de résolution décrite au chapitre 10

Exemple : $(x_1, y_1) = (1,1)$, $(x_2, y_2) = (2,3)$, $(x_3, y_3) = (3,6)$.

On obtient $P(x) = \frac{x^2}{2} + \frac{x}{2}$.

3.2. Calcul de *P*(*x*)

Reprenons $n = 3$ et posons :

$$L_1(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)}, \quad L_2(x) = \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)}$$

et

$$L_3(x) = \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)}.$$

On voit que $L_1(x_2) = L_1(x_3) = 0$, $L_2(x_1) = L_2(x_3) = 0$, $L_3(x_1) = L_3(x_2) = 0$ tandis que $L_1(x_1) = L_2(x_2) = L_3(x_3) = 1$.

Posons alors $P(x) = y_1\ L_1(x) + y_2\ L_2(x) + y_3\ L_3(x)$: P est le polynôme cherché.

La méthode se généralise aisément à une valeur quelconque de n et permet ainsi un calcul de $P(x)$ sans avoir à expliciter le polynôme P lui-même.

```
10 CLS
20 PRINT "Quelle est la valeur de n";
30 INPUT n
40 DIM x(n):DIM y(n)
50 FOR i=1 TO n
60 PRINT "x";i;"=";
70 INPUT x(i)
80 PRINT "y";i;"=";
90 INPUT y(i)
100 NEXT i
110 CLS
120 INPUT "Valeur de x=";x
```

```
130 p=0
140 FOR i=1 TO n
150 l=1
160 FOR j=1 TO n
170 IF j=i THEN 190
180 l=l*(x-x(j))/(x(i)-x(j))
190 NEXT j
200 p=p+y(i)*l
210 NEXT i
220 PRINT "P(";x;")=";p
230 PRINT "un autre calcul ? (o/n)
240 INPUT r$
250 IF r$="n" THEN END ELSE GOTO 120
```

Exemple : soient $(x_1, y_1) = (1,1)$, $(x_2, y_2) = (2,3)$, $(x_3, y_3) = (3,8)$ et $(x_4, y_4) = (4,20)$.

On a $P(5) = 43$, $P\left(\frac{1}{2}\right) = -\frac{1}{8}$.

4. Opérations sur les polynômes

Laissant de côté l'addition et la soustraction des polynômes, nous ne traiterons que la **multiplication** et la **division euclidienne** des polynômes.

4.1. Multiplication

Soient $A(x) = a_0 x^n + a_1 x^{n-1} + ... + a^n$ et $B(x) = b_0 x^m + b_1 x^{m-1} + ... + b_m$.

Posons $C(x) = A(x)\,B(x)$: le polynôme C est de degré $n + m$. Les coefficients $c_0, c_1, c_2, ..., c_{n+m}$ de ce polynôme sont donnés par le programme suivant :

```
10 INPUT "Degre du 1er polynome ";n
20 INPUT "Degre du 2eme  polynome ";m
30 DIM a(n+m+1):DIM b(n+m+1)
40 PRINT "Coefficients du 1er polynome:"
50 FOR i=0 TO n
60     PRINT "a(";i;")=";
70     INPUT a(i)
80 NEXT i
90 PRINT "Coefficients du 2eme polynome:"
100 FOR j=0 TO m
110    PRINT "b(";j;")=" ;
120    INPUT b(j)
130 NEXT j
140 PRINT "Coefficients du produit :"
150 DIM c(n+m+1)
160 FOR k=0 TO n+m
170   FOR i=0 TO k
```

```
180      j=k-i
190      c(k)=c(k)+a(i)*b(j)
200    NEXT i
210    PRINT "c(";k;")=";c(k)
220 NEXT k
```

Exemple : $A(x) = 5x^4 + 3x^3 - 8x^2 + x - 1$ et $B(x) = -8x^3 + 5x^2 - 6x + 2$.
On obtient $C(x) = -40x^7 + x^6 + 49x^5 - 56x^4 + 67x^3 - 27x^2 + 8x - 2$.

4.2. Division

Rappelons sur un exemple comment s'effectue la division des polynômes.

Soit à diviser $6x^5 - 7x^4 - x^3 + x + 1$ par $2x^3 - 3x^2 + 1$:

$$\begin{array}{l|l}
6x^5 - 7x^4 - x^3 \qquad + x + 1 & 2x^3 - 3x^2 + 1 \\
 & 3x^2 + x + 1 \\
6x^5 - 9x^4 \qquad 3x^2 & \\
\hline
\quad 2x^4 - x^3 - 3x^2 + x + 1 & \\
\quad 2x^4 - 3x^3 \qquad + x & \\
\hline
\qquad 2x^3 - 3x^2 \qquad + 1 & \\
\qquad 2x^3 - 3x^2 \qquad + 1 & \\
\hline
\qquad\qquad\quad 0 &
\end{array}$$

On obtient $3x^2 + x + 1$ comme quotient et 0 comme reste.

On peut aisément généraliser cette méthode à des polynômes A et B de degrés respectifs n et m, avec $n \geqslant m$.

On a alors $A(x) = B(x)\,Q(x) + R(x)$: Q est un polynôme de degré $n - m$ tandis que R est un polynôme de degré au plus égal à $m - 1$. Les coefficients de Q et de R sont donnés par le programme suivant :

```
10 PRINT " 1er polynome:"
20 INPUT "Degre du 1er polynome ";n
30 DIM a(n+1)
40 FOR i=0 TO n
50 PRINT "coeff. du terme de degre ";n-i;
60 INPUT a(i)
70 NEXT i
80 PRINT " 2eme polynome:"
90 INPUT "Degre du 2eme  polynome ";m
100 IF m>n THEN PRINT "Le 2eme polynome a un ";
110 IF m>n THEN PRINT "plus eleve que le 1er"
120 IF m>n THEN END
130 DIM b(m+1)
140 FOR j=0 TO m
150     PRINT "coeff. du terme de degre ";m-j;
160 INPUT b(j)
170 NEXT j
```

```
180 PRINT "Coefficients du quotient:"
190 p=n-m
200 DIM c(p+1)
210 FOR k=0 TO p
220 c(k)=a(k)/b(0)
230 PRINT "coeff. du terme en x↑";p-k;"=";c(k)
240 FOR i=k+1 TO k+m
250 a(i)=a(i)-c(k)*b(i-k)
260 NEXT i
270 NEXT k
280 PRINT "Coefficients du reste :"
290 FOR i=p+1 TO n
300 PRINT "coeff. du terme en x^";n-i;"=";a(i)
310 NEXT i
```

5. Racines d'un polynôme

Soit P un polynôme de degré $n \geqslant 3$, à coefficients réels :

$$P(x) = a_0 x^n + a_1 x^{n-1} + \dots + a_n.$$

Pourvu que l'on considère qu'une racine d'ordre m équivaut à m racines simples, on peut affirmer qu'un tel polynôme a exactement n racines, réelles ou complexes.

Pour $n \geqslant 5$, il n'existe pas de formule de calcul qui donne explicitement ces racines. La **méthode de Bairstow** permet de les calculer de manière approchée.

5.1. Principe

Choisissons deux nombres p et q et divisons $P(x)$ par $x^2 + px + q$: nous obtenons $P(x) = (x^2 + px + q)\,Q(x) + R(x)$.

Q est un polynôme de degré $n - 2$ tandis que R est de degré 1 au plus. Il existe donc deux nombres $\alpha(p, q)$ et $\beta(p, q)$ dépendants de p et de q tels que $R(x) = \alpha(p, q)x + \beta(p, q)$. Deux cas peuvent se présenter :

1) si $\alpha(p, q) = \beta(p, q) = 0$ les racines de $x^2 + px + q$ sont également des racines de P.

2) dans le cas contraire, on doit remplacer p et q par deux nombres nouveaux p' et q' et recommencer le calcul.

Dès que deux racines de P ont été trouvées, on remplace P par Q et on recommence la recherche avec ce nouveau polynôme. Le degré de Q étant moindre que celui de P, on est assuré d'achever le calcul en un nombre fini d'étapes.

Pour élaborer le programme correspondant, il faut :

- déterminer les formules qui permettent d'exprimer les coefficients de Q et de R en fonction de p et de q.
- déterminer les formules qui permettent de remplacer p et q par p' et q' le cas échéant.

5.2. Calcul des coefficients de Q et de R

Posons

$$Q(x) = b_0 x^{n-2} + b_1 x^{n-3} + \dots + b_{n-2}.$$

En développant $(x^2 + px + q)Q(x)$ et en identifiant le résultat à $P(x)$, nous pouvons écrire

$$\begin{aligned} b_0 &= a_0, \quad b_1 = a_1 - b_0 p \\ b_i &= a_i - pb_{i-1} - qb_{i-2} \qquad \text{pour } 2 \leqslant i \leqslant n-2 \\ \alpha &= a_{n-1} - pb_{n-2} - qb_{n-3} \\ \beta &= a_n - qb_{n-2} \end{aligned} \tag{2.1}$$

Posons $b_{n-1} = \alpha$ et $b_n = a_n - p\, b_{n-1} - q\, b_{n-2}$. Les formules précédentes deviennent :

$$\begin{aligned} b_0 &= a_0, \quad b_1 = a_1 - b_0 p \\ b_i &= a_i - pb_{i-1} - qb_{i-2} \qquad \text{pour } 2 \leqslant i \leqslant n \\ \alpha &= b_{n-1}, \quad \beta = b_n + pb_{n-1} \end{aligned} \tag{2.2}$$

5.3. Remplacement de p et de q

Le système $\alpha(p, q) = \beta(p, q) = 0$ n'étant pas vérifié, il faut remplacer p par p' et q par q' : p' et q' doivent vérifier le système $\alpha(p', q') = \beta(p', q') = 0$.

Posons $p' = p + \Delta p$ et $q' = q + \Delta q$. La méthode de Newton-Raphson décrite au chapitre 11 nous permet de remplacer le système précédent par le système linéaire suivant :

$$\begin{cases} \alpha + \Delta p \dfrac{\partial \alpha}{\partial p} + \Delta q \dfrac{\partial \alpha}{\partial q} = 0 \\ \beta + \Delta p \dfrac{\partial \beta}{\partial p} + \Delta q \dfrac{\partial \beta}{\partial q} = 0 \end{cases} \tag{2.3}$$

Dans ce système $\dfrac{\partial \alpha}{\partial p}$, $\dfrac{\partial \alpha}{\partial q}$, $\dfrac{\partial \beta}{\partial p}$ et $\dfrac{\partial \beta}{\partial q}$ désignent respectivement les dérivées partielles des fonctions $\alpha(p, q)$ et $\beta(p, q)$ selon p et selon q.

5.4. Calcul des dérivées partielles

1) Puisque $\alpha = b_{n-1}$, on a $\dfrac{\partial \alpha}{\partial p} = \dfrac{\partial b_{n-1}}{\partial p}$.

En dérivant les relations (2.2), on obtient :

$\frac{\partial b_0}{\partial p} = 0, \frac{\partial b_1}{\partial p} = - b_0$

$\frac{\partial b_i}{\partial p} = - b_{i-1} - p\frac{\partial b_{i-1}}{\partial p} - q\frac{\partial b_{i-2}}{\partial p}$ pour $2 \leqslant i \leqslant n$.

Posons, pour $0 \leqslant i \leqslant n$, $c_i = \frac{\partial b_i}{\partial p}$. Il vient

$c_0 = 0, c_1 = - b_0$

$c_i = - b_{i-1} - p\, c_{i-1} - q\, c_{i-2}$ pour $2 \leqslant i \leqslant n$.

2) Puisque $\beta = b_n + pb_{n-1}$, on obtient $\frac{\partial \beta}{\partial p} = c_n + b_{n-1} + pc_{n-1}$ soit $\frac{\partial \beta}{\partial p} = - qc_{n-2}$

3) Calculons maintenant $\frac{\partial \alpha}{\partial q}$ et $\frac{\partial \beta}{\partial q}$. Puisque $\frac{\partial \alpha}{\partial q} = \frac{\partial b_{n-1}}{\partial q}$, on peut dériver par rapport à q les relations (2,2). On obtient :

$\frac{\partial b_0}{\partial q} = 0, \frac{\partial b_1}{\partial q} = 0$

$\frac{\partial b_i}{\partial q} = - p\frac{\partial b_{i-1}}{\partial q} - q\frac{\partial b_{i-1}}{\partial q} - b_{i-2}$ pour $2 \leqslant i \leqslant n$.

Posons, pour $0 \leqslant i \leqslant n$, $c'_i = \frac{\partial b_i}{\partial q}$. On voit que $c'_i = - b_{i-2} - p\, c'_{i-1} - qc'_{i-2}$ d'où, pour $2 \leqslant i \leqslant n$, $c'_i = c_{i-1}$.

On en tire $\frac{\partial \alpha}{\partial q} = c_{n-2}$.

Puisque $\frac{\partial \beta}{\partial q} = \frac{\partial b_n}{\partial q} + p\frac{\partial b_{n-1}}{\partial q} = c'_n + pc'_{n-1}$, on obtient $\frac{\partial \beta}{\partial q} = c_{n-1} + pc_{n-2}$.

5.5. Calcul de Δp et de Δq

En utilisant les expressions de $\frac{\partial \alpha}{\partial p}, \frac{\partial \alpha}{\partial q}, \frac{\partial \beta}{\partial p}$ et $\frac{\partial \beta}{\partial q}$ qui viennent d'être calculées, on peut transformer le système (2.3) en le suivant :

$$\begin{cases} \alpha + \Delta p\, c_{n-1} + \Delta q\, c_{n-2} = 0 \\ \beta - q\Delta p\, c_{n-2} - \Delta q(c_{n-1} + pc_{n-2}) = 0 \end{cases} \quad (2.4)$$

Posons $d = c_{n-1}^2 + c_{n-2}(pc_{n-1} + qc_{n-2})$,
$f = - b_{n-1}(qc_{n-2} + pc_{n-1}) - b_n c_{n-1}$ et $g = b_n c_{n-2} - b_{n-1} c_{n-1}$.

Si $d \neq 0$, le système (2.4) a pour solution $\Delta p = \frac{g}{d}$ et $\Delta q = \frac{f}{d}$.

5.6. Le programme

Soit $\varepsilon = 0{,}001$. Nous considèrerons que p et q sont des valeurs acceptables lorsque les inégalités $|\alpha(p, q)| < \varepsilon$ et $|\beta(p, q)| < \varepsilon$ sont vérifiées.

Une fois choisies les valeurs intiales de p et de q, en général 0 pour les 2 nombres, on effectue un maximum de 100 itérations pour trouver des valeurs convenables de p et de q. Si ces 100 itérations ne donnent rien, il faut choisir d'autres valeurs intiales pour q et q.

A noter que nous nous sommes limités aux racines réelles de l'équation $P(x) = 0$. Si l'on veut calculer les racines complexes, le programme est facile à modifier.

```
10 REM donnees
20 INPUT "degre du polynome ";n
30 DIM a(n+1):DIM b(n+1):DIM c(n+1)
40 FOR i=0 TO n
50 PRINT "coeff. du terme de degre ";n-i;
60 INPUT a(i)
70 NEXT i
80 s=0
90 IF n=2 THEN b(0)=1: b(1)=a(1)/a(0)
100 IF n=2 THEN b(2)=a(2)/a(0):GOTO 480
110  IF n=1 THEN b(0)=a(0):b(1)=a(1):GOTO 490
120 e=0.001
130 REM calcul de dp et de dq
140 INPUT "valeur de p ";p
150 INPUT "valeur de q ";q
160 j=1
170 b(0)=a(0):b(1)=a(1)-p*b(0)
180 c(0)=0:c(1)=-b(0)
190 FOR i=2 TO n
200 b(i)=a(i)-p*b(i-1)-q*b(i-2)
210 c(i)=-b(i-1)-p*c(i-1)-q*c(i-2)
220 NEXT i
230 d=c(n-1)*c(n-1)+c(n-2)*(p*c(n-1)+q*c(n-2))
240 f=-b(n-1)*(p*c(n-1)+q*c(n-2))-b(n)*c(n-1)
250 g=b(n)*c(n-2)-b(n-1)*c(n-1)
260 IF ABS(d)<0.000001 THEN PRINT "la methode ";
270 IF ABS(d)<0.000001 THEN PRINT"ne convient pas"
280 IF ABS(d)<0.000001 THEN END
290 dp=g/d:dq=f/d
300 IF ABS(dp)<e AND ABS(dq)<e THEN GOTO 380
310 REM calcul des nouvelles valeurs de p et de q
320 p=p+dp:q=q+dq
330 IF j<100 THEN j=j+1 :GOTO 170
340 PRINT "Trop d'iterations.Changez les ";
350 PRINT "valeurs initiales de p et de q"
360 END
370 REM resolution de x2+px+q=0
380 det=p*p-4*q
390 IF det<0 THEN GOTO 430
400 PRINT -p/2-SQR(det)/2:s=s+1
410 PRINT -p/2+SQR(det)/2:s=s+1
420 REM remplacement de p(x) par q(x)
430 n=n-2
```

```
440 FOR i=0 TO n
450 a(i)=b(i)
460 NEXT i
470 IF n>2 THEN 160
480 IF n=2 THEN p=b(1)/b(0):q=b(2)/b(0)
490 IF n=2 THEN GOTO 380
500 IF n=1 THEN PRINT -b(1)/b(0) :s=s+1
510 IF s=0 THEN PRINT "pas de solutions";
520 IF s=0 THEN PRINT "reelles"
```

5.7. Exemples

1) $P(x) = x^4 - 10x^3 + 35x^2 - 50x + 24$. Avec $p = q = 0$, on obtient 4 racines réelles :

$$x_1 = 1{,}000\,009\,61 \qquad x_3 = 3{,}001\,601\,1$$
$$x_2 = 1{,}999\,185\,99 \qquad x_4 = 3{,}999\,203\,29$$

les racines exactes sont 1, 2, 3 et 4.

2) $P(x) = x^5 + x^4 + x^3 - x^2 - x - 1$, $p = q = 0$. On obtient une racine réelle $x = 1{,}000\,000\,62$.

3) $P(x) = x^6 - 7x^5 + 7x^4 + 35x^3 + 56x^2 - 28x + 48$, $p = q = 0$. On obtient 6 racines réelles.

$$x_1 = -\,0{,}999\,999\,717 \qquad x_4 = 2{,}999\,984\,99$$
$$x_2 = 0{,}999\,991\,427 \qquad x_5 = 2{,}000\,018\,74$$
$$x_3 = -\,1{,}999\,999\,69 \qquad x_6 = 4{,}000\,004\,25$$

4) $P(x) = (x - 1)^4 = x^4 - 4x^3 + 6x^2 - 4x + 1$.

La méthode ne fonctionne ni avec $p = q = 0$, ni avec $p = q = 1$, ni avec $p = q = 2$.

5.8. Précision de la méthode

Soit r une racine du polynôme P et soit $\bar{r}$ une valeur approchée de r. Pour évaluer le nombre $|r - \bar{r}|$, on peut utiliser le polynôme dérivé de P.

1) Supposons $P'(\bar{r}) \neq 0$. On peut assimiler $P'(\bar{r})$ à $\dfrac{P(\bar{r}) - P(r)}{\bar{r} - r}$ c'est-à-dire à $\dfrac{P(\bar{r})}{\bar{r} - r}$.

On en tire $\bar{r} - r = \dfrac{P(\bar{r})}{P'(\bar{r})}$ d'où $|\bar{r} - r| \leqslant \left|\dfrac{P(\bar{r})}{P'(\bar{r})}\right|$.

2) Si $P'(\bar{r}) = 0$ la méthode précédente ne peut servir à évaluer $|\bar{r} - r|$. Ceci se produit, par exemple, quand r est racine multiple. Dans ce cas, la précision de la méthode de Bairstow est nettement moins bonne que dans le cas d'une racine simple.

Chapitre 3
Suites numériques

Soit $u_1, u_2, u_3,\ldots$ une suite de nombres réels dont le terme général sera désigné par u_n.

Une telle suite **converge** vers un nombre réel ℓ si et seulement si le nombre $|\ell - u_n|$ devient de plus en plus petit au fur et à mesure que n augmente.

On peut tenter de déterminer **expérimentalement** la limite d'une suite lorsque cette limite existe : pour cela, il suffit de calculer un assez grand nombre de termes de la suite. Il convient toutefois de rester très prudent sur les conclusions à tirer d'une telle méthode.

1. Exemple de suites convergentes

Nous emprunterons ces exemples aux formes de suites les plus usuelles : suites définies par une fonction numérique connue, suites **récurrentes**, suites **entrelacées.**

1.1. Suites du type $u_n = f(n)$

1) Soit $u_n = n \sin \frac{1}{n}$, avec $n \geqslant 1$. Calculons les 1 000 premiers termes de cette suite.

```
10 FOR n=1 TO 1000
20 u=n*SIN(1/n)
30 PRINT n,u
40 NEXT
```

On obtient $u_{30} = 0{,}999\,814\,825$, $u_{140} = 0{,}999\,991\,497$, $u_{800} = 0{,}999\,999\,740$: la suite converge vers 1.

2) Pour $n \geqslant 1$, posons $u_n = \dfrac{1 + \cos(\pi/3n)}{n \sin(\pi/4n)}$ et calculons les termes dont les numéros sont compris entre 6 000 et 6 999 :

```
10 FOR n=6000 TO 6999
20 a=1+COS(PI/3/n)
30 b=n*SIN(PI/4/n)
40 u=a/b
50 PRINT n,u
60 NEXT
```

Bien que tous les termes calculés soient égaux à 2,546 479 08, la suite n'est pas encore stationnaire. Elle le devient à partir de $n = 9\,008$.

La limite de cette suite est $\dfrac{8}{\pi} = 2{,}546\,479\,09$.

1.2. Suites du type $u_{n+1} = f(u_n)$

1) Soit, pour $n \geqslant 1$, $u_{n+1} = \sqrt{1 + u_n}$ avec $u_1 = 1$. Calculons les 1 000 premiers termes de la suite :

```
10 u=0.5
20 FOR n=2 TO 1000
30 u=(u+2/u)/3
40 PRINT n,u
50 NEXT
```

A partir de $n = 18$, u_n devient constant : la suite converge vers 1,618 033 99.

2) Posons $u_{n+1} = \dfrac{1}{3}\left(u_n + \dfrac{2}{u_n}\right)$, pour $n \geqslant 1$, avec $u_1 = 0{,}5$.

```
10 u=1
20 FOR n=2 TO 1000
30 u=SQR(1+u)
40 PRINT n,u
50 NEXT
```

A partir de $n = 20$, nous obtenons $u_n = 1$: la suite converge.

1.3. Suite du type $u_{n+2} = f(u_{n+1}, u_n)$

Prenons $u_1 = 1$, $u_2 = 2$ et posons $u_{n+2} = \dfrac{u_{n+1} + u_n}{2}$.

```
10 u1=1
20 u2=2
30 FOR n=3 TO 1000
40 u=(u1+u2)/2
50 PRINT n,u
60 u1=u2
70 u2=u
80 NEXT
```

La suite converge vers $\frac{5}{3} = 1{,}666\,666\,67$.

1.4. Suites entrelacées

Soient (u_n) et (v_n) deux suites définies par les relations $u_{n+1} = \sqrt{u_n v_n}$ et $v_{n+1} = \dfrac{u_n + v_n}{2}$ avec $u_1 = 2$ et $v_1 = 3$.

Il s'agit là d'un exemple de suite **arithmético-géométrique** de Gauss.

```
10 u1=2
20 v1=3
30 FOR n=2 TO 1000
40 u=SQR(u1*v1)
50 v=(u1+v1)/2
60 PRINT n,u,v
70 u1=u
80 v1=v
90 NEXT
```

A partir de $n = 4$, on peut constater que les deux suites ont la même limite (l'étude de l'influence du choix des nombres u_1 et v_1 sur cette limite a été faite par Gauss).

2. *Exemples de suites divergentes*

Une suite qui ne possède pas de limite est dite **divergente** : ceci peut se produire de plusieurs manières.

2.1. Les nombres de Fibonacci

Soient $u_1 = u_2 = 1$. Pour $n \geqslant 1$, posons $u_{n+2} = u_{n+1} + u_n$. Calculons les premiers nombres de la suite :

```
10 u1=1
20 u2=1
30 FOR n=3 TO 1000
40 u=u1+u2
```

```
50 PRINT n,u
60 u1=u2
70 u2=u
80 NEXT
```

La suite diverge : un dépassement de capacité se produit à partir de $n = 185$.

2.2. Suite périodique simple

Soit, pour $n \geqslant 1$, $u_{n+1} = \dfrac{5 - u_n}{1 + u_n}$, avec $u_1 = 2$.

La suite ainsi définie est **périodique** : on a tour à tour $u_n = 1$ et $u_n = 2$.

```
10 u=2
20 FOR n=2 TO 1000
30 u=(5-u)/(1+u)
40 PRINT n,u
50 NEXT
```

2.3. Un attracteur étrange

Soit, pour $n \geqslant 1$, $u_{n+1} = 4au_n(1 - u_n)$ avec $u_1 = 0{,}5$; a désigne un nombre réel quelconque.

```
10 INPUT"a=";a
20 u=0.5
30 FOR n=2 TO 1000
40 u=4*a*u*(1-u)
50 PRINT n,u
60 NEXT
```

Pour $a = 0{,}785$, la suite **oscille** entre deux valeurs : 0,780 464 117 et 0,538 007 221.

Pour $a = 0{,}87$, elle oscille entre quatre valeurs : 0,831 680 114; 0,487 159 285; 0,869 426 204 et 0,395 064 494.

Avec $a = 0{,}89$, on peut observer une oscillation entre 8 valeurs.

Ce type de suite, nommé **attracteur étrange,** est utilisé comme modèle de certains phénomènes de turbulence.

3. Contre-exemple

Par suite des erreurs d'arrondi qui interviennent à chaque étape de calcul, il est possible que la suite effectivement calculée n'ait pas le même comportement que la suite à étudier.

3.1. Premier exemple

Soit $u_{n+1} = u_n(3u_n - 1)$ avec $u_1 = \frac{2}{3}$. La suite est une suite constante puisque $u_n = \frac{2}{3}$ quel que soit $n \geqslant 1$.

Calculons les cent premiers termes de la suite grâce au programme suivant :

```
10 u=0.666666667
20 FOR n=2 TO 1000
30 u=u*(3*u-1)
40 PRINT n,u
50 NEXT
```

Avec $u_1 = 0{,}666\,666\,667$, la suite semble diverger puisqu un dépassement de capacité se produit à partir de $n = 27$.

Avec $u_1 = 0{,}666\,666\,666$, la suite semble osciller autour de 0, les termes étant alternativement positifs et négatifs.

Si on fait $u = \frac{2}{3}$ en ligne 10, on obtient une suite effectivement constante.

3.2. Second exemple

Considérons la suite définie par la relation de récurrence $u_{n+1} = a\, u_n + b$, relation dans laquelle a et b désignent des nombres réels quelconques.

On montre aisément que $u_{n+1} = a^n\left(u_1 - \frac{b}{1-a}\right) + \frac{b}{1-a}$. Il en résulte plusieurs possibilités :

- si $|a| < 1$, la suite converge vers $\frac{b}{1-a}$
- *si* $a \neq 1$ et si $u_1 = \frac{b}{1-a}$, la suite est stationnaire
- si $a = 1$ et si $b = 0$, la suite est également stationnaire.
- dans les autres cas, la suite diverge.

Le programme de calcul de u_{n+1} est le suivant :

```
10 INPUT"a=";a
20 INPUT "b=";b
30 INPUT"u1=";u
40 FOR n=2 TO 1000
50 u=a*u+b
60 PRINT n,u
70 NEXT
```

Prenons, par exemple, $a = 8$, $b = -5$ et $u_1 = \frac{5}{7}$: la suite est **stationnaire.**

Comme $\frac{5}{7}$ n'est pas décimal, le programme précédent ne peut fonctionner qu'avec une valeur approchée : 0,714 285 174 ou 0,714 285 715.

Dans l'un et l'autre cas, la suite effectivement calculée diverge. (Notons qu'il en va ainsi sur tout micro-ordinateur puisqu'il est impossible de représenter exactement $\frac{5}{7}$).

4. Vitesse de convergence

Soit (u_n) une suite qui converge vers une limite ℓ. Pour qu'un terme quelconque de la suite puisse être considéré comme une approximation convenable de ℓ, il faut que son rang ne soit pas trop élevé : la suite étudiée doit donc converger rapidement.

4.1. Extraction de racine carrée

Soit a un nombre positif. Pour $n \geqslant 1$, posons $u_{n+1} = \frac{1}{2}\left(\frac{a}{u_n} + u_n\right)$ avec $u_1 = \frac{a}{2}$.

Cette suite, déjà connue des Babyloniens, permet de calculer très rapidement le nombre $\sqrt{a}$:

```
10 INPUT "a=";a
20 u=a/2
30 FOR n=2 TO 1000
40 u=0.5*(a/u+u)
50 PRINT n,u
60 NEXT
```

Prenons par exemple $a = 2$: nous obtenons $\sqrt{2} = 1{,}414\,213\,56$ dès que n atteint la valeur 5. La **vitesse de convergence** de cette suite est très grande. Essayons de l'expliquer.

Posons $\varepsilon_1 = u_1 - \sqrt{a}$ $\quad \varepsilon_2 = u_2 - \sqrt{a}, \ldots$ $\varepsilon_{n+1} = u_{n+1} - \sqrt{a}$.

On obtient aisément la relation $\varepsilon_{n+1} = \frac{\varepsilon_n^2}{2u_n}$.

On en déduit que tous les nombres $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_{n+1}$ ont le même signe. Comme on montre facilement que $\varepsilon_{n+1} \leqslant \frac{\varepsilon_n}{2}$, on peut affirmer qu'à chaque nouveau pas l'erreur commise est au moins divisée par 2, d'où la grande vitesse de convergence.

4.2. La constante d'Euler

Examinons maintenant le cas d'une suite qui, contrairement à la précédente, converge très lentement. Soit, pour $n \geqslant 1$, $u_n = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n} - \log n$. Cette suite converge vers un nombre c appelé **constante d'Euler**. En 1769, Euler a montré que $c = 0{,}577\,215\,664\,901\,532\ldots$

Pour calculer c, posons $S_n = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}$.

On obtient $S_n = S_{n-1} + \frac{1}{n}$ d'où, pour $n \geqslant 2$, $u_n = S_n - \text{Log } n$.

Le programme de calcul est le suivant :

```
10 INPUT"n=";n
20 s=0
30 FOR i=1 TO n
40 s=s+1/i
50 NEXT
60 PRINT s-LOG(n)
```

Dans le tableau ci-dessous sont rassemblées quelques valeurs de u_n de même que le temps de calcul nécessaire à l'obtention de ces valeurs :

n	u_n	temps de calcul (en secondes)
2 000	0,577 465 646	11,07
4 000	0,577 340 670	21,14
8 000	0,577 278 141	41,47
16 000	0,577 246 819	82,65
32 000	0,577 231 176	164,59
128 000	0,577 219 479	657,27
512 000	0,577 216 528	2 638

Comme on peut le constater, la convergence de la suite est lente puisque seules 5 décimales sont connues après le calcul de 512 000 termes de la suite.

Il est illusoire de penser améliorer la précision en calculant encore plus de termes car les erreurs d'arrondi risquent alors de devenir très importantes. Il faut plutôt chercher à « **accélérer la convergence** ».

5. *Accélération de la convergence*

Reprenons le calcul de la constante d'Euler c et montrons, sur cet exemple, comment il est possible d'améliorer considérablement la précision du résultat en diminuant le temps du calcul. La méthode consiste à remplacer la suite (u_n) qui converge lentement vers c par une suite (v_n) qui converge plus rapidement vers c.

5.1. Première amélioration

Supposons qu'il existe un nombre a et un nombre ε_n tels que, pour tout $n \geqslant 1$,

$u_n = c + \frac{a}{n} + \varepsilon_n$. (1)

On peut alors écrire $u_{2n} = c + \frac{a}{2n} + \varepsilon_{2n}$ (2).

En résolvant le système formé des équations (1) et (2) par rapport à c et à a, on peut écrire $c = 2u_{2n} - u_n - (2\varepsilon_{2n} - \varepsilon_n)$. Sous réserve que le nombre $2\varepsilon_{2n} - \varepsilon_n$ tende vers 0 lorsque n tend vers l'infini, la suite $v_n = 2u_{2n} - u_n$ converge vers c. Par exemple, $v_{2\,000} = 2u_{4\,000} - u_{2\,000} = 2 \times 0{,}577\,340\,670 - 0{,}577\,465\,646 = 0{,}577\,215\,694$.

Dans ce résultat, 7 décimales sont exactes.

5.2. Seconde amélioration

Au lieu de prendre un développement limité à l'ordre 1, prenons un développement limité à l'ordre 2.

Supposons donc qu'il existe des nombres a, b et ε_n tels que, pour $n \geqslant 1$,

$$u_n = c + \frac{a}{n} + \frac{b}{n^2} + \varepsilon_n \quad (1).$$

En appliquant cette relation aux indices $2n$ et $3n$, on obtient :

– $u_{2n} = c + \frac{a}{2n} + \frac{b}{4n^2} + \varepsilon_{2n}$ (2).

– $u_{3n} = c + \frac{a}{3n} + \frac{b}{9n^2} + \varepsilon_{3n}$ (3).

En résolvant le système formé des équations (1), (2) et (3) par rapport à c, a et b on obtient $c \simeq \left(u_{3n} - \frac{u_{2n}}{3} - \frac{u_n}{6}\right)$.

Ainsi pour $n = 1\,000$, on a $c = 0{,}577\,215\,658$: l'erreur est inférieure à 10^{-8}.

5.3. Remarque

On peut mettre en évidence l'existence d'un développement limité de la forme $c + \frac{a}{n} + \frac{b}{n^2} + \ldots$ en étudiant expérimentalement la suite (u_n).

Cependant, l'étude expérimentale doit être légitimée par des moyens théoriques.

Chapitre 4
Séries numériques

Soit (u_n) une suite de nombres réels. Supposons $n \geqslant 1$ et posons $s_1 = u_1$, $s_2 = u_1 + u_2$, $s_3 = u_1 + u_2 + u_3$,... etc. Nous définissons ainsi une **série** de terme général u_n. Les nombres s_1, s_2, s_3,... etc sont des **sommes partielles** de la série.

(Habituellement, la somme partielle s_n est désignée par la notation $\sum_{k=1}^{k=n} s_k$ ou, plus simplement, par $\sum_1^n s_k$).

Une série peut **converger** ou **diverger**. Dans le premier cas, on peut tenter de déterminer expérimentalement sa limite en calculant les premières sommes partielles.

1. Exemples

Les exemples traités sont des exemples très classiques.

1.1. Les séries géométriques

Soit $s = \frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3} + ...$: le terme général de la série est $\frac{1}{2^n}$.

Il s'agit d'une **série géométrique** de raison $q = \frac{1}{2}$ qui converge très rapidement vers 1 :

```
10 INPUT"n=";n
20 FOR k=1 TO n
30 s=s+1/(2↑k)
40 PRINT k,s
50 NEXT
```

D'une manière générale, une série géométrique de raison q converge vers $\frac{u_1}{1-q}$ lorsque $|q| < 1$.

1.2. La série harmonique

Il s'agit de la série de terme général $\frac{1}{n}$. $s = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \ldots$

```
10 INPUT"n=";n
20 FOR k=1 TO n
30 s=s+1/k
40 PRINT k,s
50 NEXT
```

Avec ce programme, on obtient $s_{100} = 5{,}187\,377\,52$, $s_{1\,000} = 7{,}485\,470\,87$, $s_{10\,000} = 9{,}787\,605\,97$, $s_{100\,000} = 12{,}090\,146$: cette série n'est pas convergente, mais la divergence s'effectue très lentement.

1.3. La série harmonique alternée

Soit $s = \frac{1}{1} - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \ldots$ une série de terme général $\frac{(-1)^{n-1}}{n}$.

```
10 INPUT"n=";n
20 FOR k=1 TO n
30 a=k/2
40 b=INT(k/2)
50 IF a=b THEN e=-1 ELSE e=1
60 s=s+e/k
70 NEXT
80 PRINT s
```

On obtient $s_{100} = 0{,}688\,172\,18$, $s_{1\,000} = 0{,}692\,647\,431$, $s_{10\,000} = 0{,}693\,097\,198$ et $s_{20\,000} = 0{,}693\,122\,206$.
La série converge lentement vers $s = \text{Log } 2 = 0{,}693\,147\,181$.

1.4. La série de Leibniz

Soit $s = \frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \ldots$ une série de terme général $\frac{(-1)^{n-1}}{2n-1}$.

```
10 INPUT"n=";n
20 FOR k=1 TO n
30 a=k/2
40 b=INT(k/2)
50 IF a=b THEN e=-1 ELSE e=1
60 s=s+e/(2*k-1)
70 NEXT
80 PRINT s
```

La série de Leibniz converge très lentement vers $\frac{\pi}{4} = 0{,}785\,398\,163$.

On obtient $s_{100} = 0{,}782\,898\,227$, $s_{1\,000} = 0{,}785\,148\,166$, $s_{10\,000} = 0{,}785\,373\,172$.

1.5. Les séries de Riemann

Soit $s = \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \ldots$ une série de terme général $\frac{1}{n^2}$.

```
10 INPUT"n=";n
20 FOR k=1 TO n
30 s=s+1/k/k
40 NEXT
50 PRINT s
```

On obtient $s_{1\,000} = 1{,}643\,934\,57$, $s_{10\,000} = 1{,}644\,834\,08$, $s_{20\,000} = 1{,}644\,884\,09$.

La série converge lentement vers $\frac{\pi^2}{6} = 1{,}644\,934\,07$. Cette série est un cas particulier des **séries de Riemann** de terme général $\frac{1}{n^p}$ avec p entier. Ces séries convergent pour $p > 1$ (on désigne alors par $\zeta(p)$ la somme de la série; on définit ainsi pour $p > 1$ une fonction qui joue un rôle important en théorie des nombres).

2. Critères de convergence

Les exemples précédents montrent qu'il est parfois difficile de se prononcer sur l'existence et la valeur d'une limite. Nous rappellerons donc trois résultats utiles pour étudier la convergence éventuelle d'une série.

2.1. Limite du terme général

Si le terme général u_n d'une série ne tend pas vers 0, la série ne converge pas. Il est donc inutile de calculer s_n.

Soit par exemple la série de terme général $u_n = \frac{2^n + 1}{1 + n \log 2}$: u_n tend vers l'infini avec n. La série diverge (effectivement, $s_{100} = 3{,}642\,29 \times 10^{28}$).

2.2. Utilisation d'une intégrale

Posons $u_n = f(n)$: si la fonction f est positive et décroissante sur $[1, \infty[$, la série de terme général u_n converge si et seulement si le nombre $\int_1^{\infty} f(x)dx$ est fini. On peut calculer ce type d'intégrale par la méthode du chapitre 8 paragraphe 5.

Reprenons par exemple la série de Riemann de terme général $\frac{1}{n^2}$.

La fonction $x \mapsto \frac{1}{x^2}$ est positive et décroissante sur $[1, +\infty[$. $\int_1^{\infty} f(x)dx = 1$, donc la série de Riemann dont il est question converge.

2.3. Règle de d'Alembert

Appelons ℓ la limite pour n infini de $\left|\frac{u_n + 1}{u_n}\right|$: si $\ell < 1$, la série de terme général u_n converge. Pour $\ell > 1$, elle diverge.

Si $\ell = 1$, il faut une étude plus détaillée.

Soit, par exemple, $s = 1 + \frac{2}{3} + \frac{3}{3^2} + \frac{4}{3^3} + \ldots$ une série de terme général $\frac{n}{3^{n-1}}$.

$\left|\frac{u_{n+1}}{u_n}\right| = \frac{n+1}{3n}$ tend vers $\frac{1}{3}$ et la série converge.

```
10 INPUT"n=";n
20 FOR k=1 TO n
30 s=s+k/(3↑(k-1))
40 NEXT
50 PRINT s
```

On obtient très rapidement $s = 2{,}25$.

Considérons maintenant la série de somme $s = 1 + \frac{2^2}{2!} + \frac{3^3}{3!} + \ldots \frac{n^n}{n!} + \ldots$

$\left|\frac{u_{n+1}}{u_n}\right| = \left(1 + \frac{1}{n}\right)^n$ tend vers e quand n tend vers l'infini. Comme $e = 2{,}78\ldots$, la série diverge.

3. *Accélération de la convergence*

Comme pour les suites, on peut utiliser différentes méthodes pour diminuer le nombre des calculs nécessaires à l'obtention suffisamment précise de la somme d'une série convergente.

3.1. Exemple de la série de Leibniz

Puisque la série de Leibniz est convergente, nous pouvons regrouper les termes deux à deux **dans l'ordre où ils sont écrits.** Nous obtenons ainsi une nouvelle série $s = \left(1 - \frac{1}{3}\right) + \left(\frac{1}{5} - \frac{1}{7}\right) + \left(\frac{1}{9} - \frac{1}{11}\right) + \dots$ soit $s = \frac{2}{1 \times 3} + \frac{2}{5 \times 7} + \frac{2}{9 \times 11} + \dots$

```
10 INPUT "n=";n
20 FOR k=1 TO n STEP 2
30 s=s+1/(2*k-1)/(2*k+1)
40 NEXT
50 PRINT 2*s
```

On obtient les mêmes valeurs que précédemment, par exemple $s_{10\,000} = 0{,}785\,373\,172$, mais nous savons à présent que cette valeur est une approximation par défaut.

Utilisons maintenant un autre type de regroupement :

$$s = 1 - \left(\frac{1}{3} - \frac{1}{5}\right) - \left(\frac{1}{7} - \frac{1}{9}\right) - \dots,$$

soit
$$s = 1 - 2\left(\frac{1}{3 \times 5} + \frac{1}{7 \times 9} + \dots\right).$$

```
10 INPUT "n=";n
20 FOR k=1 TO n STEP 2
30 s=s+1/(2*k+1)/(2*k+3)
40 NEXT
50 PRINT 1-2*s
```

Nous obtenons cette fois des valeurs approchées par excès; par exemple, $s_{10\,000} = 0{,}785\,423\,16$. Prenons comme approximation de la limite cherchée la moyenne des deux nombres. Nous obtenons 0,785 398 166 soit 8 décimales exactes contre 3 précédemment.

3.2. Exemple de la série alternée

Nous avons $s = \frac{1}{1} - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \dots$.

Supposons que, pour tout entier $n \geqslant 1$, il existe des réels $\alpha_1, \alpha_2, \alpha_3, \dots$ tels que

$$s = s_n + \frac{\alpha_1}{n} + \frac{\alpha_2}{n^2} + \frac{\alpha_3}{n^3} + \dots .$$

$s_n + \frac{\alpha_1}{n}$ est une valeur approchée de s, $s_{2n} + \frac{\alpha_1}{2n}$ également.

Résolvons le système $\begin{cases} s = s_n + \dfrac{\alpha_1}{n} \\ s = s_{2n} + \dfrac{\alpha_1}{2n} . \end{cases}$

Nous obtenons $s = 2s_{2n} - s_n$.
Avec $s_{2\,000} = 0{,}692\,897\,246$ et $s_{1\,000} = 0{,}692\,647\,431$, la formule précédente donne $s = 0{,}693\,147\,061$: 6 décimales sont exactes contre 3 auparavant. (Naturellement pour utiliser cette méthode, il faut justifier théoriquement la formule $s = s_n + \frac{\alpha_1}{n} + \frac{\alpha_2}{n^2} + \ldots$).

4. Évaluation de l'erreur commise

En remplaçant s par s_n on commet une erreur de méthode ε_n :

$$\varepsilon_n = s - s_n = \sum_{n+1}^{\infty} u_k.$$

Montrons sur deux exemples comment on peut évaluer ε_n.

4.1. Cas d'une série géométrique

Reprenons l'exemple $s = \frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3} + \ldots$.

$\varepsilon_n = \sum_{n+1}^{\infty} \frac{1}{2^k}$. On montre facilement que $\sum_{n+1}^{p} \frac{1}{2^k} = \frac{1}{2^n} - \frac{1}{2^p}$ pour $p > n + 1$ d'où $\sum_{n+1}^{\infty} \frac{1}{2^k} = \frac{1}{2^n}$.

L'erreur commise est divisée par 2 à chaque nouveau pas de calcul. Pour avoir, par exemple $\varepsilon_n < 10^{-8}$, il suffira de prendre $n = 12$.

4.2. Cas d'une série de Riemann

Reprenons maintenant l'exemple $s = \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \ldots$ $\varepsilon_n = \sum_{n+1}^{\infty} \frac{1}{k^2}$. Pour obtenir une majoration de ε_n, on peut remarquer que $k(k-1) < k^2$ pour $k \geqslant 1$, donc $\frac{1}{k^2} < \frac{1}{k(k-1)}$. Or $\frac{1}{k(k-1)} = \frac{1}{k-1} - \frac{1}{k}$ pour $k \geqslant 2$.

Il en résulte que, pour p entier supérieur à $n + 1$,

$$\sum_{n+1}^{p} \frac{1}{k^2} < \frac{1}{n} - \frac{1}{p} \quad \text{d'où} \quad \sum_{n+1}^{\infty} \frac{1}{k^2} < \frac{1}{n}.$$

On obtient donc facilement $s - s_n = \varepsilon_n < \frac{1}{n}$.

Pour avoir, par exemple, $\varepsilon_n < 10^{-6}$ il faudrait prendre $n = 10^6$ (en réalité les erreurs d'arrondi deviendraient alors prépondérantes et la précision souhaitée ne serait pas atteinte).

5. Séries entières

On appelle **série entière** une série dont le terme général est de la forme $u_n(x) = a_n x^n$, a_n et x réels.

5.1. Rayon de convergence

La somme d'une série entière, lorsqu'elle existe, dépend du choix de x. On démontre qu'il existe un nombre ρ tel que la série soit convergente pour $|x| < \rho$, divergente pour $|x| > \rho$. Pour $|x| = \rho$ une étude particulière doit être faite.

ρ est le **rayon de convergence** de la série.

Dans la plupart des cas, on détermine ρ en calculant la limite du rapport $\dfrac{u_n(x)}{u_{n+1}(x)}$ pour n infini : $\rho = \lim\limits_{n\infty} \left|\dfrac{u_n(x)}{u_{n+1}(x)}\right|$

5.2. Exemples

1) Soit $s = 1 + 1!\,x + 2!\,x^2 + 3!\,x^3 + \ldots$

$$\left|\frac{u_n(x)}{u_{n+1}(x)}\right| = \left|\frac{n!\,x^n}{(n+1)!\,x^{n+1}}\right| = \frac{1}{(n+1)\,|x|}$$

donc $\rho = 0$.

2) soit $s = x + \dfrac{x^2}{2} + \dfrac{x^3}{3} + \dfrac{x^4}{4} + \ldots$

$$\left|\frac{u_n(x)}{u_{n+1}(x)}\right| = \left|\frac{x^n/_n}{x^{n+1}/n+1}\right| = \frac{n+1}{n}\,|x|$$

donc $\rho = 1$.

6. Calcul de constantes

En utilisant des séries, on peut calculer des constantes telles que π ou e.

6.1. Calcul de π

Utilisons un résultat dû à Euler :

$$\frac{\pi^4}{90} = \frac{1}{1^4} + \frac{1}{2^4} + \frac{1}{3^4} + ...$$

```
10 INPUT"n=";n
20 FOR k=1 TO n
30 s=s+1/(k↑4)
40 NEXT k
50 PRINT (90*s)↑0.25
```

Avec $n = 236$, on obtient $\pi = 3{,}141\,592\,64$.

6.2. Calcul de *e*

Quel que soit le réel x, la série $s = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + ...$ converge vers le nombre e^x.

Faisons $x = 1$: on a $e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + ...$

```
10 INPUT"n=";n
20 u=1
30 s=1
40 FOR k=1 TO n
50 u=u/k
60 s=s+u
70 NEXT k
80 PRINT s
```

La série converge rapidement : on obtient $e = 2{,}718\,281\,83$ à partir de $k = 11$.

Chapitre 5
Fonctions numériques

Rappelons que l'expression **fonction numérique** désigne une fonction à valeurs dans $\mathbb{R}$ ou $\overline{\mathbb{R}}$.

Dans ce chapitre nous utiliserons essentiellement

- des fonctions numériques d'une seule variable réelle
- des fonctions numériques de deux variables réelles.

1. Les fonctions disponibles

En BASIC standard, on dispose d'un certain nombre de fonctions pré-programmées.

1.1. Les fonctions trigonométriques

- $\cos(x)$, $\sin(x)$, $\tan(x)$ donnent respectivement le **cosinus,** le **sinus** et la **tangente** d'un angle x exprimé en radians.

(Si x_d et x_r désignent respectivement les mesures en degrés et en radians d'un angle x quelconque, rappelons que $\dfrac{x_d}{x_r} = \dfrac{180}{\pi}$)

• atn(x) donne Arctg x c'est-à-dire l'angle compris entre $-\pi/2$ et $\pi/2$ dont la tangente est x.

1.2. Exponentielles et logarithmes

exp(x) et log(x) donnent respectivement le nombre e^x et le logarithme népérien du nombre x.

Rappelons deux formules utiles :

pour $a > 0$, $a^x = e^{x \log a}$ et $\log_a x = \dfrac{\log x}{\log a}$.

En particulier, le **logarithme décimal** $\log_{10} x$ est donné par la formule $\log_{10} x = \dfrac{\log x}{\log 10}$.

1.3. Autres fonctions

- abs(x) donne la **valeur absolue** du nombre x
- int(x) donne la **partie entière** du nombre x c'est-à-dire le plus grand entier inférieur à x.
- sgn(x) donne 1,0 ou -1 selon que x est positif, nul ou négatif.
- sqr(x) donne, pour $x \geqslant 0$, la **racine carrée** du nombre x.

2. Valeur d'une fonction numérique

Pour calculer la valeur d'une fonction numérique, on peut utiliser les instructions *DEF FN* et *FN* si elles sont disponibles ou bien, à défaut, un sous-programme spécifique.

2.1. Fonction d'une variable

Prenons à titre d'exemple $f(x) = \dfrac{5 + 2 \sin x}{\cos x}$. Les deux programmes qui suivent permettent de calculer $f(x)$:

1)
```
10 DEF FN (x)=(5+2*SIN(x))/COS(x)
20 INPUT"x=";x
30 y=FN f(x)
40 PRINT y
```

2)
```
10 INPUT"x=";x
20 GOSUB 100
30 PRINT y
40 END
100 y=(5+2*SIN(x))/COS(x)
110 RETURN
```

2.2. Fonction de 2 variables

Soit par exemple $f(x, y) = 5x^2 + y^2 + 1$. On calcule facilement $f(x, y)$ grâce aux programmes suivants :

```
10 DEF FN (x,y)=5*x*x+y*y+1
20 INPUT"x=";x
30 INPUT"y=";y
40 z=FN f(x,y)
50 PRINT z
```

ou bien

```
10 INPUT"x=";x
20 INPUT"y=";y
30 GOSUB 100
40 PRINT z
50 END
100 z=5*x*x+y*y+1
110 RETURN
```

3. *Recherche d'une limite*

Soit f une fonction numérique d'une variable réelle. On peut essayer, avec prudence, d'utiliser l'ordinateur pour calculer lorsqu'elle existe la limite de f en $x = x_0$.

3.1. Premier exemple

Soit $f(x) = \dfrac{\sqrt{1+x} - \sqrt{1-x}}{\sin x}$, $x_0 = 0$.

Pour déterminer $\lim\limits_{x = x_0} f(x)$, nous calculons $f(x)$ pour des valeurs positives, puis négatives de plus en plus proches de 0.

Les résultats de ces calculs sont consignés dans le tableau suivant :

x	$f(x)$	x	$f(x)$
10^{-1}	1,002 926 21	-10^{-1}	1,002 926 21
10^{-2}	1,000 029 17	-10^{-2}	1,000 029 17
10^{-3}	1,000 000 56	-10^{-3}	1,000 000 56
10^{-4}	0,999 999 303	-10^{-4}	0,999 998 303
10^{-5}	0,999 984 332	-10^{-5}	0,999 984 332
10^{-6}	0,999 541 953	-10^{-6}	0,999 541 953

$f(x)$ tend vers 1 quand x tend vers 0. On peut constater un phénomène général : la précision du calcul commence par croître pour décroître par la suite. Ceci est dû au fait que les erreurs d'arrondi deviennent prépondérantes. Il ne sert donc à rien dans la recherche d'une limite de choisir des valeurs trop « proches » de x_0.

3.2. Second exemple

Soit $f(x) = \sqrt{x^2 - 2x - 1} - \sqrt{x^2 - 7x + 3}$, $x_0 = +\infty$
$f(10^3) = 2{,}503\,640\,65$; $f(10^4) = 2{,}500\,347\,14$; $f(10^5) = 2{,}5$; $f(10^6) = 2{,}499\,511\,72$; la limite est $\frac{5}{2}$.

Avec $x = 10^9$, on obtient $f(x) = 0$: cette valeur aberrante est due au phénomène des « différences évanescentes ».

4. *Maximums et minimums locaux*

Soit f une fonction numérique définie et continue sur un intervalle I. Il est souvent utile de rechercher le maximum et le minimum de f sur cet intervalle.

4.1. Cas d'une seule variable

Soit $x_{min} \leqslant x \leqslant x_{max}$. Après avoir divisé l'intervalle $[x_{min}, x_{max}]$ en n parties égales, nous calculons la valeur de f en chacun des points de la subdivision. La comparaison des différentes valeurs calculées est effectuée au fur et à mesure du calcul :

```
10  DEF FN f(x)=.........
20  n=100
30  fmax=-1E+30
40  fmin=1E+30
50  INPUT"x1="; x1
60  u=x1:v=x1
70  INPUT"x2="; x2
80  FOR i=0 TO n
90  x=x1+i*(x2-x1)/n
100 y=FN f(x)
110 IF y>fmax THEN u=x
120 IF y>fmax THEN fmax=y
130 IF y<fmin THEN v=x
140 IF y<fmin THEN fmin=y
150 NEXT
160 PRINT "maximum de f=";fmax
170 PRINT "maximum atteint pour x=";u
180 PRINT "minimum de f=";fmin
190 PRINT "minimum atteint pour x=";v
```

Exemple : soit $f(x) = x^3 - 3x^2 + 1$; sur l'intervalle $[-2,2]$, f a pour minimum $m = -19$ et pour maximum $M = 1$.

4.2. Cas de deux variables

Soient $x_{min} \leqslant x \leqslant x_{max}$ et $y_{min} \leqslant y \leqslant y_{max}$: nous diviserons ces intervalles en n_1 et n_2 parties égales respectivement.

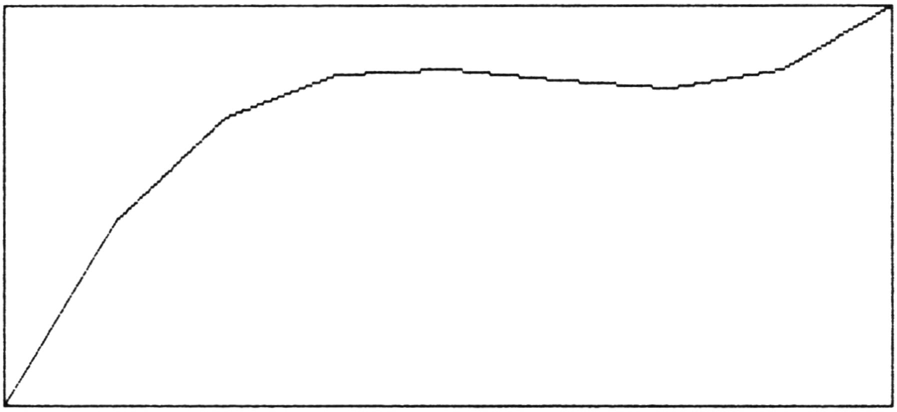
```
10 DEF FN f(x,y)=........
20 INPUT"xmin=";xmin
30 INPUT"xmax=";xmax
40 INPUT"ymin=";ymin
50 INPUT"ymax=";ymax
60 n1=20
70 n2=40
80 DIM x(n1+1):DIM y(n2+1)
90 FOR i=0 TO n1
100 x(i)=xmin+i*(xmax-xmin)/n1
110 NEXT i
120 FOR i=0 TO n2
130 y(i)=ymin+i*(ymax-ymin)/n2
140 NEXT i
150 zmax=-1E+30:zmin=1E+30
160 FOR i=0 TO n1
170 FOR j=0 TO n2
180 z=FN f(x(i),y(j))
190 IF z>zmax THEN k1=i:l1=j
200 IF z<zmin THEN k2=i:l2=j
210 IF z>zmax THEN zmax=z
220 IF z<zmin THEN zmin=z
230 NEXT j
240 NEXT i
250 PRINT"maximum=";zmax;
260 PRINT " en x=";x(k1);" et y=";y(l1)
270 PRINT"minimum=";zmin;
280 PRINT " en x=";x(k2);" et y=";y(l2)
```

Exemple : soit $f(x, y) = \dfrac{x^2 - 2y^2 + 1}{1 + x^2 + y^2}$

Pour $-4 \leqslant x \leqslant 4$ et $-5 \leqslant y \leqslant 5$, on obtient $m = -1{,}884\,615\,38$ et $M = 1$.

5. Représentation graphique d'une fonction d'une seule variable

Soit f une fonction numérique **définie** et **continue** sur un intervalle $[x_{min}, x_{max}]$. En divisant cet intervalle en n parties égales, on obtient les abcisses $x_1, x_2, \ldots, x_{n+1}$ de $n + 1$ points $P_1, P_2, \ldots, P_{n+1}$ de la courbe. On peut avoir une représentation approchée de f sur l'intervalle $[x_{min}, x_{max}]$ en reliant les points $P_1, P_2, ..., P_{n+1}$ par des segments de droite ainsi que l'indique la figure.



5.1. Cadrage du graphique

Sur l'écran nous inscrivons le graphe de f dans une fenêtre rectangulaire dont le coin inférieur gauche aura pour coordonnées u_{min} et v_{min} et dont le coin supérieur droit aura pour coordonnées u_{max} et v_{max}.

Soient y_{min} et y_{max} respectivement le minimum et le maximum de f sur l'intervalle étudié.

On peut associer un point (u, v) de l'écran à un point (x, y) du graphe de f en posant $u = k_1 x + \ell_1$ et $v = k_2 y + \ell_2$. Les nombres k_1, ℓ_1, k_2 et ℓ_2 vérifient alors les systèmes :

$$\begin{cases} u_{min} = k_1 x_{min} + \ell_1 \\ u_{max} = k_1 x_{max} + \ell_2 \end{cases} \quad \text{et} \quad \begin{cases} v_{min} = k_2 y_{min} + \ell_2 \\ v_{max} = k_2 y_{max} + \ell_2 \end{cases}$$

5.2. Programme

```
10 DEF FN f(x)=
20 REM intervalle de definition de x
30 INPUT "xmin=...";xmin
40 INPUT "xmax=...";xmax
50 REM calcul des points de subdivision de
60 REM  l'intervalle (xmin,xmax)
70 n=100
80 DIM x(n+1):DIM y(n+1)
90 ymax=-1E+30:ymin=1E+30
100 h=(xmax-xmin)/n
110 FOR i=0 TO n
120     x(i)=xmin+i*h
130     y(i)=FN f(x(i))
140     IF y(i)>ymax THEN ymax=y(i)
150     IF y(i)<ymin THEN ymin=y(i)
```

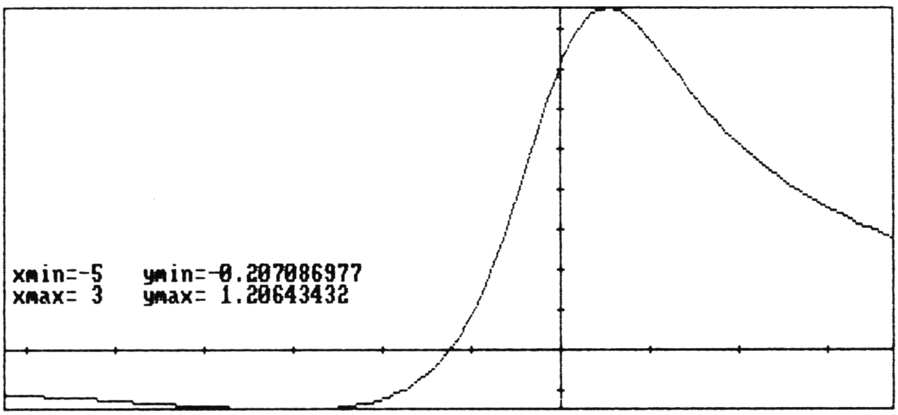
```
160 NEXT i
170 REM caracteristiques de la fenetre
180 umin=50:vmin=50
190 umax=600:vmax=350
200 k1=(umax-umin)/(xmax-xmin)
210 l1=umax-k1*xmax
220 k2=(vmax-vmin)/(ymax-ymin)
230 l2=vmax-k2*ymax
240 FOR i=0 TO n
250    x(i)=k1*x(i)+l1
260    y(i)=k2*y(i)+l2
270 NEXT i
280 REM dessin de la fenetre
290 CLS
300 MOVE umin,vmin:DRAW umax,vmin
310 DRAW umax,vmax:DRAW umin,vmax
320 DRAW umin,vmin
330 REM dessin eventuel des axes
340 IF xmin*xmax<=0 THEN GOSUB 480
350 IF ymin*ymax<=0 THEN GOSUB 590
360 REM dessin de la courbe
370 FOR i=0 TO n-1
380 PLOT x(i),y(i)
390 DRAW x(i+1),y(i+1)
400 NEXT i
410 LOCATE 15,24
420 PRINT "xmin=";xmin,"ymin=";ymin
430 LOCATE 15,25
440 PRINT "xmax=";xmax,"ymax=";ymax
450 END
460 REM sous-programme pour le dessin de l'axe
470 REM                                  des y
480 MOVE l1, vmin:DRAW l1,vmax
490 g=(vmax-vmin)/10
500 FOR i=1 TO 10
510    IF l2+i*g<vmax THEN MOVE l1-2,l2+i*g
520    IF l2+i*g<vmax THEN DRAW l1+2,l2+i*g
530    IF l2-i*g>vmin THEN MOVE l1-2,l2-i*g
540    IF l2-i*g>vmin THEN DRAW l1+2,l2-i*g
550 NEXT i
560 RETURN
570 REM sous-programme pour le dessin de l'axe
580 REM                                  des x
590 MOVE umin,l2:DRAW umax,l2
600 g=(umax-umin)/10
610 FOR i=0 TO 10
620    IF l1+i*g<umax THEN MOVE l1+i*g,l2-2
630    IF l1+i*g<umax THEN DRAW l1+i*g,l2+2
640    IF l1-i*g>umin THEN MOVE l1-i*g,l2-2
650    IF l1-i*g>umin THEN DRAW l1-i*g,l2+2
660 NEXT i
670 RETURN
```

D'une manière générale, les échelles ne sont pas les mêmes sur les deux axes. Lorsque ceux-ci sont représentés, les parties visibles comportent dix graduations régulièrement espacées.

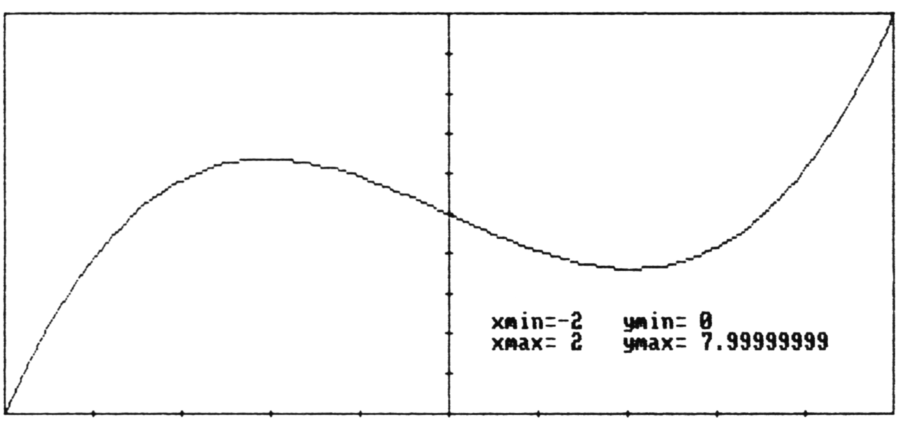
Sur l'axe des x, une graduation correspond donc à $\frac{x_{\max} - x_{\min}}{10}$; sur l'axe des y, une graduation correspond à $\frac{y_{\max} - y_{\min}}{10}$.

5.3. Exemples

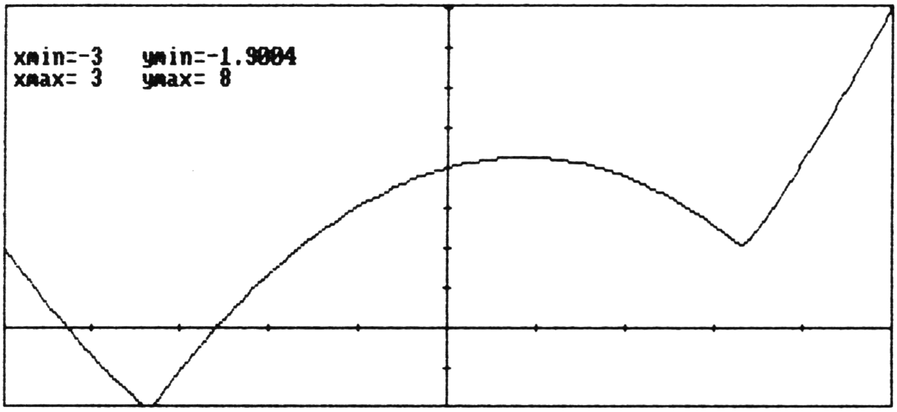
1) $f(x) = \frac{1 + x}{1 + x^2}, \ -5 \leqslant x \leqslant 3.$



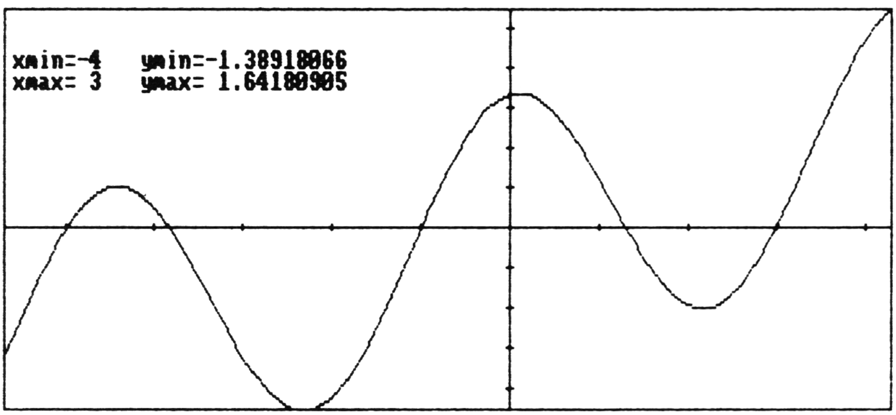
2) $f(x) = x^3 - 2x + 4, \ -2 \leqslant x \leqslant 2.$



3) $f(x) = x + |x^2 - 4|, \ -3 \leqslant x \leqslant 3.$



4) $f(x) = \cos 2x + \sin \frac{x}{4}, \; -4 \leqslant x \leqslant 3.$



6. *Représentation graphique d'une fonction de 2 variables réelles*

Soit f une fonction des variables réelles x et y, définie et continue pour $x_{\min} \leqslant x \leqslant x_{\max}$ et $y_{\min} \leqslant y \leqslant y_{\max}$.

Posons $z = f(x, y)$.

A tout point $P = (x, y, z)$ du graphe de f on peut associer un point $P' = (x', y')$ d'un plan de projection grâce à une **perspective cavalière.**

On a alors $x' = x + ky \cos \alpha$ et $y' = z' + ky \sin \alpha$, α étant l'**angle de fuite** de la perspective et k le **coefficient de réduction**. En général, on prend $\alpha = 45°$ et $k = \frac{1}{2}$.

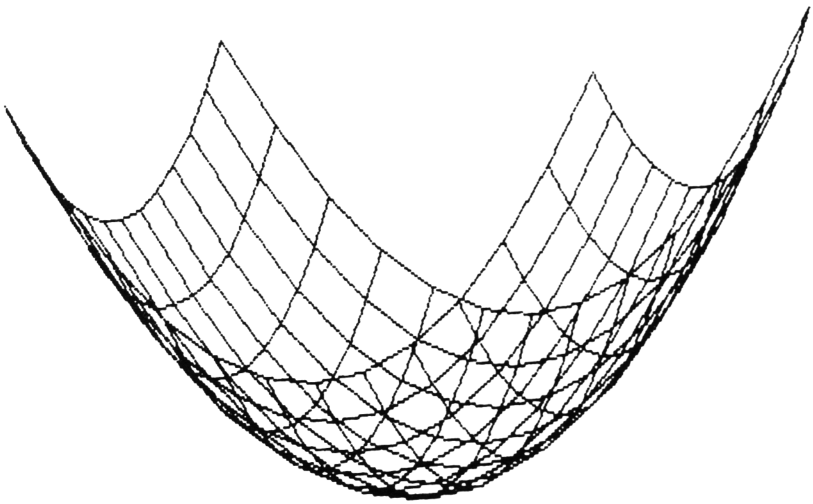
Le cadrage de l'image sur l'écran s'effectue par les même formules que pour la représentation graphique des fonctions d'une seule variable.

6.1. Programme

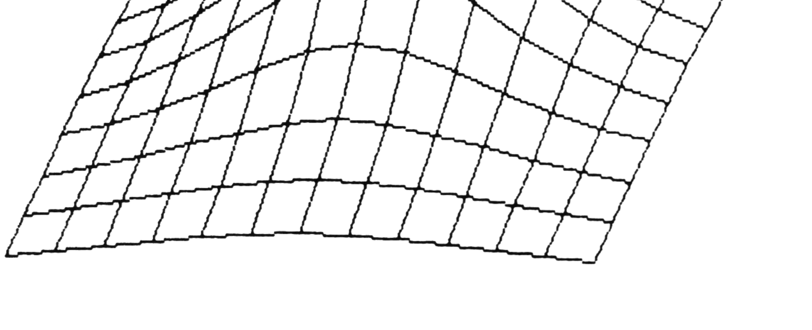
```
5 REM donnees
10 DEF FN z(x,y)=........
20 INPUT"xmin=...";xmin
30 INPUT "xmax=...";xmax
40 INPUT "ymin=...";ymin
50 INPUT "ymax=...";ymax
60 nx=12:ny=12
65 REM division des intervalles en parties
66 REM egales
70 DIM x(nx+1):DIM y(ny+1):DIM z(nx+1,ny+1)
80 FOR i=0 TO nx
90 x(i)=xmin+i*(xmax-xmin)/nx
100 NEXT i
110 FOR j=0 TO ny
120 y(j)=ymin+j*(ymax-ymin)/ny
130 NEXT j
140 FOR i=0 TO nx
150 FOR j=0 TO ny
160 z(i,j)=FN z(x(i),y(j))
170 NEXT j
180 NEXT i
185 REM calcul de la perspective
190 DIM v(nx+1,ny+1):DIM w(nx+1,ny+1)
200 vmax=-1E+30:vmin=1E+30
210 wmax=-1E+30:wmin=1E+30
220 FOR i=0 TO nx
230 FOR j=0 TO ny
240 v(i,j)=x(i)+0.375*y(j)
250 w(i,j)=z(i,j)+0.375*y(j)
260 IF v(i,j)<vmin THEN vmin=v(i,j)
270 IF v(i,j)>vmax THEN vmax=v(i,j)
280 IF w(i,j)<wmin THEN wmin=w(i,j)
290 IF w(i,j)>wmax THEN wmax=w(i,j)
300 NEXT j
310 NEXT i
315 REM adaptation de l'image aux dimensions
316 REM de l'ecran
320 k1=500/(vmax-vmin):k2=350/(wmax-wmin)
330 l1=501-k1*vmax:l2=351-k2*wmax
340 FOR i=0 TO nx
350 FOR j=0 TO ny
360 v(i,j)=k1*v(i,j)+l1
370 w(i,j)=k2*w(i,j)+l2
380 NEXT j
390 NEXT i
395 REM execution du dessin
400 CLS
410 FOR i=0 TO nx
420 FOR j=0 TO ny
430 IF i<=nx-1 THEN MOVE v(i,j),w(i,j)
440 IF i<=nx-1 THEN DRAW v(i+1,j),w(i+1,j)
450 IF j<=ny-1 THEN MOVE v(i,j),w(i,j)
460 IF j<=ny-1 THEN DRAW v(i,j+1),w(i,j+1)
470 NEXT j
480 NEXT i
```

6.2. Exemples

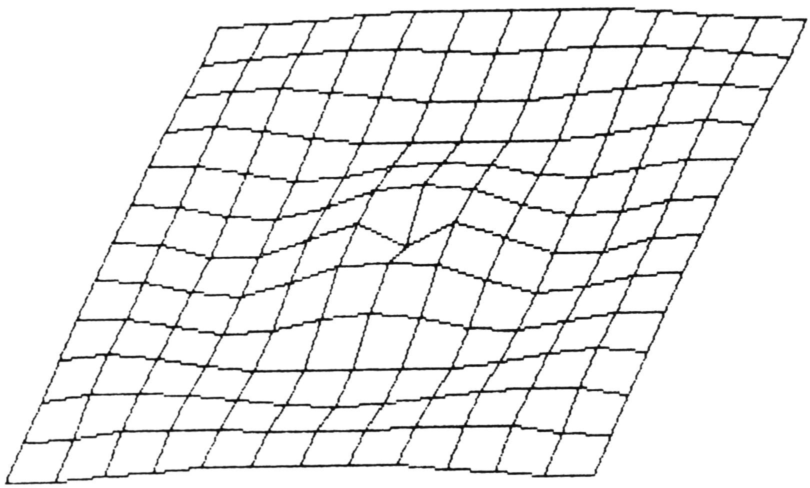
1) $f(x, y) = 2x^2 + y^2$ avec $-2 \leqslant x \leqslant 2$ et $-2 \leqslant y \leqslant 2$:



2) $f(x, y) = \dfrac{1}{x^2 + y^2 + 1}$ avec $-2 \leqslant x \leqslant 2$ et $-2 \leqslant y \leqslant 2$:



3) $f(x, y) = \dfrac{\sin \sqrt{x^2 + y^2}}{1 + \sqrt{x^2 + y^2}}$ avec $-8 \leqslant x \leqslant 8$ et $-8 \leqslant y \leqslant 8$:



Chapitre 6 Développements limités

Soit f une fonction numérique d'une variable réelle, définie et continue sur un intervalle I.

Soit x_0 un nombre appartenant à I.

En général, il existe un nombre positif α et un polynôme P_n de degré n tels que, pour $x_0 - \alpha \leqslant x \leqslant x_0 + \alpha$, les nombres $P_n(x)$ et $f(x)$ soient très peu différents.

P_n constitue alors un **développement limité à l'ordre n** pour la fonction f, **au voisinage** de x_0.

Dans le cas d'une fonction numérique de deux variables réelles, on retrouve la même possibilité; P est alors un polynôme de deux variables.

1. Cas d'une seule variable

Nous supposerons la fonction f **indéfiniment dérivable** sur I. Cette hypothèse est vérifiée par la quasi-totalité des fonctions usuelles.

1.1. Calcul de P_n

Ce polynôme est choisi de façon à vérifier les n relations suivantes :

- $P_n(x_0) \ = f(x_0)$
- $P'_n(x_0) \ = f'(x_0)$

$\vdots \qquad\qquad \vdots$

- $P_n^{(n)}(x_0) = f^{(n)}(x_0)$

Posons $P_n(x) = a_0 + a_1(x - x_0) + ... + a_n(x - x_0)^n$. En faisant $x = x_0$ dans cette égalité, on obtient $a_0 = f(x_0)$.

En dérivant ensuite P_n jusqu'à l'ordre n et en faisant chaque fois $x = x_0$, on peut écrire $a_i = \dfrac{f^{(i)}(x_0)}{i!}$ pour $1 \leqslant i \leqslant n$.

Il en résulte

$$P_n(x) = f(x_0) + \frac{x - x_0}{1!} f'(x_0) + ... + \frac{(x - x_0)^n}{n!} f^{(n)}(x_0).$$

Cette égalité constitue la **formule de Taylor.**

1.2. Évaluation de $f(x) - P_n(x)$

Posons $\varepsilon_n(x) = f(x) - P_n(x)$. On démontre qu'il existe un nombre ξ de l'intervalle I tel que $\varepsilon_n(x) = \dfrac{(x - x_0)^{n+1}}{(n+1)!} f^{(n+1)}(\xi)$.

Malheureusement, l'étude directe de $\varepsilon_n(x)$ n'est guère aisée aussi se contente-t-on le plus souvent d'un critère plus grossier.

Supposons $f^{(n+1)}$ **bornée** sur I : ceci veut dire qu'il existe un nombre positif M_{n+1} tel que $|f^{(n+1)}(x)| \leqslant M_{n+1}$ pour $x \in I$.

Il en résulte $|\varepsilon_n(x)| \leqslant M_{n+1} \dfrac{(x - x_0)^{n+1}}{(n+1)!}$.

Exemple : soit $f(x) = \sqrt{1 + x}$ avec $x \geqslant -1$. En calculant, au voisinage de $x = 0$, un développement limité à l'ordre 2 pour f, on obtient :

$$\sqrt{1 + x} = 1 + \frac{x}{2} - \frac{1}{8} x^2 + \varepsilon_2(x).$$

Prenons $0 \leqslant x \leqslant 1$: on a alors $|\varepsilon_2(x)| \leqslant \dfrac{x^3}{3!} f^{(3)}(\xi)$.

Le calcul montre que $f^{(3)}(\xi) = \dfrac{15}{8} \times \dfrac{1}{(\sqrt{1 + \xi})^7}$.

Pour $0 \leqslant \xi \leqslant 1$, $\dfrac{1}{(\sqrt{1 + \xi})^7} \leqslant 1$.

On a donc $|\varepsilon_2(x)| \leqslant \dfrac{5x^3}{16}$.

Ainsi pour $x = \frac{1}{2}$, $f(x) = 1{,}224\,744\,87$, $P_2(x) = 1{,}218\,750$ et $|\varepsilon_2| < 0{,}039$ (en réalité 0,006).

2. *Cas de deux variables*

Soit f une fonction numérique des variables réelles x et y.

Supposons que f admette des **dérivées partielles** de tous ordres.

On démontre que f admet des développements limités de tous ordres au voisinage d'un point (x_0, y_0).

2.1. Développements à l'ordre 1

Nous nous contenterons de rappeler la formule à l'ordre 1 seulement.

Cette formule est utilisée en particulier dans la méthode de Bairstow (voir chapitre 2).

On a $f(x, y) = f(x_0, y_0) + (x - x_0) f'_x(x_0, y_0) + (y - y_0) f'_y(x_0, y_0) + \ldots$

Dans cette formule f'_x et f'_y désignent respectivement les dérivées partielles de f à l'ordre 1 suivant x et suivant y.

2.2. Exemple

Soit $f(x, y) = x \cos y + xy + 1$

$f'_x = \cos y + y$ et $f'_y = -x \sin y + x$.

Au voisinage de (0, 0), on a donc $f(x, y) = 1 + x$.

3. *Application au calcul de constantes*

Calculons par exemple le nombre e et $\sin 50°$ à l'aide de développements limités.

3.1. Calcul de *e*

Soit $f(x) = e^x$. Quel que soit l'entier n, $f^{(n)}(x) = e^x$.

En développant e^x à l'ordre n, on peut donc écrire :

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + ... + \frac{x^n}{n!} + \frac{x^{n+1}}{(n+1)!} e^{\xi} \text{ avec } |\xi| < |x|.$$

Soit $x = 1$. On a alors :

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + ... + \frac{1}{n!} + \frac{e^{\xi}}{(n+1)!} \text{ avec } 0 < \xi < 1.$$

En prenant $1 + \frac{1}{1!} + ... + \frac{1}{n!}$ comme valeur approchée du nombre e, on commet une erreur égale à $\frac{e^{\xi}}{(n+1)!}$.

La fonction f étant croissante sur $\mathbb{R}$, on a pour $0 < \xi < 1$, $1 < e^{\xi} < e < 3$.

L'erreur commise dans le calcul de e est donc inférieure à $\frac{3}{(n+1)!}$.

Prenons par exemple $n = 8$: l'erreur commise est inférieure à $\frac{3}{9!} = 0{,}000\,008$.

Le calcul donne effectivement $e = 2{,}718\,278\,77$ au lieu de $2{,}718\,281\,83...$

3.2. Calcul de sin 50°

Pour tout réel x, on montre que :

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - ... + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + \varepsilon_n.$$

Le terme ε_n est égal à $(-1)^n \frac{x^{2n+3}}{(2n+3)!} \varphi(\xi)$, $\varphi(\xi)$ désignant la valeur en $x = \xi$ de la dérivée à l'ordre $n + 1$ de la fonction sinus.

Suivant les valeurs de n, φ peut être un sinus ou un cosinus : de toute façon $|\varphi(\xi)| \leqslant 1$.

Si on remplace sin x par un développement limité à l'ordre n, on commet une erreur inférieure à $\frac{|x|^{2n+3}}{(2n+3)!}$.

Prenons $x = \frac{50\,\pi}{180}$ soit 50°. Pour que l'erreur commise soit inférieure à, par exemple, 10^{-6} il faut choisir n de façon à avoir $\left(\frac{50\,\pi}{180}\right)^{2n+3} \times \frac{1}{(2n+3)!} \leqslant 10^{-6}$.

Grâce à l'ordinateur lui-même, on vérifie facilement qu'il suffit de prendre $n = 4$.

On obtient alors $\sin 50° = 0{,}766\,043\,64$ (au lieu de $0{,}766\,044\,443...$).

4. Application au graphisme

Pour obtenir sur l'écran de visualisation une **courbe,** il faut en général utiliser l'**équation** de cette courbe.

On peut accélérer considérablement le tracé en remplaçant l'équation représentative de la courbe par un développement limité à un ordre suffisamment élevé.

4.1. Cas d'un cercle

Soit C un cercle de centre $0 = (x_c, y_c)$ et de rayon R.

Relativement à un repère orthonormé d'origine 0, ce cercle a pour **équations paramétriques** $x = x_c + R \cos t$ et $y = y_c + R \sin t$ avec t variant de 0 à 2π.

Quitte à effectuer une translation, il suffit de s'intéresser au cas $x_c = y_c = 0$.

1) Considérons deux points voisins $P_1 = (x_1\ y_1)$ et $P_2 = (x_2, y_2)$ d'un cercle de rayon R, centré à l'origine du repère.

$x_1 = R \cos t_1$, $y_1 = R \sin t_1$, $x_2 = R \cos t_2$ et $y_2 = R \sin t_2$.

Posons $t_2 = t_1 + h$ et remplaçons les fonctions sinus et cosinus par des développements limités à l'ordre 2.

On obtient :

$$x_2 = R \cos (t_1 + h) = R\left(\cos t_1 - h \sin t_1 - \frac{h^2}{2} \cos t_1\right)$$

$$y_2 = R \sin (t_1 + h) = R\left(\sin t_1 + h \cos t_1 - \frac{h^2}{2} \sin t_1\right).$$

On en tire :

$$x_2 = R \cos t_1\left(1 - \frac{h^2}{2}\right) - hR \sin t_1$$

$$y_2 = hR \cos t_1 + \left(1 - \frac{h^2}{2}\right)R \sin t_1$$

soit
$$x_2 = \left(1 - \frac{h^2}{2}\right)x_1 - hy_1$$

et
$$y_2 = hx_1 + \left(1 - \frac{h^2}{2}\right)y_1$$

Ces formules permettent de proche en proche de déterminer une suite $P_1, P_2, \ldots, P_n$ de points du cercle.

En reliant ces points par des segments de droite, on pourra tracer le cercle de manière approchée.

2) Les paramètres *h* et *n* ont été choisis empiriquement de façon à avoir un tracé satisfaisant.

Le programme est le suivant :

```
10 INPUT"abscisse du centre";xc
20 INPUT"ordonnee du centre";yc
30 INPUT"rayon";r
40 x1=r:y1=0
50 FOR i=1 TO 45
60 x2=0.99*x1-0.141*y1
70 y2=0.99*y1+0.141*x1
80 PLOT xc+x1,yc+y1
90 DRAW xc+x2,yc+y2
100 x1=x2:y1=y2
110 NEXT
```

La durée d'exécution est inférieure à 2 *s*.

4.2. Cas d'une ellipse

Le même principe a été appliqué à une ellipse de centre (x_c, y_c), de grand axe $2a$, de petit axe $2b$.

Le programme est le suivant :

```
10 INPUT"abscisse du centre";xc
20 INPUT"ordonnee du centre";yc
30 INPUT"grand axe";a:a=a/2
40 INPUT"petit axe";b:b=b/2
50 x1=a:y1=0
60 FOR i=1 TO 45
70 x2=0.99*x1-0.141*y1*a/b
80 y2=0.99*y1+0.141*x1*b/a
90 PLOT xc+x1,yc+y1
100 DRAW xc+x2,yc+y2
110 x1=x2:y1=y2
120 NEXT
```

Chapitre 7
Dérivation numérique

Soit f une fonction numérique d'une variable réelle, définie et continument dérivable sur un intervalle I de $\mathbb{R}$.

Il est possible de calculer, pour $x \in I$, les valeurs des dérivées successives f', f'', ..., $f^{(n)}$ de la fonction f en employant d'autres moyens que les règles habituelles de dérivation; cependant nous nous limiterons essentiellement au calcul de f' et de f''.

1. Calcul des dérivées à droite

Soit $x \in I$. Par définition, f admet **une dérivée** f'_d **à droite** de x si le rapport $\dfrac{f(x+h) - f(x)}{h}$ tend vers une limite finie ℓ quand h tend vers 0 en restant positif.

On a alors $f'_d(x) = \ell$.

Les définitions des nombres $f''_d(x)$, $f'''_d(x)$,... etc se déduisent facilement de celle-ci.

1.1. Calcul de $f'_d(x)$ et de $f''_d(x)$

En prenant par exemple $h = 0{,}01$ on peut adopter $\dfrac{f(x+h) - f(x)}{h}$ comme valeur approchée de $f'_d(x)$. Cependant la médiocre précision de cette formule fait qu'on préfère utiliser les développements limités pour calculer $f'_d(x)$ *et* $f''_d(x)$.

Au voisinage de x, on peut écrire :

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(x) + \frac{h^3}{6} f'''(x) + \dots \quad (1)$$

$$f(x+2h) = f(x) + 2hf'(x) + \frac{4h^2}{2} f''(x) + \frac{8h^3}{6} f'''(x) + \dots \quad (2)$$

Retranchons membres à membres 4 fois l'égalité (1) à l'égalité (2). Il vient :

$$f(x+2h) - 4f(x+h) = -3f(x) - 2h\,f'(x) + \frac{4h^3}{6} f'''(x) + \dots$$

d'où $$f'(x) = \frac{4f(x+h) - f(x+2h) - 3f(x)}{2h} + \frac{h^2}{3}\,(f'''(x) + \dots)$$

Retranchons maintenant l'égalité (2) à 2 fois l'égalité (1). Il vient :

$$2f(x+h) - f(x+2h) = f(x) - h^2 f''(x) - h^3 f'''(x) + \dots$$

d'où $$f''(x) = \frac{f(x) + f(x+2h) - 2f(x+h)}{h^2} - h\,f'''(x) + \dots$$

Finalement nous obtenons pour $f'_d(x)$ et $f''_d(x)$ les expressions suivantes :

- $f'_d(x) = \dfrac{4f(x+h) - f(x+2h) - 3f(x)}{2h} + \dfrac{h^2}{3}\,\varphi(x)$

- $f''_d(x) = \dfrac{f(x) + f(x+2h) - 2f(x+h)}{h^2} + h\,\psi(x).$

Dans la mesure où les nombres inconnus $\frac{h^2}{3}\,\varphi(x)$ et $h\psi(x)$ restent négligeables, les formules ci-dessus peuvent être considérées comme des valeurs approchées de $f'_d(x)$ et de $f''_d(x)$.

```
10 DEF FN f(x)=.......
20 INPUT"x=";x
30 h=0.001
40 y1=FN f(x)
50 y2=FN f(x+h)
60 y3=FN f(x+2*h)
70 d1=(4*y2-y3-3*y1)/2/h
80 d2=(y1+y3-2*y2)/h/h
90 PRINT "derivee 1ere=";d1
100 PRINT "derivee 2eme=";d2
```

1.2. Exemple

Soit $f(x) = |x^2 - 1|$. Prenons $h = 0{,}001$ et $x = 1$. On obtient :

$$f'_d(1) = 2{,}000\,000\,33 \quad \text{et} \quad f''_d(1) = 1{,}999\,549\,57.$$

Les valeurs exactes sont égales à 2 dans chaque cas.

2. Calcul des dérivées à gauche

Pour définir $f'_g(x)$, il faut cette fois considérer des rapports du type $\dfrac{f(x) - f(x-h)}{h}$ avec $h > 0$.

2.1. Calcul de $f'_g(x)$ et de $f''_g(x)$

La même démarche que précédemment permet d'écrire les deux égalités suivantes :

- $f'_g(x) = \dfrac{3f(x) - 4f(x-h) + f(x-2h)}{2h} + \dfrac{h^2}{3}\,\varphi(x)$

et • $f''_g(x) = \dfrac{f(x) - 2f(x-h) + f(x-2h)}{h^2} + h\,\psi(x)$.

Ces égalités fournissent des valeurs approchées de $f'_g(x)$ et de $f''_g(x)$.

```
10 DEF FN f(x)=.......
20 INPUT"x=";x
30 h=0.001
40 y1=FN f(x)
50 y2=FN f(x-h)
60 y3=FN f(x-2*h)
70 d1=(3*y1-4*y2+y3)/2/h
80 d2=(y1+y3-2*y2)/h/h
90 PRINT "derivee 1ere=";d1
100 PRINT "derivee 2eme=";d2
```

2.2. Exemple

Soit $f(x) = |x^2 - 1|$. Calculons $f'_g(1)$ et $f''_g(1)$.

Les formules du paragraphe 2.1. donnent :

$$f'_g(1) = -1{,}999\,999\,63 \quad \text{et} \quad f''_g(1) = -1{,}999\,549\,57$$

(les valeurs exactes sont -2 pour $f'(1)$ et pour $f''(1)$).

3. Calcul des dérivées centrales

Soit $x \in I$. Si $f'_d(x) = f'_g(x)$, on note $f'(x)$ le nombre ainsi défini. $f'(x)$ est appelé **dérivée centrale** de f pour la valeur x.

On définit de même $f''(x)$, $f'''(x)$,...

3.1. Principe du calcul

Choisissons n nombres positifs h_1, h_2,..., h_n de telle manière que les nombres $x + h_1$, $x + h_2$,... $x + h_n$ restent « voisins » de x.

Écrivons les développements à l'ordre n des nombres $f(x + h_1)$, $f(x + h_2)$,..., $f(x + h_n)$.

On obtient :

$$f(x + h_1) = f(x) + h_1 f'(x) + ... + \frac{h_1^n}{n!} f^{(n)}(x)$$

$$\vdots$$

$$f(x + h_n) = f(x) + h_n f'(x) + ... + \frac{h_n^n}{n!} f^{(n)}(x)$$

L'ensemble de ces relations constitue un système linéaire de n équations à n inconnues, les nombres $f'(x)$, $f''(x)$, ..., $f^{(n)}(x)$.

3.2. Influence du choix de *n*

Les expressions obtenues pour $f'(x)$, $f''(x)$,..., $f^n(x)$ dépendent essentiellement du choix de n.

Ainsi les formules donnant $f'(x)$ seront différentes selon qu'on aura pris $n = 2$ ou $n = 4$.

3.3. Calcul de $f'(x)$ et de $f''(x)$

1) Prenons $n = 2$, $h_1 = h$, $h_2 = -h$.

On obtient alors

$$f'(x) = \frac{f(x + h) - f(x - h)}{2h}$$

$$\text{et} \quad f''(x) = \frac{f(x + h) + f(x - h) - 2f(x)}{h^2}$$

Il en résulte le programme suivant :

```
10 DEF FN f(x)=.......
20 INPUT"x=";x
30 h=0.001
40 y1=FN f(x-h)
50 y2=FN f(x)
60 y3=FN f(x+h)
70 d1=(y3-y1)/2/h
80 d2=(y3+y1-2*y2)/h/h
90 PRINT"derivee 1ere=";d1
100 PRINT"derivee 2eme=";d2
```

2) Prenons maintenant $n = 4$, $h_1 = -2h$, $h_2 = -h$, $h_3 = h$ et $h_4 = 2h$.

Le système qui en résulte permet de calculer $f'(x)$, $f''(x)$, $f'''(x)$ et $f^{(4)}(x)$.

En particulier :

$$f'(x) = \frac{8f(x+h) + f(x-2h) - 8f(x-h) - f(x+2h)}{12h}$$

et $$f''(x) = \frac{16f(x+h) + 16f(x-h) - f(x+2h) - f(x-2h) - 30f(x)}{12h^2}$$

Exemple : Soit $f(x) = x \sin x$. Pour $h = 0{,}001$ et $x = 1$, les premières formules donnent
$f'(1) = 1{,}381\,772\,92$ et $f''(1) = 0{,}240\,281\,224$.

Les secondes donnent $f'(1) = 1{,}381\,773\,29$ et $f''(1) = 0{,}240\,281\,224$.

Les valeurs exactes sont respectivement $1{,}381\,773\,29$ et $0{,}239\,133\,627$.

4. Précision des calculs

Quel que soit l'ordre choisi, le calcul approché de $f^{(n)}(x)$, par les méthodes exposées dans ce chapitre, reste une opération **relativement imprécise.**

D'une manière générale, on peut constater qu'il existe une valeur optimale de h rendant l'erreur minimale (voir chapitre 1). Il est assez difficile de déterminer cette valeur autrement que de manière empirique. L'estimation de l'erreur commise est également très malaisée et repose sur l'étude du reste d'une **série de Taylor.**

5. Remarque

Les formules de calcul de ce chapitre peuvent servir à déterminer les valeurs approchées des **dérivées partielles** d'une fonction de plusieurs variables.

Par exemple soit $f(x, y) = x^2 \cos y + y \sin x$. En modifiant, le programme du paragraphe 3.2, on peut calculer la **dérivée partielle** première $\dfrac{\partial f}{\partial x}$ de la fonction f :

```
10 DEF FN f(x,y)=x*x*COS(y)+y*SIN(x)
20 INPUT"x=";x
30 INPUT"y=";y
40 h=0.001
50 y1=FN f(x-h,y)
```

```
60 y2=FN f(x,y)
70 y3=FN f(x+h,y)
80 d1=(y3-y1)/2/h
90 PRINT"derivee 1ere=";d1
```

Pour $x = 2$ et $y = 1$, on obtient $\dfrac{\partial f}{\partial x} = 1{,}745\,062\,41$.

La valeur exacte est 1,745 062 39

Chapitre 8
Intégration numérique

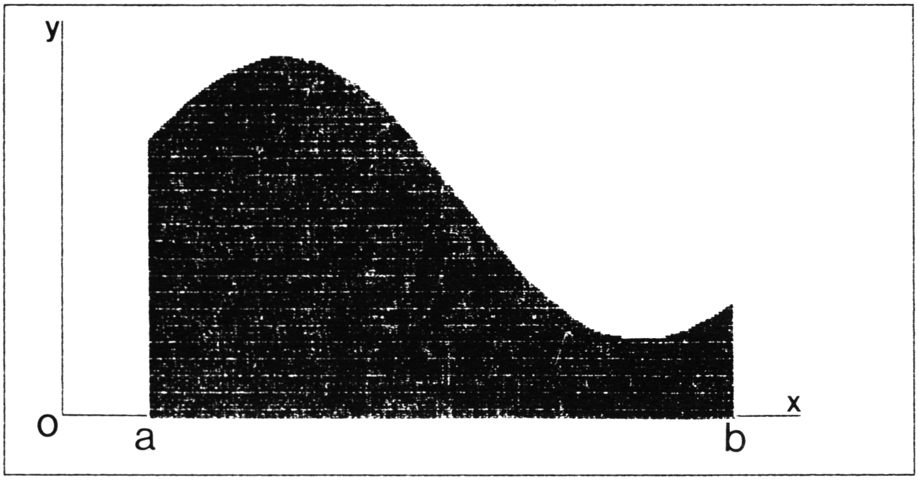
Soit f une fonction numérique définie et continument dérivable sur un intervalle $[a, b]$.

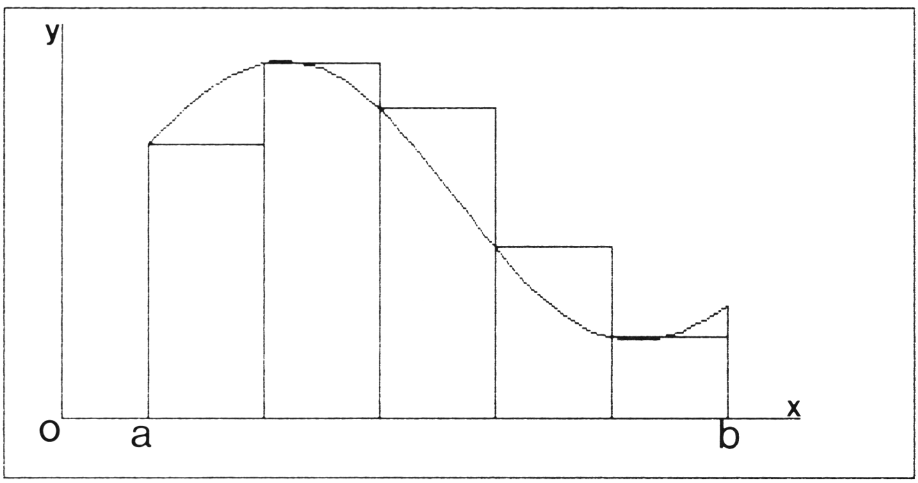
Pour calculer le nombre $I = \int_a^b f(x)\,\mathrm{d}x$, il existe plusieurs méthodes : les unes sont lentes mais permettent une estimation aisée de l'erreur commise, les autres sont rapides mais ne permettent pas aussi facilement d'évaluer l'erreur commise.

1. Méthode des rectangles

Représentons graphiquement la fonction f sur l'intervalle $[a, b]$
L'**aire** limitée par la courbe obtenue, l'axe des x et les droites Δ_1 et Δ_2 a pour mesure $I = \int_a^b f(x)\,\mathrm{d}x$ (voir figure 8.1).

En remplaçant cette surface par n **rectangles** de même largeur, on peut aisément obtenir une valeur approchée S de I.





1.1. Cas général

Chaque rectangle a pour largeur $\frac{b-a}{n}$ et pour hauteur $f(x_i)$ avec, pour $1 \leqslant i \leqslant n$,

$x_i = a + (i-1)\frac{(b-a)}{n}$.

On a donc $S = \frac{b-a}{n} \sum_{i=1}^{i=n} f(x_i)$.

Évaluons l'erreur commise. On a :

$$I = \int_{x_1}^{x_2} f(x)\,\mathrm{d}x + \int_{x_2}^{x_3} f(x)\,\mathrm{d}x + \ldots + \int_{x_n}^{x_{n+1}} f(x)\,\mathrm{d}x = \sum_{1}^{n}\left(\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x\right).$$

D'après le **théorème des accroissements finis,** il existe sur chaque intervalle $[x_i, x_{i+1}]$ un nombre ξ_i tel que $f(x) = f(x_i) + (x - x_i) f'(\xi_i)$ pour $x_i \leqslant x \leqslant x_{i+1}$. Il en résulte :

$$I = \sum_1^n \int_{x_i}^{x_{i+1}} [f(x_i) + (x - x_i) f'(\xi_i)] \, dx$$

soit $I = \sum_1^n (x_{i+1} - x_i) f(x_i) + \sum_1^n \frac{(x_{i+1} - x_i)^2}{2} f'(\xi_i)$.

La première somme n'est autre que S.

Comme $x_{i+1} - x_i = \frac{b-a}{n}$, on peut écrire :

$$I = S + \frac{(b-a)^2}{2n^2} \sum_1^n f'(\xi_i).$$

Les nombres $f'(\xi_i)$ ne sont pas connus.

Supposons les majorés, en valeur absolue, par un nombre M_1. Nous obtenons alors :

$$|I - S| \leqslant M_1 \frac{(b-a)^2}{2n}.$$

Dans le programme qui suit, M_1 est évalué grâce à l'une des formules données au chapitre 7.

```
10 DEF FN f(x)=.......
20 INPUT"a=";a
30 INPUT"b=";b
40 INPUT"n=";n
50 l=(b-a)/n
60 m1=1E-30
70 h=0.001
80 FOR i=1 TO n
90 x=a+(i-1)*l
100 y=FN f(x)
110 d=(FN f(x+h)-FN f(x-h))/2/h
120 IF ABS(d)>m1 THEN m1=ABS(d)
130 s=s+l*y
140 NEXT i
150 ds=(m1*(b-a)↑2)/2/n
160 PRINT"integrale =";s
170 PRINT "erreur<=";ds
```

Exemple : Soit $I = \int_0^2 3^x e^x \, dx$.

Avec $n = 50$, on on obtient $I = 29\,920\,114\,1 \pm 5{,}14$

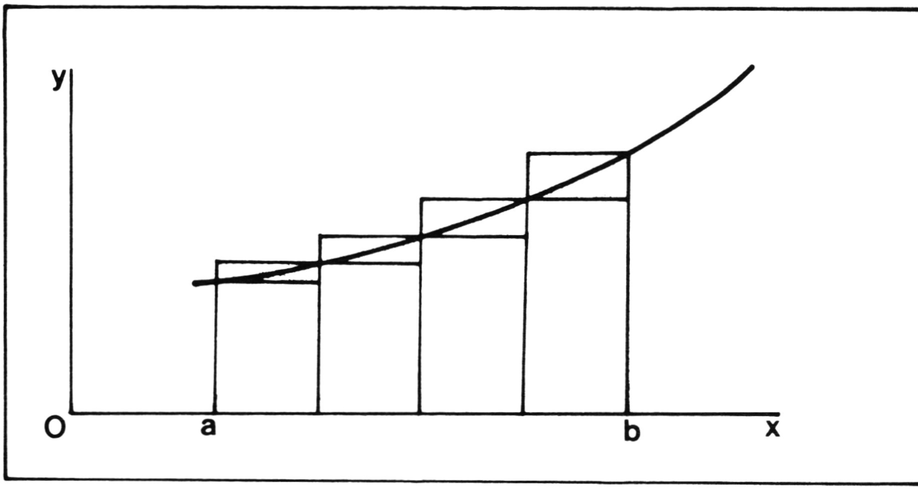
Avec $n = 100$, on obtient $I = 30.561\,384\,9 \pm 2{,}68$

Avec $n = 500$, on obtient $I = 31.080\,998\,3 \pm 0{,}554$.

1.2. Cas d'une fonction monotone

Supposons f monotone sur $[a, b]$: f est alors **croissante** ou **décroissante** sur cet intervalle.

Plaçons nous dans le premier cas (voir figure 8.3). Il devient possible de trouver un encadrement de I : on a :



$$S_1 \leqslant i \leqslant S_2 \quad \text{avec} \quad S_1 = \frac{b-a}{n} \sum_{1}^{n} f(x_i) \quad \text{et} \quad S_2 = \frac{b-a}{n} \sum_{2}^{n+1} f(x_i).$$

Le programme correspondant est le suivant :

```
10 DEF FN f(x)=1/x
20 INPUT"a=";a
30 INPUT"b=";b
40 INPUT"n=";n
50 l=(b-a)/n
60 FOR i=1 TO n
70 x=a+(i-1)*l
80 y=FN f(x)
90 IF i<n+1 THEN s1=s1+l*y
100 IF i>1 THEN s2=s2+l*y
110 NEXT i
120 s=(s1+s2)/2
130 ds=ABS((s1-s2)/2)
140 PRINT"integrale=";s
150 PRINT"erreur<=";ds
```

(Dans le cas d'une fonction décroissante sur $[a, b]$ on échange le rôle de S_1 et de S_2).

Exemple : soit $I = \displaystyle\int_1^2 \frac{dx}{x}$

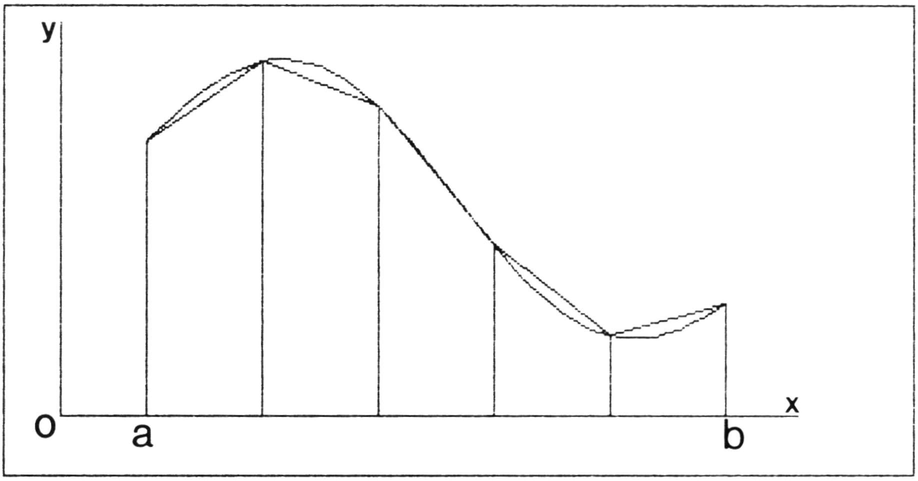
La fonction est décroissante sur [1,2].

Avec $n = 100$, nous obtenons $I = 0{,}690\,653\,431 \pm 0{,}005$.

La valeur exacte est $I = \text{Log } 2 = 0{,}693\,147\,181$.

2. Méthode des trapèzes

Au lieu de prendre des rectangles pour évaluer I, on peut utiliser des **trapèzes** (voir figure 8.4).



2.1. Cas général

Le $i^{ème}$ trapèze a pour aire $(x_{i+1} - x_i)\left(\dfrac{f(x_{i+1}) + f(x_i)}{2}\right)$ d'où, en ajoutant,

$$S = \frac{b-a}{n} \sum_{1}^{n} \frac{f(x_{i+1}) + f(x_i)}{2}.$$

Supposons qu'il existe un nombre positif M_2 tel que $|f''(x)| \leqslant M_2$ pour toute valeur x de $[a, b]$. On démontre alors que le calcul de I par la méthode exposée donne lieu à une erreur inférieure à $\dfrac{M_2(b-a)^3}{12n^2}$.

Dans le programme qui suit la dérivée seconde de f est évaluée grâce à la formule $f''(x) = \dfrac{f(x+2h) + f(x-2h) - 2f(x)}{4h^2}$ (voir chapitre 7).

```
10 DEF FN f(x)=.........
20 INPUT"a=";a
30 INPUT"b=";b
40 INPUT"n=";n
50 l=(b-a)/n
60 m2=1E-30
```

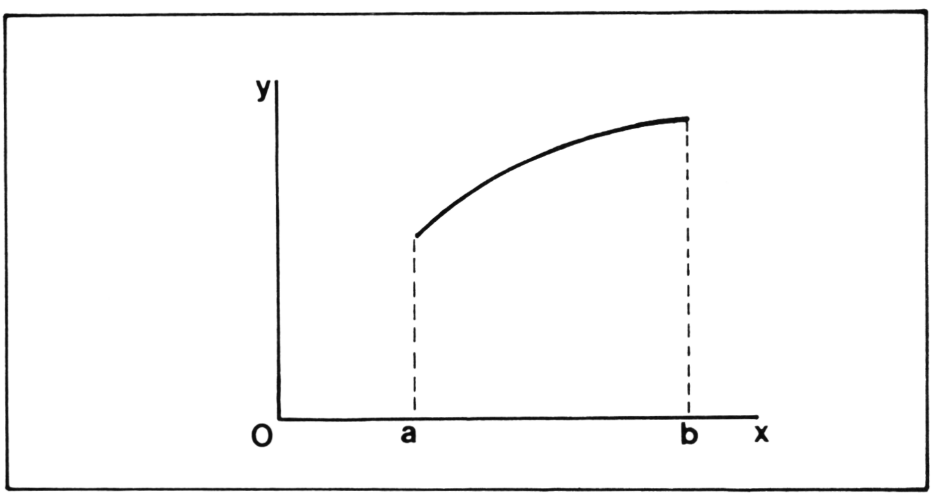
```
70 h=0.001
80 FOR i=1 TO n
90 x1=a+(i-1)*l
100 x2=x1+l
110 y1=FN f(x1)
120 y2=FN f(x2)
130 d=(FN f(x2+2*h)+FN f(x2-2*h)-2*FN f(x2))
140 d=d/4/h/h
150 IF ABS(d)>m2 THEN m2=ABS(d)
160 s=s+l*(y1+y2)/2
170 NEXT i
180 ds=(m2*(b-a)↑3)/12/n/n
190 PRINT"integrale =";s
200 PRINT "erreur<=";ds
```

Exemple : calculons $I = \int_0^2 3^x \, e^x \, dx$.

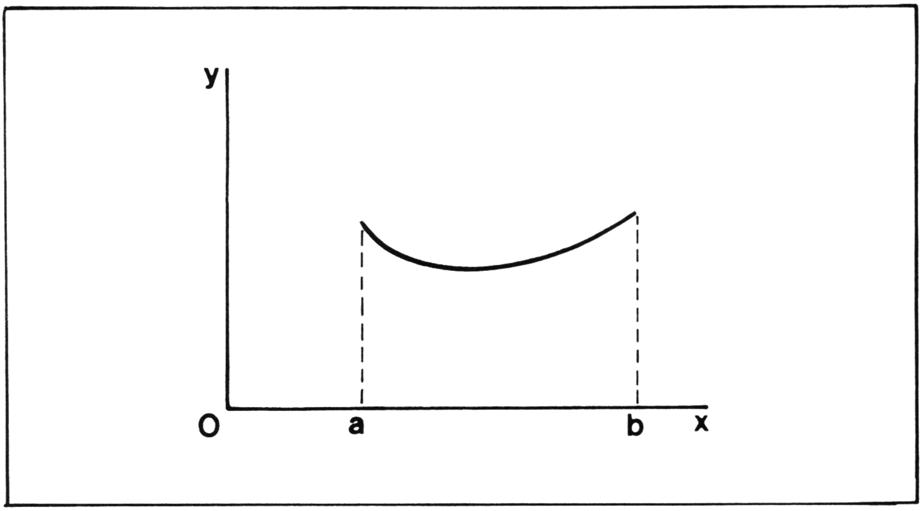
On trouve $I = 31{,}230\,144\,2$ avec $n = 50$. L'erreur commise est inférieure à 0,0781. (La valeur exacte de I est 31,211 818).

2.2. Cas d'une fonction convexe

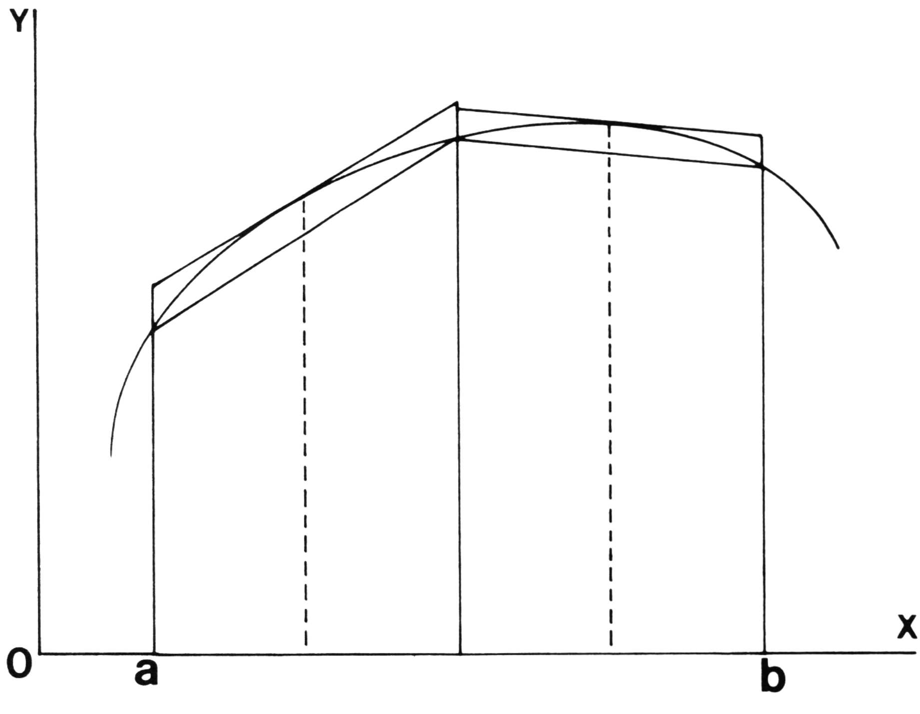
Supposons f **convexe** sur $[a, b]$: le graphe de f se présente comme la figure 8.5 ou bien comme sur la figure 8.6.



En combinant la méthode des trapèzes avec la méthode des tangentes, on peut facilement trouver un encadrement de I.



Supposons, par exemple, f convexe « vers le haut ». La figure 8.7 montre que l'aire du $i^{ème}$ trapèze est supérieure à $\frac{(b-a)}{n}\left[\frac{f(x_i)+f(x_{i+1})}{2}\right]$ et inférieure à $\frac{(b-a)}{n} f\left(\frac{x_i + x_{i+1}}{2}\right)$.



On en déduit, dans le cas présent, $S_1 \leqslant I \leqslant S_2$ avec

$$S_1 = \frac{b-a}{n} \sum_1^n \frac{f(x_i) + f(x_{i+1})}{2} \quad \text{et} \quad S_2 = \frac{b-a}{n} \sum_1^n f\left(\frac{x_i + x_{i+1}}{2}\right).$$

(Si f est convexe « vers le bas », il suffit d'échanger S_1 et S_2).

Le programme est le suivant :

```
10 DEF FN f(x)=x^4
20 INPUT"a=";a
30 INPUT"b=";b
40 INPUT"n=";n
50 l=(b-a)/n
60 FOR i=1 TO n
70 x1=a+(i-1)*l
80 x2=x1+l
90 y1=FN f(x1)
100 y2=FN f(x2)
110 s1=s1+l*(y1+y2)/2
120 s2=s2+l*FN f((x1+x2)/2)
130 NEXT i
140 s=(s1+s2)/2
150 ds=ABS((s1-s2)/2)
160 PRINT"integrale=";s
170 PRINT"erreur<=";ds
```

Exemple : calculons $I = \displaystyle\int_0^2 x^4 \, \mathrm{d}x$

La fonction $x \mapsto x^4$ est convexe sur $[0,2]$. Avec $n = 50$, on obtient $I = 6{,}401\,066\,6 \pm 0{,}0032$. La valeur exacte est $I = 6{,}4$.

3. Méthode de Simpson

On peut obtenir une valeur approchée de $I = \displaystyle\int_a^b f(x) \, \mathrm{d}x$ en remplaçant $f(x)$ par un développement limité à l'ordre 3 sur des sous-intervalles de $[a, b]$.

3.1. Principe

Soit P un polynôme de degré 2 défini par $P(x) = \alpha x^2 + \beta x + \gamma$. On montre facilement que $I = \displaystyle\int_a^b P(x) \, \mathrm{d}x$ ne dépend que de $P(a)$, $P\left(\dfrac{a+b}{2}\right)$ et $P(b)$. Plus exactement, on a $I = \dfrac{b-a}{6}\left[P(a) + 4P\left(\dfrac{a+b}{2}\right) + P(b)\right]$.

En fait cette formule reste valable pour un polynôme de degré 3.

Soit maintenant à calculer $I = \int_a^b f(x)\,dx$. Il existe un unique polynôme P de degré 3 tel que $P(a) = f(a)$, $P\left(\frac{a+b}{2}\right) = f\left(\frac{a+b}{2}\right)$ et $P(b) = f(b)$ (voir chapitre 2).

En remplaçant f par P sur $[a, b]$ on peut alors approcher I par un nombre S.

3.2. Formule de Simpson

Divisons $[a, b]$ en $2n$ parties égales : on obtient $2n + 1$ points d'abscisses $x_i = a + (i - 1)\frac{b - a}{2n}$.

Appliquons la formule précédente aux intervalles $[x_1, x_3]$ $[x_3, x_5]$,..., $[x_{2n-1}, x_{2n+1}]$: on obtient

$$S = \frac{b - a}{6n}\,[f(x_1) + 4f(x_2) + 2f(x_3) + 4f(x_4) + \dots + f(x_{2n+1})].$$

On démontre que l'erreur commise est inférieure à $\frac{M_4(b-a)^5}{2\,880 n^4}$, M_4 désignant un majorant de $|f^{(4)}(x)|$ sur l'intervalle $[a, b]$.

Le programme est le suivant :

```
10 DEF FN f(x)=........
20 INPUT "a=";a
30 INPUT "b=";b
40 IF a>=b THEN PRINT "erreur sur a":GOTO 20
50 INPUT "n=",n
60 l=(b-a)/2/n
70 FOR i=1 TO 2*n-1 STEP 2
80 x1=a+(i-1)*l
90 x2=x1+l
100 x3=x2+l
110 s=s+FN f(x1)+4*FN f(x2)+FN f(x3)
120 NEXT i
130 s=s*l/3
140 PRINT "s=";s
```

(Dans ce programme l'erreur commise n'a pas été évaluée).

Exemple : soit $a = 0$ et $b = 2$; soit $f(x) = 3^x\,e^x$. Calculons I.

On obtient 31,211 839 pour $n = 20$; 31,211 818 5 pour $n = 50$; 31,211 818 1 pour $n = 100$.

La valeur exacte est 31,211 818.

4. Méthode de Gauss

Parmi les diverses méthodes connues sous le nom de méthodes de Gauss, nous avons choisi celle à **six points.**

4.1. Principe

Soit P un polynôme de degré inférieur ou égal à 11. On démontre qu'il existe 6 valeurs $x_1, x_2, ..., x_6$ de la variable et 6 nombres $k_1, k_2, ..., k_6$ tels que :

$$\int_a^b P(x)\,dx = \frac{b-a}{2}\sum_1^6 k_i\, P\left(\frac{a+b}{2} + \frac{b-a}{2}x_i\right).$$

Les valeurs des x_i et des k_i sont les suivants :

$$\begin{aligned}
x_1 &= -\,0{,}932\,469\,514\\
x_2 &= -\,0{,}661\,209\,386\\
x_3 &= -\,0{,}238\,619\,186\\
x_6 &= -\,x_1,\ x_5 = -\,x_2,\ x_4 = -\,x_3.\\
k_1 &= k_6 = 0{,}171\,324\,492\\
k_2 &= k_5 = 0{,}360\,761\,573\\
k_3 &= k_4 = 0{,}467\,913\,935
\end{aligned}$$

Soit maintenant une fonction f définie sur l'intervalle $[a, b]$.

En remplaçant f par un polynôme de degré 11, on obtient une valeur approchée S de $I = \int_a^b f(x)\,dx$ en posant

$$S = \frac{b-a}{2}\sum_1^6 k_i\, f\left(\frac{a+b}{2} + \frac{b-a}{2}x_i\right).$$

L'erreur commise dépend de la dérivée à l'ordre 12 de la fonction f.

Bien qu'il soit très malaisé d'évaluer cette erreur, la méthode de Gauss est très rapide et très précise, pour autant que l'intervalle $[a, b]$ ne soit pas trop grand.

4.2. Programme et exemples

```
10 DEF FN f(x)=.......
20 INPUT "a=",a
30 INPUT "b=",b
40 DIM x(6):DIM k(6)
50 x(1)=-0.932469514:x(6)=-x(1)
60 x(2)=-0.661209386:x(5)=-x(2)
70 x(3)=-0.238619186:x(4)=-x(3)
80 k(1)=0.171324492:k(6)=k(1)
```

```
90 k(2)=0.360761573:k(5)=k(2)
100 k(3)=0.467913935:k(4)=k(3)
110 FOR i=1 TO 6
120 x=(a+b)/2+(b-a)/2*x(i)
130 y=k(i)*FN f(x)
140 s=s+y
150 NEXT i
160 s=(b-a)*s/2
170 PRINT "s=";s
```

1) Soit $I = \displaystyle\int_1^4 \frac{x^2}{\sqrt{1+x^3}}\,\mathrm{d}x$: le programme précédent donne 4,432 028 44 au lieu de 4,432 029 5.

2) Soit $I = \displaystyle\int_0^2 3^x\,\mathrm{e}^x\,\mathrm{d}x$: on obtient 31,211 817 9 valeur exacte à 10^{-6} près.

5. *Cas d'une borne infinie*

On peut essayer d'évaluer des intégrales de la forme $\displaystyle\int_{-\infty}^{a} f(x)\,\mathrm{d}x$ ou $\displaystyle\int_{a}^{+\infty} f(x)\,\mathrm{d}x$ ou bien encore $\displaystyle\int_{-\infty}^{+\infty} f(x)\,\mathrm{d}x$.

On suppose chaque fois f définie et dérivable sur l'intervalle d'intégration.

5.1. Principe

Prenons l'exemple d'une intégrale I de la forme $I = \displaystyle\int_a^{+\infty} f(x)\,\mathrm{d}x$.

On peut écrire, pour $b > a$,

$$I = \int_a^b f(x)\,\mathrm{d}x + \int_b^{+\infty} f(x)\,\mathrm{d}x.$$

Si b est suffisamment grand et si l'intégrale converge, le nombre $\displaystyle\int_b^{+\infty} f(x)\,\mathrm{d}x$ peut être négligeable. Dans ce cas $\displaystyle\int_a^{+\infty} f(x)\,\mathrm{d}x$ constitue une valeur approchée de I.

Pour savoir si b est bien choisi, on peut calculer $\displaystyle\int_b^{2b} f(x)\,\mathrm{d}x$.

5.2. Exemples

1) Soit $I = \int_0^{+\infty} e^{-x^2/2}\,dx$. Avec $n = 50$, la méthode de Simpson donne les résultats suivants :

- pour $b = 5$, $I = 1{,}253\,313\,42$
- pour $b = 10$, $I = 1{,}253\,314\,13$
- pour $b = 20$, $I = 1{,}253\,314\,14$
- pour $b = 50$, $I = 1{,}253\,314\,14$
- pour $b = 100$, $I = 1{,}247\,305\,02$

La valeur exacte est $\sqrt{\frac{\pi}{2}} = 1{,}253\,314\,14$.

Pour $b = 10$, on a $\int_{10}^{20} e^{-x^2/2}\,dx = 1{,}919\,36 \times 10^{-23}$.

2) Soit maintenant $I = \int_0^{+\infty} \frac{dx}{1 + x^2}$.

Avec $n = 200$, la méthode de Simpson donne :

- $1{,}560\,793\,01$ pour $b = 100$
- $1{,}563\,851\,74$ pour $b = 200$
- $1{,}552\,301\,63$ pour $b = 300$

La valeur exacte de I est $\frac{\pi}{2} = 1{,}570\,796\,33$. Comme on le voit, la précision est nettement moins bonne que dans le premier exemple.

Chapitre 9

Résolution d'une équation de la forme $f(x) = 0$.

Soit f une fonction numérique d'une variable réelle, définie sur un intervalle I de $\mathbb{R}$.

Pour résoudre l'équation $f(x) = 0$, on doit d'abord en **localiser** les solutions : il s'agit de trouver des intervalles disjoints qui contiennent chacun une et une seule solution de l'équation.

On peut ensuite aborder le **calcul effectif** de chacune des solutions de l'équation étudiée.

1. Localisation des solutions

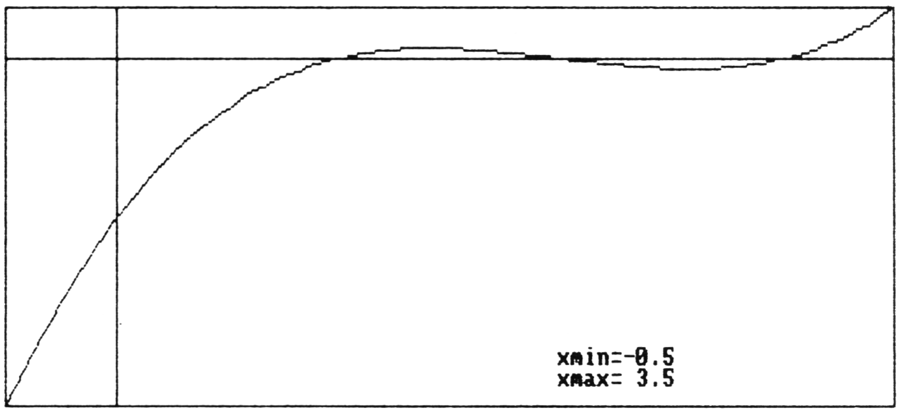
La recherche d'intervalles qui contiennent chacun une solution de l'équation $f(x) = 0$ peut s'effectuer **graphiquement** ou **par un calcul pas à pas.**

1.1. Recherche graphique

Il suffit de représenter la fonction f sur un intervalle quelconque afin de voir s'il y a ou non intersection du graphe de f avec l'axe des x. Pour cela on utilisera le programme du chapitre 5.

Exemple : soit à résoudre $x^3 - 6x^2 + 11x - 6 = 0$.

La figure 9.1 montre qu'il existe 3 solutions de l'équation dans l'intervalle $\left[-\frac{1}{2}, \frac{7}{2}\right]$.



1.2. Recherche pas à pas

L'intervalle $[x_{min}, x_{max}]$ choisi est divisé en n parties égales. Pour $0 \leqslant i \leqslant n$, on a $x_i = x_{min} + \frac{i(x_{max} - x_{min})}{n}$.

Le programme recherche les intervalles $[x_i, x_{i+1}]$ tels que $f(x_i)$ et $f(x_{i+1})$ soient de signes contraires et signale au passage les valeurs x_i telles que $f(x_i) = 0$.

Le programme est le suivant :

```
10 DEF FN f(x)=.........
20 CLS
30 INPUT "xmin=";xmin
40 INPUT "xmax=";xmax
50 INPUT "n=";n
60 DIM x(n+1):DIM s(n+1)
70 FOR i=0 TO n
80 x(i)=xmin+i*(xmax-xmin)/n
90 x=x(i)
100 s(i)=SGN(FN f(x))
110 NEXT i
120 c=0
130 FOR i=1 TO n
140 IF s(i-1)=0 THEN PRINT x(i-1);
150 IF s(i-1)=0 THEN PRINT " est solution"
160 IF s(i-1)=0 THEN c=c+1
170 IF s(i-1)=0 THEN 240
180 IF s(i)=0 THEN 240
190 s=s(i-1)*s(i)
200 IF s=1 THEN 240
210 c=c+1
```

```
220 PRINT "L'intervalle ( ";x(i-1);",";x(i);
230 PRINT ") contient une solution"
240 NEXT i
250 IF s(n)=0 THEN PRINT xmax;" est solution"
260 IF s(n)=0 THEN c=c+1
270 IF c=0 THEN PRINT "Pas de solution ";
280 IF c=0 THEN PRINT "entre ";xmin;" et ";xmax
```

Exemple : soit $f(x) = x^3 - 6x^2 + 11x - 6$. Pour $x_{min} = -5$, $x_{max} = 5$ et $n = 25$, on obtient le résultat suivant : 1 est solution, 3 est solution et il existe une 3e solution dans l'intervalle.

2. *Méthode des approximations successives*

Pour utiliser cette méthode, il faut écrire l'équation $f(x) = 0$ sous la forme $x = \varphi(x)$.

2.1. Principe

Supposons connu un nombre x_1 voisin de la solution cherchée r et posons $x_2 = \varphi(x_1)$, $x_3 = \varphi(x_2)$,..., $x_{n+1} = \varphi(x_n)$.

On définit ainsi une suite qui, si elle converge, a pour limite r.

Choisissons alors un nombre e suffisamment « petit », par exemple 10^{-8}. Le nombre x_n peut être considéré comme une valeur approchée satisfaisante de r si $|x_{n+1} - x_n|$ est inférieur à e.

Dans ce cas on peut vérifier que $f(x_n)$ est très proche de 0 et que les nombres $f(x_n + e)$ et $f(x_n - e)$ ont des signes contraires.

2.2. Programme

Il est possible que le processus de recherche de r n'aboutisse pas. C'est la raison pour laquelle on ne pourra pas effectuer plus de 100 itérations.

En cas d'insuccès, il faut choisir une nouvelle valeur de x_1 ou ... changer de méthode.

```
10 DEF FN phi(x)=........
20 e=0.00000001
30 CLS
40 INPUT "valeur initiale=",x1
50 IF FN phi(x1)=x1 THEN r=x1:GOTO 140
60 x2=FN phi(x1)
70 c=c+1
80 IF ABS(x2-x1)<e THEN 130
90 x1=x2
100 IF c<100 THEN 50
```

```
110 PRINT "Trop d'iterations"
120 END
130 r=x2
140 PRINT "solution=";r
150 PRINT "verification:"
160 PRINT "f(r)=";r-FN phi(r)
170 PRINT "f(r+e)=";r-FN phi(r+e)
180 PRINT "f(r-e)=";r-FN phi(r-e)
```

2.3. Exemples

1) Résolvons l'équation $x - \cos x = 0$.

Écrivons $x = \cos x$ et prenons $x_1 = 0{,}5$.

Nous obtenons $r = 0{,}739\,085\,14$.

2) Soit $x^3 - 3x^2 + x - 1 = 0$.

Écrivons $x = 3x^2 + 1 - x^3$ et prenons $x_1 = 1$. La suite obtenue oscille entre 1 et 3 et n'est pas convergente.

Écrivons alors $x = \dfrac{x - 1 + x^3}{3x}$. $x_1 = 1$ ne permet pas d'aboutir car il y a trop d'itérations. $x_1 = 2$ produit un dépassement de capacité.

Écrivons maintenant $x = \dfrac{3x^2 + 1 - x}{x^2}$.

Avec $x_1 = 2$, nous obtenons la solution $r = 2{,}769\,292\,35$.

3) Soit $x = 2^x - 1$. Cette équation admet les solutions évidentes $r_1 = 0$ et $r_2 = 1$.

En prenant $x_1 = \dfrac{1}{2}$ on obtient r_1 mais, à moins de prendre d'emblée $x_1 = 1$ on ne peut pas obtenir r_2 par la méthode des approximations successives.

3. *Méthode de Newton*

Désignons toujours par $[a, b]$un intervalle qui contient une et une seule solution r de l'équation $f(x) = 0$.

On suppose que f est **dérivable** sur $[a, b]$ et que $f'(x) \neq 0$ pour tout réel x compris entre a et b.

3.1. Principe

Soit x_1 une valeur approchée de r. Posons $r = x_1 + \Delta x$ et cherchons à déterminer Δx. En utilisant un développement limité à l'ordre 1 de $f(x_1 + \Delta x)$ au voisinage de x_1, on peut écrire $f(x_1 + \Delta x) = f(x_1) + \Delta x\, f'(x_1) + \varepsilon$.

Dans cette égalité le nombre ε dépend de Δx mais reste, en général, inconnu.

Puisque $f(x_1 + \Delta x) = 0$, Δx doit vérifier l'équation $f(x_1) + \Delta x f'(x_1) + \varepsilon = 0$.

On en tire puisque, $f'(x_1) \neq 0$, $\Delta x = x_1 - \dfrac{f(x_1) + \varepsilon}{f'(x_1)}$ ou, de manière approchée, $\Delta x = x_1 - \dfrac{f(x_1)}{f'(x_1)}$.

Posons alors $x_2 = x_1 + \Delta x$ et recommençons le même raisonnement en remplaçant x_1 par x_2. De proche en proche on construit ainsi une suite (x_n) définie par la relation $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$ pour $n > 1$.

On démontre que cette suite a pour limite r lorsqu'elle converge.

3.2. Estimation de l'erreur commise

Soit $e = 10^{-8}$ par exemple. Supposons la suite convergente et les calculs arrêtés lorsque $|x_{n+1} - x_n| \leqslant e$.

Posons $r = x_n + \varepsilon_n$. D'après le **théorème des accroissements finis**, il existe un nombre ξ compris entre r et x_n et tel que $f(r) = f(x_n) + \varepsilon_n f'(\xi)$. Comme $f(r) = 0$ et $f'(\xi) \neq 0$, on peut écrire $\varepsilon_n = -\dfrac{f(x_n)}{f'(\xi)}$.

$f'(\xi)$ est inconnu mais proche de $f'(x_n)$. On a donc sensiblement $\varepsilon_n = -\dfrac{f(x_n)}{f'(x_n)}$ soit $\varepsilon_n = x_{n+1} - x_n$. Il en résulte que $|\varepsilon_n|$ diffère peu de e.

3.3. Programme

La suite (x_n) pouvant ne pas être convergente, on a limité à 100 le nombre des itérations possibles.

Si la méthode ne permet pas de trouver r on peut essayer une nouvelle valeur de x_1.

En cas d'échecs répétés, il faut changer de méthode.

```
10 DEF FN f(x)=..........
20 e=0.00000001
30 CLS
40 INPUT "valeur initiale=",x1
50 IF FN f(x1)=0 THEN r=x1:GOTO 180
60 c=c+1
70 y1=FN f(x1+0.001)
80 y2=FN f(x1-0.001)
90 d=(y1-y2)/0.002
100 IF ABS(d)<0.00001 THEN 240
110 x2=x1-FN f(x1)/d
120 IF ABS(x2-x1)<e THEN 170
```

```
130 x1=x2
140 IF c<100 THEN 60
150 PRINT "trop d'iterations"
160 END
170 r=(x2+x1)/2
180 PRINT "solution="; r
190 PRINT "verification:"
200 PRINT "f(r)=";FN f(r)
210 PRINT "f(r+e)=";FN f(r+e)
220 PRINT "f(r-e)=";FN f(r-e)
230 END
240 PRINT " f' s'annule;x1 est mal choisie"
```

Remarque : si on obtient $|f'(x_p)| < 0{,}000\,01$ pour une certaine valeur p de n, on considère que f' s'annule sur l'intervalle choisi et on cesse le calcul.

Notons que f' est calculée, lignes 70, 80 et 90, grâce à une formule du chapitre 7.

3.4. Exemples

1) Considérons l'équation $xe^x - \cos x = 0$. Avec $x_1 = -1$, on obtient $r_1 = -1{,}863\,995\,19$ tandis que $x_1 = 0$ conduit à $r_2 = 0{,}517\,757\,364$.

2) Avec l'équation $(x - 1)^4 = 0$ la méthode ne fonctionne pas. 1 étant une **racine multiple** d'ordre pair, f' s'annule dans tout intervalle contenant 1.

4. Résolution par dichotomie

Plus lente que les autres méthodes, celle-ci présente deux avantages :

- dès qu'on a localisé une solution r de l'équation dans un intervalle $[a, b]$, on est certain de pouvoir calculer r, pour autant que $f(a)$ et $f(b)$ soient de **signes contraires.**
- le calcul de l'erreur commise devient facile.

4.1. Principe

Posons $x = \dfrac{a+b}{2}$. Trois cas peuvent se présenter :

1) si $f(x) = 0$, alors $r = x$.

2) si $f(a)f(x) < 0$, r est compris entre a et x.

3) si $f(a)f(x) > 0$, r est compris entre x et b.

Dans les deux derniers cas, l'intervalle de recherche est réduit de moitié. En répétant le procédé, on peut obtenir r avec la précision voulue.

4.2. Précision de calcul

Soit e l'incertitude sur r et soit Δr l'incertitude effectivement obtenue.

Appelons $[a_n, b_n]$ le dernier intervalle calculé.

Il existe 3 raisons à l'arrêt des calculs :

1) si $|b_n - a_n| \leqslant 2e$ et si $f(a_n) f(b_n) < 0$, on prend :

$$r = \frac{a_n + b_n}{2} \quad \text{et} \quad \Delta r = \frac{b_n - a_n}{2}.$$

On a alors $\Delta r \leqslant e$.

2) si $f\left(\frac{a_n + b_n}{2}\right) = 0$, on a évidemment $r = \frac{a_n + b_n}{2}$. Si $f(a_n) f(b_n) < 0$, on prend $\Delta r = \frac{|b_n - a_n|}{2}$.

Malheureusement il peut arriver que Δr n'ait aucun intérêt pratique car très supérieur à e.

Il faut alors recommencer le calcul en changeant légèrement les valeurs de a et de b.

3) Par suite d'erreurs d'arrondi, il peut arriver que $f(a_n) f(b_n) > 0$. On doit alors arrêter le calcul et prendre $r = \frac{a_{n-1} + b_{n-1}}{2}$ avec $\Delta r = \frac{|b_{n-1} - a_{n-1}|}{2}$.

4.3. Programme

On a choisi $e = 10^{-8}$.

```
10 DEF FN f(x)=3^x-x-5
20 CLS
30 INPUT "valeur de a=",a
40 INPUT "valeur de b=",b
50 e=0.00000001
60 IF FN f(a)=0 THEN r=a:z=1:GOTO 210
70 IF FN f(b)=0 THEN r=b:z=1:GOTO 210
80 s1=SGN( FN f(a)):s2=SGN (FN f(b))
90 IF s1*s2 >0 THEN PRINT"mauvais intervalle"
100 IF s1*s2 >0 THEN GOTO 30
110 x=(a+b)/2
120 IF FN f(x)=0 THEN r=x:z=1:GOTO 210
130 c=a:d=b
140 s1=SGN( FN f(a)):s2=SGN (FN f(x))
150 IF s1*s2<0 THEN b=x ELSE a=x
160 s3=SGN(FN f(a)):s4=SGN(FN f(b))
170 IF s3*s4=1 THEN a=c:b=d:GOTO 190
180 IF ABS(b-a)>2*e THEN 110
190 r=(a+b)/2
200 dr=ABS(b-a)/2
210 PRINT "solution=";r
220 IF z=0 THEN PRINT "incertitude=";dr
230 IF z=1 THEN PRINT "incertitude inconnue"
240 PRINT "verification:"
250 PRINT "f(r)=";FN f(r)
260 IF z=0 THEN PRINT "f(r+dr)=";FN f(r+dr)
270 IF z=0 THEN  PRINT "f(r-dr)=";FN f(r-dr)
```

4.4. Exemples

1) Soit $3^x - x - 5 = 0$. Avec $a = 0$ et $b = 2$, on obtient $r = 1{,}736\,289\,79$, $\Delta r = 7{,}45\ 10^{-9}$.

2) Soit $x^3 - 6x^2 + 11x - 6 = 0$.

Avec $a = 3$ et $b = 5$, on obtient $r = 3$ mais la précision est inconnue.

Prenons alors $a = 2{,}8$ et $b = 5$. Cette fois $r = 3$ et $\Delta r = 8\ .\ 10^{-9}$.

5. *Méthode graphique*

Écrivons l'équation $f(x) = 0$ sous la forme $h(x) = g(x)$ et représentons graphiquement les deux fonctions h et g en prenant $a \leqslant x \leqslant b$.

Si les deux courbes obtenues ont un point commun, l'abscisse de celui-ci est solution de l'équation $h(x) = g(x)$ donc de l'équation $f(x) = 0$.

5.1. Détermination de la solution

Pour représenter graphiquement les deux fonctions h et g, on utilise le même principe que celui qui a été décrit au chapitre 5 : à tout point (x, y) du plan, on associe le point de l'écran dont les coordonnées sont u et v.

Puisque $u = k_1x + \ell_1$ et $v = k_2y + \ell_2$, on obtient $x = \dfrac{u - \ell_1}{k_1}$.

5.2. Obtention des coordonnées-écran

On utilise pour cela le principe des ardoises magiques. On amène un curseur sur le point de l'écran voulu grâce aux touches de déplacement $\rightarrow$, $\uparrow$, $\downarrow$ et $\leftarrow$.

Il est alors facile d'obtenir la coordonnée u. Notons l'emploi des instructions TEST (x, y), PLOT x, y, 1 et PLOT x, y, 0.

La première instruction permet de savoir si le pixel (x, y) est allumé ou éteint, les suivantes permettent de marquer ou d'effacer un pixel de l'écran.

5.3. Programme

```
10 DEF FN g(x)=.........
20 DEF FN h(x)=.........
30 REM Choix de l'intervalle
40 INPUT"xmin=";xmin
50 INPUT"xmax=";xmax
```

```
60 IF xmin>xmax THEN 40
70 CLS
80 REM Definition et dessin de la fenetre
90 umin=50:vmin=50
100 umax=500:vmax=350
110 MOVE umin,vmin
120 DRAW umax,vmin,1
130 DRAW umax,vmax,1
140 DRAW umin,vmax,1
150 DRAW umin,vmin,1
160 REM Partage de (xmin,xmax) en n parties
170 REM egales.Recherche des maxima et des
180 REM minimas des fonctions g et h
190 n=50
200 DIM x(n+1):DIM g(n+1):DIM h(n+1)
210 gmax=-1E+30:gmin=1E+30
220 FOR i=0 TO n
230 x(i)=xmin+i*(xmax-xmin)/n
240 g(i)=FN g(x(i))
250 h(i)=FN h(x(i))
260 IF g(i)>gmax THEN gmax=g(i)
270 IF h(i)>gmax THEN gmax=h(i)
280 IF g(i)<gmin THEN gmin=g(i)
290 IF h(i)<gmin THEN gmin=h(i)
300 NEXT i
310 REM Adaptation a la taille de l'ecran
320 k1=(umax-umin)/(xmax-xmin)
330 l1=umax-k1*xmax
340 k2=(vmax-vmin)/(gmax-gmin)
350 l2=vmax-k2*gmax
360 FOR i=0 TO n
370 x(i)=k1*x(i)+l1
380 g(i)=k2*g(i)+l2
390 h(i)=k2*h(i)+l2
400 NEXT i
410 REM Dessin des 2 courbes
420 FOR i=0 TO n-1
430 MOVE x(i),g(i):DRAW x(i+1),g(i+1),1
440 MOVE x(i),h(i):DRAW x(i+1),h(i+1),1
450 NEXT i
460 LOCATE 10,24
470 PRINT "xmin=";xmin,"xmax=";xmax
480 REM Recherche de l'abscisse du point
490 REM d'intersection des 2 courbes
500 x=300:y=200
510 a$=INKEY$
520 IF a$="r" THEN RUN
530 IF a$=CHR$(243) THEN x=x+1
540 IF a$=CHR$(242) THEN x=x-1
550 IF a$=CHR$(240) THEN y=y+1
560 IF a$=CHR$(241) THEN y=y-1
570 IF a$="s" THEN 640
580 t=TEST(x,y)
590 IF t=1 THEN PLOT x,y,0
600 PLOT x,y,1
610 IF t=0 THEN PLOT x,y,0
620 GOTO 510
```

```
630 REM Affichage de la solution
640 LOCATE 10,24
650 sol=(x-l1)/k1
660 PRINT "solution de l'equation:";sol
670 PRINT "verification:"
680 PRINT "g(";sol;")=";FN g(sol)
690 PRINT "h(";sol;")=";FN h(sol)
```

Le programme étant lancé par RUN, il faut choisir l'intervalle $[x_{min}, x_{max}]$ sur lequel on veut représenter g et h.

Deux cas peuvent se présenter :

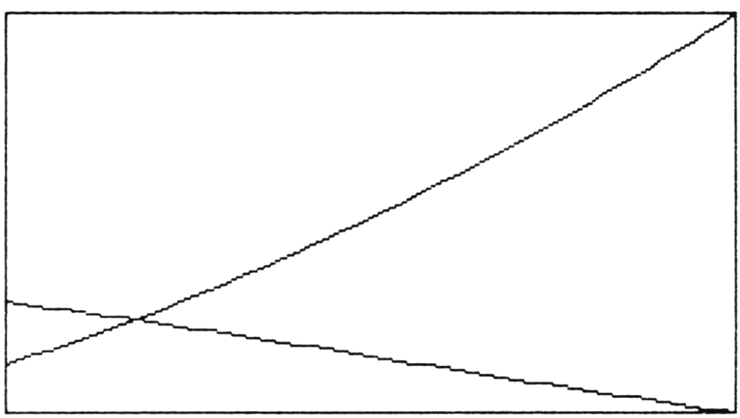
1) les deux courbes ne se coupent pas : en appuyant sur *R*, on pourra choisir un nouvel intervalle.

2) si les deux courbes se coupent, on pourra amener le curseur clignotant sur le point d'intersection grâce aux touches →, ←, ↑ et ↓. (attention, ce curseur est un simple point!).

En appuyant sur *S*, on aura la valeur de x et, à titre de vérification, celles de $g(x)$ et de $h(x)$.

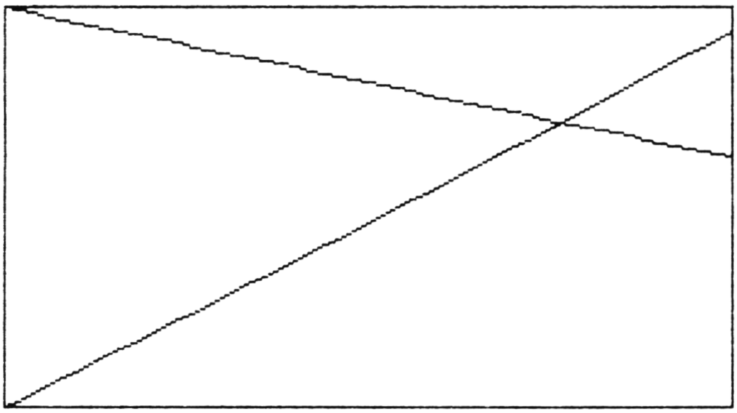
Exemple : soit à résoudre $x^2 - 1 = \cos x$.

Prenons $x_{min} = 1$ et $x_{max} = 2$.



```
         solution de l'equation: 1.18
verification:
g( 1.18 )= 0.3924          h( 1.18 )= 0.380924824
```

Prenons alors $x_{\min} = 1{,}1$ et $x_{\max} = 1{,}2$:



```
        solution de l'equation: 1.17644445
verification:
g( 1.17644445 )= 0.384021532            h( 1.17644445 )= 0.384209898
```

En choisissant un intervalle encore plus petit on peut espérer améliorer la précision tant que les deux courbes restent bien distinctes.

Chapitre 10
Résolution d'un système d'équations linéaires

Considérons le système de n équations à n inconnues suivant :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n. \end{cases}$$

Un tel système est linéaire et peut être résolu de manière approchée par de nombreuses méthodes.

Nous avons choisi d'utiliser la **méthode d'élimination de Gauss,** appelée encore **méthode du pivot**.

Cette méthode convient pour $n \leqslant 100$ en simple précision et $n \leqslant 200$ en double précision.

1. La méthode du pivot

En combinant convenablement les lignes du système initial, on peut le transformer en un **système triangulaire** caractérisé par la présence de zéros sous la diagonale principale.

1.1. Exemple

Soit à résoudre $\begin{cases} x_1 + 2x_2 + x_3 = 2 \\ -x_1 - 2x_2 + x_4 = -7 \\ x_1 - x_2 + x_3 - x_4 = 0 \\ -x_1 + x_2 + x_3 + x_4 = -2 \end{cases}$

Associons à ce système la matrice à 4 lignes et 5 colonnes suivantes :

$$\begin{bmatrix} 1 & 2 & 1 & 0 & 2 \\ -1 & -2 & 0 & 1 & -7 \\ 1 & -1 & 1 & -1 & 0 \\ -1 & 1 & 1 & 1 & -2 \end{bmatrix}$$

Les seules manipulations de lignes autorisées seront au nombre de trois :

- on peut échanger deux lignes,
- on peut remplacer une ligne par la somme ou la différence de cette ligne avec une autre,
- on peut multiplier tous les éléments d'une ligne par un réel non nul.

Naturellement, on peut combiner les trois types de manipulation.

Si L_1, L_2, L_3,... désignent symboliquement les lignes du système à résoudre, nous désignerons le résultat d'une manipulation par des notations telles que $L_1 + L_2$, $2L_2$, $L_3 - 2L_4$,... etc.

1) Remplaçons L_2 par $L_1 + L_2$, L_3 par $L_3 - L_1$ et L_4 par $L_4 + L_1$. On obtient :

$$\begin{bmatrix} 1 & 2 & 1 & 0 & 2 \\ 0 & 0 & 1 & 1 & -5 \\ 0 & -3 & 0 & -1 & -2 \\ 0 & 3 & 2 & 1 & 0 \end{bmatrix}$$

2) puisque $a_{22} = 0$, échangeons L_2 et L_3

$$\begin{bmatrix} 1 & 2 & 1 & 0 & -2 \\ 0 & -3 & 0 & -1 & -2 \\ 0 & 0 & 1 & 1 & -5 \\ 0 & 3 & 2 & 1 & 0 \end{bmatrix}$$

3) Remplaçons L_4 par $L_4 + L_2$:

$$\begin{bmatrix} 1 & 2 & 1 & 0 & 2 \\ 0 & -3 & 0 & -1 & -2 \\ 0 & 0 & 1 & 1 & -5 \\ 0 & 0 & 2 & 0 & -2 \end{bmatrix}$$

4) Remplaçons L_4 par $L_4 - 2L_3$:

$$\begin{bmatrix} 1 & 2 & 1 & 0 & 2 \\ 0 & -3 & 0 & -1 & -2 \\ 0 & 0 & 1 & 1 & -5 \\ 0 & 0 & 0 & -2 & 8 \end{bmatrix}$$

5) Les manipulations sont terminées. Le système initial est devenu :

$$\begin{cases} x_1 + 2x_2 + x_3 = 2 \\ - 3x_2 - x_4 = -2 \\ x_3 + x_4 = -5 \\ - 2x_4 = 8 \end{cases}$$

De proche en proche, on obtient $x_4 = -4$, $x_3 = -1$, $x_2 = 2$ et $x_1 = -1$.

1.2. Discussion

Si le système triangulaire obtenu contient un zéro sur la diagonale principale, le système initial est **indéterminé** ou **sans solution.**

Exemples

$$\begin{cases} 2x_1 - x_2 + x_3 = 1 \\ x_2 + x_3 = 2 \\ 0x_3 = 0 \end{cases} \quad \text{est indéterminé}$$

$$\begin{cases} x_1 + x_2 - x_3 = 5 \\ x_2 + x_3 = -2 \\ 0x_3 = 1 \end{cases} \quad \text{est sans solution}$$

2. Programme

Lorsque la valeur absolue d'un coefficient diagonal est inférieure à 0,001, on essaye de procéder à un échange de lignes.

Si cette tentative reste sans succès, on ne traite pas le système qui est alors qualifié de **singulier.**

```
10 REM Donnees
20 INPUT"Combien d'inconnues ";n
30 DIM a(n,n+1):DIM x(n)
40 DIM t(n,n+1):DIM y(n)
50 FOR l=1 TO n
60 FOR c=1 TO n+1
70 IF c<n+1 THEN PRINT "a(";l;",";c;")=";
```

```
80 IF c=n+1 THEN PRINT "b(";l;")=";
90 INPUT a(l,c)
100 t(l,c)=a(l,c)
110 NEXT c
120 NEXT l
130 REM Echange eventuel des lignes
140 FOR c=1 TO n-1
150 IF ABS(a(c,c))>0.001 THEN 260
160 FOR i=c+1 TO n
170 IF ABS(a(i,c))>0.001 THEN 200
180 NEXT i
190 PRINT"cas singulier":END
200 FOR j=c TO n+1
210 z=a(i,j)
220 a(i,j)=a(c,j)
230 a(c,j)=z
240 NEXT j
250 REM Calcul des nouveaux coefficients
260 FOR l=c+1 TO n
270 IF a(l,c)=0 THEN 320
280 k=-a(l,c)/a(c,c)
290 FOR j=1 TO n+1
300 a(l,j)=a(l,j)+k*a(c,j)
310 NEXT j
320 NEXT l
330 NEXT c
340 IF ABS(a(n,n))<0.001 THEN 190
350 REM Clacul des x(i)
360 FOR l=n TO 1 STEP -1
370 s=0
380 FOR c=l+1 TO n
390 s=s+a(l,c)*x(c)
400 NEXT c
410 x(l)=(a(l,n+1)-s)/a(l,l)
420 NEXT l
430 REM Affichage de la solution et verification
440 FOR l=1 TO n
450 y(l)=0
460 FOR c=1 TO n
470 y(l)=y(l)+t(l,c)*x(c)
480 NEXT c
490 PRINT "x(";l;")=";x(l),"b'(";l;")=";y(l)
500 NEXT l
```

En plus des x_i, le programme calcule, à titre de **vérification,** les nombres b'_i obtenus en portant les valeurs des x_i dans chacune des équations du système.

3. Précision des calculs

L'évaluation de l'erreur de calcul commise dépasse très nettement le niveau de cet ouvrage.

Aussi nous contenterons-nous d'esquisser ce calcul.

3.1. Évaluation de l'erreur

En notation matricielle, un système linéaire peut s'écrire sous la forme $AX = B$.

Si X et $\overline{X}$ désignent respectivement la solution exacte et la solution approchée du système, on a $A(X - \overline{X}) = B - \overline{B}$, avec $\overline{B} = A\overline{X}$.

On en tire $X - \overline{X} = \overline{A}^{-1}(B - \overline{B})$, $\overline{A}^{-1}$ étant la matrice inverse de A.

On démontre que la précision du calcul est « bonne » lorsque les coefficients de $\overline{A'}$ ne sont pas trop grands par rapport à ceux de A.

Si ce n'est pas le cas, le système est dit **mal conditionné.**

3.2. Exemple de systèmes mal conditionnés

Considérons les systèmes $\begin{cases} 5x - y = 9 \\ 4x + 3y = 11 \end{cases}$ et $\begin{cases} 5{,}01x - 0{,}98y = 9 \\ 3{,}99x + 3{,}01y = 11. \end{cases}$

Ces systèmes ont des coefficients très voisins. Leurs solutions sont également très voisines puisqu'on trouve $x = 2$, $y = 1$ pour le premier et $x = 1{,}994$, $y = 1{,}011$ pour le second.

Au contraire, les systèmes $\begin{cases} 3{,}48x + 5{,}27y = 2{,}54 \\ 6{,}25x + 9{,}45y = 5{,}03 \end{cases}$ et $\begin{cases} 3{,}49x + 5{,}25y = 2{,}54 \\ 6{,}24x + 9{,}44y = 5{,}03 \end{cases}$, bien que très voisins, ont des solutions très dissemblables : $x = 48{,}642$, $y = -31{,}638$ dans le premier cas contre $x = -13{,}092$, $y = 9{,}187$ dans le second.

Ces deux derniers systèmes sont mal conditionnés car leurs déterminants sont proches de 0 :

$$3{,}48 \times 9{,}45 - 5{,}27 \times 6{,}25 = -0{,}005\,15$$
$$3{,}49 \times 9{,}44 - 5{,}25 \times 6{,}24 = 0{,}185\,59.$$

De faibles variations des coefficients induisent donc dans ce cas des variations très grandes au niveau de la solution calculée : celle-ci devient sans valeur.

4. *Deux exemples*

On a très souvent besoin, en calcul numérique, de résoudre des systèmes d'équations linéaires. Traitons deux exemples.

4.1. Parabole passant par 3 points

Soient $A = (1,0)$, $B = (2,2)$ et $C = (3,5)$ trois points du plan distincts deux à deux.

Cherchons l'équation $y = ax^2 + bx + c$ d'une parabole passant par ces 3 points.

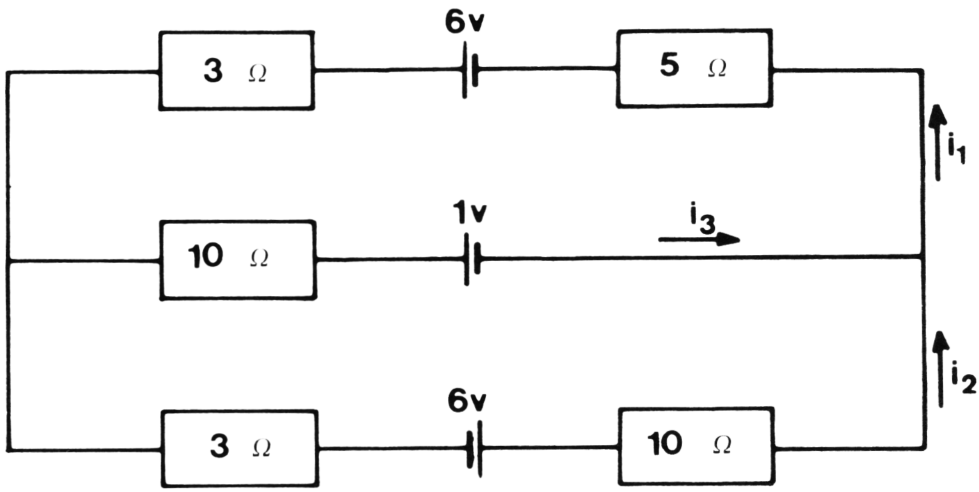
Nous devons résoudre le système

$$\begin{cases} a + b + c = 0 \\ 4a + 2b + c = 2 \\ 9a + 3b + c = 5 \end{cases}$$

On obtient $a = 0{,}5$, $b = 0{,}5$ et $c = -1$.

4.2. Circuit électrique

Calculons i_1, i_2 et i_3 dans le circuit de la figure ci-dessous.



D'après les lois de Kirchoff, on peut écrire :

$$\begin{cases} i_1 - i_2 - i_3 = 0 \\ 8i_1 + 10i_3 = 5 \\ 13i_2 - 10i_3 = 7 \end{cases}$$

On obtient $i_1 = 0{,}589\ A$, $i_2 = 0{,}561\ A$ et $i_3 = 0{,}029\ A$.

Chapitre 11
Résolution d'un système d'équations non linéaires

Soient $f_1, f_2, ..., f_n$ n fonctions numériques des variables réelles $x_1, x_2, ..., x_n$.
Le système des n équations à n inconnues :

$$f_1(x_1, x_2, ..., x_n) = f_2(x_1, x_2, ..., x_n) = ... = f_n(x_1, x_2, ..., x_n) = 0$$

peut fort bien ne pas être **linéaire.**

Deux méthodes de résolution de tels systèmes seront exposées dans ce chapitre.

Nous nous limiterons toutefois à des systèmes de 2 équations à 2 inconnues du type $f(x, y) = g(x, y) = 0$.

Les fonctions f et g seront supposées posséder des dérivées partielles de tous ordres.

1. *Méthode des approximations successives*

Cette méthode est tout à fait semblable à celle qui a déjà été étudiée pour résoudre les équations du type $f(x) = 0$ (voir chapitre 9, paragraphe 2).

1.1. Principe

Écrivons le système initial sous la forme

$$\begin{cases} x = \varphi(x, y) \\ y = \psi(x, y) \end{cases}$$

Après avoir choisi deux nombres x_0 et y_0, on calcule successivement les nombres $x_1 = \varphi(x_0, y_0)$ et $y_1 = \psi(x_0, y_0)$, puis les nombres $x_2 = \varphi(x_1, y_1)$ et $y_2 = \psi(y_1, y_1)$ et ainsi de suite.

On construit donc, de proche en proche, deux suites numériques (x_n) et (y_n).

Si ces suites sont convergentes, elles ont pour limites respectives les solutions x et y du système.

1.2. Le programme

Selon le choix de x_0 et de y_0, le calcul peut aboutir ou non, aussi nous avons limité le nombre des itérations possibles à 100.

```
10 DEF FN phi(x,y)=...........
20 DEF FN psi(x,y)=...........
30 e=0.001
40 INPUT "valeur initiale de x";x1
50 INPUT "valeur initiale de y";y1
60 FOR i=1 TO 100
70 x2=FN phi(x1,y1)
80 y2=FN psi(x1,y1)
90 a=ABS(x2-x1):b=ABS(y2-y1)
100 IF a<=e  AND b<=e THEN z=1:GOTO 130
110 x1=x2:y1=y2
120 PRINT x2,y2
130 NEXT i
140 IF z=0 THEN PRINT "trop d'iterations"
150 IF z=1 THEN GOSUB 170
160 END
170 PRINT "solution:"
180 PRINT "x=";x2,"y=";y2
```

1.3. Exemples

1) Le système $\begin{cases} x - \sqrt{x + y} - 1 = 0 \\ y - \sqrt{x - y} = 0 \end{cases}$ peut s'écrire $\begin{cases} x = \sqrt{x + y} + 1 \\ y = \sqrt{x - y} \end{cases}$. Avec $x_0 = 0$, $y_0 = 0$ nous obtenons $x = 3{,}106\,797\,28$ et $y = 1{,}332\,410\,42$.

2) Soit maintenant $x(x + 1) = \sin y$ et $y - \cos(x - y) = 0$. Ce système peut s'écrire $\begin{cases} x = \dfrac{\sin y}{x + 1} \\ y = \cos(x - y) \end{cases}$

Avec $x_0 = y_0 = 1$, on obtient $x = 0{,}522\,652\,871$ et $y = 0{,}921\,997\,266$.

2. Méthode de Newton-Raphson

Il s'agit là encore d'une **méthode itérative.** On commence par choisir un nombre $\varepsilon > 0$ et deux nombres x_0 et y_0 suffisamment proches des solutions recherchées. On construit ensuite deux suites (x_n) et (y_n) par récurrence. Le calcul est arrêté dès que les relations $|x_{n+1} - x_n| \leqslant \varepsilon$ et $|y_{n+1} - y_n| \leqslant \varepsilon$ sont vérifiées.

Le passage des nombres x_n et y_n aux nombres x_{n+1} et y_{n+1} s'effectue grâce à l'utilisation de deux **développements limités.**

2.1. Formules de passage

Posons $x_{n+1} = x_n + \Delta x$ et $y_{n+1} = y_n + \Delta y$. Les nombres Δx et Δy doivent vérifier le système $\begin{cases} f(x_n + \Delta x, y_n + \Delta y) = 0 \\ g(x_n + \Delta x, y_n + \Delta y) = 0. \end{cases}$

Ce système ne peut être résolu directement et doit être remplacé par un système « voisin ». Pour cela, développons $f(x_n + \Delta x, y_n + \Delta y)$ et $g(x_n + \Delta x, y_n + \Delta y)$ au voisinage de (x, y) à l'ordre 1. Nous obtenons :

$$f(x_n + \Delta x, y_n + \Delta y) = f(x_n, y_n) + \Delta x\, f'_x(x_n, y_n) + \Delta y\, f'_y(x_n, y_n)$$
$$g(x_n + \Delta x, y_n + \Delta y) = g(x_n, y_n) + \Delta x\, g'_x(x_n, y_n) + \Delta y\, g'_y(x_n, y_n).$$

(Dans ces expressions, f'_x, f'_y, g'_x et g'_y désignent respectivement les dérivées partielles des fonctions f et g suivant x et suivant y).

Il en résulte le système suivant, linéaire en Δx et Δy :

$$\begin{cases} \Delta x\, f'_x(x_n, y_n) + \Delta y\, f'_y(x_n, y_n) = -f(x_n, y_n) \\ \Delta x\, g'_x(x_n, y_n) + \Delta y\, g'_y(x_n, y_n) = -g(x_n, y_n) \end{cases}$$

Soient $a = f'_x(x_n, y_n)$, $b = f'_y(x_n, y_n)$, $c = g'_x(x_n, y_n)$, $d = g'_y(x_n, y_n)$, $f = f(x_n, y_n)$ et $g = g(x_n, y_n)$.

Les solutions du système précédent existent pour $ad - bc \neq 0$: on a alors :

$$\Delta x = \frac{bg - fd}{ad - bc} \quad \text{et} \quad \Delta y = \frac{fc - ag}{ad - bc}.$$

2.2. Le programme

Les nombres a, b, c et d qui représentent des dérivées partielles ont été calculés grâce aux formules du chapitre 7, paragraphe 5, avec $h = 0{,}001$. La condition $ad - bc \neq 0$ a été remplacée par $|ad - bc| < 0{,}000\,001$: si on se trouve dans ce cas, celui-ci est signalé par l'expression « cas singulier ».

Comme pour le programme précédent, un maximum de 100 itérations est possible.

```
10 DEF FN f(x,y)=.......
20 DEF FN g(x,y)=.......
30 e=0.0001
40 h=0.001
50 INPUT "x1=";x
60 INPUT "y1=";y
70 FOR i= 1 TO 100
80 a=(FN f(x+h,y)-FN f(x-h,y))/2/h
90 b=(FN f(x,y+h)-FN f(x,y-h))/2/h
100 c=(FN g(x+h,y)-FN g(x-h,y))/2/h
110 d=(FN g(x,y+h)-FN g(x,y-h))/2/h
120 f=FN f(x,y):g=FN g(x,y)
130 de=a*d-b*c
140 IF ABS(de)<0.000001 THEN z=1:GOTO 200
150 dx=(b*g-f*d)/de
160 dy=(c*f-a*g)/de
170 IF ABS(dx)<e AND ABS(dy)<e THEN z=2
180 IF ABS(dx)<e AND ABS(dy)<e THEN GOTO 200
190 x=x+dx:y=y+dy
200 NEXT i
210 IF z=0 THEN PRINT "trop d'iterations"
220 IF z=1 THEN PRINT "cas particulier:";
230 IF z=1 THEN PRINT "pas de solution"
240 IF z=2 THEN GOSUB 260
250 END
260 PRINT "solution: "
270 PRINT "x=";x,"y=";y
280 PRINT " verification :"
290 PRINT "f(x,y)=";FN f(x,y)
300 PRINT "g(x,y)=";FN g(x,y)
310 RETURN
```

2.3. Exemples

1) $f(x, y) = x^3 - y^2 + y + 5$, $g(x, y) = x^2 + y^2 + 2x - 3$.

- pour $x_0 = y_0 - 1$, on obtient $x = -0{,}109\,871\,974$ et $y = -1{,}790\,999\,28$
- pour $x_0 = y_0 = 0$, on obtient $x = -0{,}109\,871\,109$ et $y = -1{,}790\,998\,86$
- pour $x_0 = y_0 = 1$, on obtient $x = -1{,}468\,041\,68$ et $y = 1{,}944\,485\,82$.

2) Soient $f(x, y) = x - \sin xy$ et $g(x, y) = x^2 - \cos y$.

- pour $x_0 = -1$ et $y_0 = 0$, on obtient $x = -0{,}675\,019\,377$ et $y = 1{,}097\,713\,32$
- $x_0 = 0$ et $y_0 = 0$ constituent un cas singulier
- pour $x_0 = y_0 = 1$, on obtient $x = 0{,}675\,019\,377$ et $y = 1{,}097\,713\,32$.

3. Applications

Beaucoup de problèmes peuvent se ramener à la résolution d'un système non linéaire de deux équations à deux inconnues. Nous allons en donner deux exemples.

3.1. Équation $P(z) = 0$

Considérons l'équation $z^3 + z^2 - z + 15 = 0$ dans laquelle la variable z est une **variable complexe.**

Posons $z = x + iy$, avec $x \in \mathbb{R}$ et $y \in \mathbb{R}$.

L'équation précédente devient

$(x^3 + x^2 - x - 3xy^2 - y^2 + 15) + i(3x^2y + 2xy - y - y^3) = 0$ ce qui conduit au système :

$$\begin{cases} x^3 + x^2 - x - 3xy^2 - y^2 + 15 = 0 \\ 3x^2y + 2xy - y - y^3 = 0 \end{cases}$$

La méthode de Newton-Raphson nous permet d'obtenir les 3 solutions de l'équation précédente :

- avec $x_0 = -1$ et $y_0 = -2$, on obtient $z_1 = 1{,}000\,000\,07 - 2{,}000\,000\,06i$
- avec $x_0 = y_0 = 0$, on obtient $z_2 = -3{,}000\,017\,35$
- avec $x_0 = y_0 = 1$, on obtient $z_3 = 1{,}000\,002\,41 + 1{,}999\,924\,52i$.

(Les solutions exactes de l'équation sont :

$$z_1 = 1 - 2i,\ z_2 = -3 \quad \text{et} \quad z_3 = 1 + 2i).$$

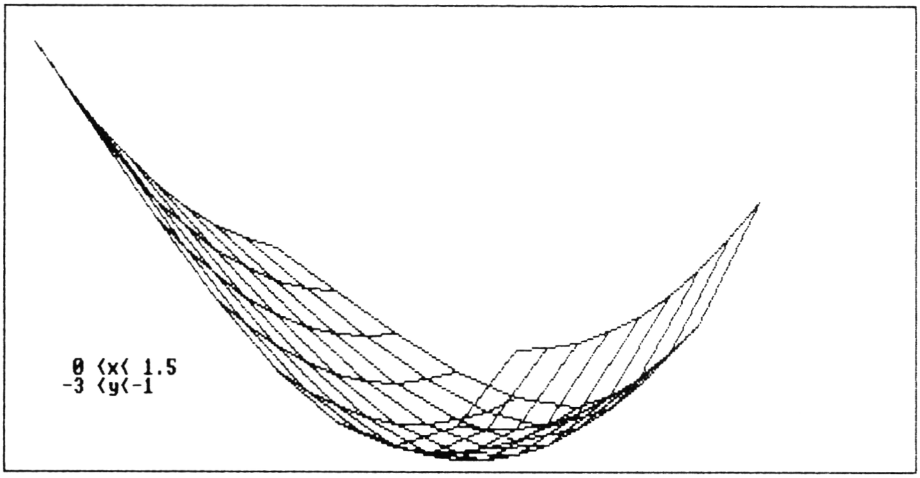
3.2. Recherche d'un extrêmum

Soit par exemple $f(x, y) = 2x^4 + 3xy + y^2 - 3x + y$ une fonction numérique de deux variables réelles. Cette fonction présente un **minimum** ou un **maximum** en (x, y) si et seulement si (x, y) est solution du système $\dfrac{\partial f}{\partial x}(x, y) = \dfrac{\partial f}{\partial y}(x, y) = 0$.

Dans le cas présent, $\dfrac{\partial f}{\partial x}(x, y) = 8x^3 + 3y - 3$ et $\dfrac{\partial f}{\partial y} = 3x + 2y + 1$.

Avec $x_0 = 0$ et $y_0 = 0$ comme valeurs initiales, la méthode de Newton-Raphson nous donne $x = 1{,}048\,369\,97$ et $y = -2{,}072\,554\,96$.

Pour ces valeurs $f(x, y) = -5{,}024\,642\,04$. Il s'agit d'un minimum comme le montre la représentation graphique de la fonction faite, pour $0 \leqslant x \leqslant 1{,}5$ et $-3 \leqslant y \leqslant -1$, avec le programme du chapitre 5, paragraphe 6.



Chapitre 12
Équations différentielles

Soit y une fonction numérique définie sur un intervalle I de $\mathbb{R}$ par une **équation différentielle** et des **conditions initiales** $x = x_0$, $y_0 = y(x_0)$, $y'_0 = y'(x_0)$,... etc. On ne sait pas toujours calculer explicitement y.

Nous nous proposons donc de calculer $y(x)$ de manière approchée pour toute valeur x de I. Nous ne traiterons que les équations différentielles d'ordre 1 ou 2.

1. Équations du premier ordre

Celles-ci doivent se présenter sous la forme $y' = f(x, y)$ avec la condition initiale $y_0 = y(x_0)$.

1.1. Méthode d'Euler

Divisons l'intervalle $[x_0, x]$ en n parties égales et posons $h = \dfrac{x - x_0}{n}$.

Pour $0 \leqslant i \leqslant n$ on a $x_i = x_0 + ih$.

En utilisant à chaque fois un développement limité à l'ordre 1, on peut assimiler $y(x_{i+1})$ à $y(x_i) + (x_{i+1} - x_i)\, y'(x_i)$.

On a donc, pour $0 \leqslant i \leqslant n$, $y(x_{i+1}) = y(x_i) + h y'(x_i)$.

Compte tenu de l'équation de départ, cette relation peut s'écrire

$$y(x_{i+1}) = y(x_i) + h f[x_i, y(x_i)].$$

Convenons d'écrire y_i au lieu de $y(x_i)$. La relation précédente devient :

$$y_{i+1} = y_i + h f(x_i, y_i) \text{ avec } 0 \leqslant i \leqslant n - 1.$$

Le programme est le suivant :

```
10 DEF FN f(x,y)=........
20 INPUT"x0=";x0
30 INPUT "y0=";y0
40 INPUT"x=";x
50 INPUT"n=";n
60 h=(x-x0)/n
70 x=x0:y=y0
80 FOR i=1 TO n
90 y=y+h*FN f(x,y)
100 x=x+h
110 NEXT i
120 PRINT "y=";y
```

Exemple : Soit $y' = 2y + 2x + 1$ avec $x_0 = 0$, $y_0 = 0$ et $x = 1$.

Avec $n = 50$, on obtient $y(1) = 5{,}106\,683\,34$. Pour $n = 200$, $y(1) = 5{,}316\,017\,84$ tandis que $y(1) = 5{,}374\,312\,41$ pour $n = 1\,000$.

La valeur exacte est 5,389 056.

La méthode d'Euler est lente et d'une précision très moyenne.

1.2. Méthode de Runge-Kutta

Sous les mêmes hypothèses que précédemment, on détermine cette fois la suite y_1, $y_2, \ldots, y_n$ à l'aide de la relation $y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$ pour $0 \leqslant i \leqslant n - 1$.

Les nombres k_1, k_2, k_3 et k_4 sont donnés par les formules :

- $k_1 = h\, f(x_i, y_i)$
- $k_2 = h\, f\left(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}\right)$
- $k_3 = h\, f\left(x_i + \frac{h}{2}, y_i + \frac{k_2}{2}\right)$
- $k_4 = h\, f(x_i + h, y_i + k_3)$

```
10 DEF FN f(x,y)=.........
20 INPUT"x0=";x0
30 INPUT "y0=";y0
40 INPUT"x=";x
50 INPUT"n=";n
```

```
60 h=(x-x0)/n
70 x=x0:y=y0
80 FOR i=1 TO n
90 k1=h*FN f(x,y)
100 k2=h*FN f(x+h/2,y+k1/2)
110 k3=h*FN f(x+h/2,y+k2/2)
120 k4=h*FN f(x+h,y+k3)
130 y=y+(k1+2*k2+2*k3+k4)/6
140 x=x+h
150 NEXT i
160 PRINT "y=";y
```

Exemple :

Reprenons l'équation $y' = 2y + 2x + 1$ avec $x_0 = 0$, $y_0 = 0$ et $x = 1$.

Avec $n = 50$, on obtient $y(1) = 5{,}389\,055\,8$. La précision est bien meilleure qu'avec la méthode d'Euler.

On démontre que l'erreur commise est de l'ordre de h^4

2. Équations du second ordre

Celles-ci doivent se présenter sous la forme $y'' = f(x, y, y')$ avec les conditions initiales $x = x_0$, $y(x_0) = y_0$ et $y'(x_0) = y'_0$.

Pour x donné, on calcule donc $y(x)$.

2.1. Méthode de Runge-Kutta

La suite $y_0, y_1, \ldots, y_n = y(x)$ est déterminée grâce aux deux relations suivantes :

$$y_{i+1} = y_i + h\left[y'_i + \frac{1}{6}(k_1 + k_2 + k_3) \right]$$

$$y'_{i+1} = y'_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4).$$

Les nombres k_1, k_2, k_3 et k_4 sont donnés par les formules

- $k_1 = hf(x_i, y_i, y'_i)$
- $k_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{h}{2} y'_i + \frac{k_1}{2} \right)$
- $k_3 = hf\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}y'_i + \frac{h}{4}k_1, y'_i + \frac{k_2}{2} \right)$
- $k_4 = hf\left(x_i + h, y_i + hy'_i + \frac{h}{2}k_2, y'_i + k_3 \right).$

2.2. Programme et exemple :

```
10 DEF FN f(x,y,dy)=........
20 INPUT"valeur initiale  de x";x0
30 INPUT"valeur initiale  de y";y0
40 INPUT"valeur initiale  de y'";dy0
50 INPUT "valeur de x";x
60 INPUT"valeur de n";n
70 h=(x-x0)/n
80 x=x0:y=y0:dy=dy0
90 FOR i=1 TO n
100 k1=h*FN f(x,y,dy)
110 k2=h*FN f(x+h/2,y+h*dy/2,dy+k1/2)
120 k3=h*FN f(x+h/2,y+h*dy/2+h*k1/4,dy+k2/2)
130 k4=h*FN f(x+h,y+h*dy+h*k2/2,dy+k3)
140 y=y+h*(dy+(k1+k2+k3)/6)
150 dy=dy+(k1+2*k2+2*k3+k4)/6
160 x=x+h
170 NEXT i
180 PRINT "y=";y
```

Soit à résoudre $y'' = 3y' + 2 - 6x$ avec $x_0 = 0$, $y_0 = 2$ et $y'_0 = 3$.

Calculons $y(1)$.

Avec $n = 25$, on obtient $y(1) = 22{,}085\,442\,7$. La valeur exacte est $22{,}085\,537$.

3. *Première application*

Nous emprunterons cet exemple à la **mécanique.** Soit M un point matériel de masse m astreint à se déplacer sur une droite D.

Soit O un point fixe de D.

Nous supposerons M soumis à deux forces :

- une force d'attraction : $f_1 = -k\,\overline{OM}$
- une force de résistance, proportionnelle à la vitesse : $f_2 = -h\dfrac{d\overline{OM}}{dt}$.

On désire calculer $\overline{OM}$ à chaque instant t.

3.1. Programme

Soit $x = \overline{OM}$. Le mouvement de M est régi par l'équation différentielle du second ordre $m\dfrac{dx^2}{dt^2} = -kx - h\dfrac{dx}{dt}$.

Le programme qui va suivre nous permettra de calculer x aux instants $t = 1, 2, 3,...$ etc.

Pour l'essentiel, ce programme est le même que celui du paragraphe 2. On a simplement changé le nom des variables (x au lieu de y, t au lieu de x, dx au lieu de dy) et ajouté quelques instructions pour permettre des calculs répétés.

```
10 INPUT"valeur de k";k
20 INPUT"valeur de h";h
30 DEF FN f(t,x,dx)=-k*x-h*dx
40 INPUT"valeur initiale  de t";t0
50 INPUT"valeur finale de t";tf
60 INPUT"valeur initiale  de x";x0
70 INPUT"valeur initiale  de dx/dt";dx0
80 n=10
90 dt=1/n
100 t=t0:x=x0:dx=dx0
110 FOR i=1 TO n
120 k1=dt*FN f(t,x,dx)
130 k2=dt*FN f(t+dt/2,x+dt*dx/2,dx+k1/2)
140 k3=dt*FN f(t+dt/2,x+dt*dx/2+dt*k1/4,dx+k2/2)
150 k4=dt*FN f(t+dt,x+dt*dx+dt*k2/2,dx+k3)
160 x=x+dt*(dx+(k1+k2+k3)/6)
170 dx=dx+(k1+2*k2+2*k3+k4)/6
180 t=t+dt
190 NEXT i
200 PRINT t,x
210 t0=t:x0=x:dx0=dx
220 IF t<tf THEN 100
```

3.2. Résultats

Prenons $m = 1$, $x_0 = 200$, $x'_0 = 0$, $n = 10$.

1) Avec $k = 2$ et $h = 0,5$, M accomplit des oscillations d'amplitude décroissante autour du point O.

t (en secondes)	x
0	200
1	55,24
2	− 106
3	− 62,88
4	46,87
5	51,12
6	− 14,26
7	− 34,96
8	− 1,04
9	20,92
10	6,43

2) Avec $k = 2$ et $h = 3$, M se rapproche du point O qu'il atteint théoriquement pour $t = \infty$. Il n'y a pas d'oscillations.

t (en secondes)	x
0	200
1	120
2	50,47
3	19,42
4	7,26
5	2,69
6	0,99
7	0,36
8	0,13
9	0,05
10	0,02

Remarque : il serait possible d'ajouter au programme les instructions nécessaires pour obtenir une représentation graphique de x en fonction de t.

4. Seconde application

Soit M un point matériel de masse m.

Le plan étant rapporté à un repère orthonormé d'origine O, nous supposerons M soumis chaque instant à une force $f = -k\overrightarrow{OM}$, $k > 0$.

Au temps $t = 0$, on connaît la position initiale $M_0 = (x_0, y_0)$ de M ainsi que les composantes du vecteur vitesse.

Nous nous proposons de déterminer la trajectoire de M.

4.1. Programme

Le mouvement de M obéit à l'équation $m\dfrac{\mathrm{d}^2\overrightarrow{OM}}{\mathrm{d}t} = -k\overrightarrow{OM}$.

Cette équation équivaut au **système**

$$\begin{cases} \dfrac{\mathrm{d}^2x}{\mathrm{d}t^2} = -\dfrac{k}{m}x \\ \dfrac{\mathrm{d}^2y}{\mathrm{d}t^2} = -\dfrac{k}{m}y. \end{cases}$$

Il s'agit là d'un système de deux équations du second ordre : on peut donc utiliser la méthode de Runge-Kutta.

Nous prendrons $\dfrac{k}{m} = 1$.

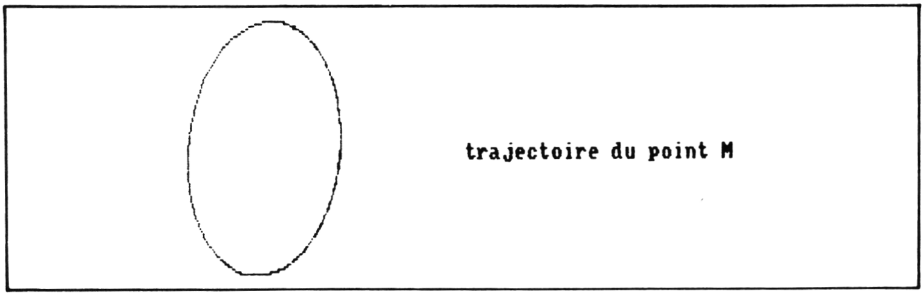
```
10 DEF FN f(t,x,dx)=-x
20 DEF FN g(t,y,dy)=-y
30 INPUT"valeur initiale  de t";t0
40 INPUT"valeur finale de t";tf
50 INPUT"valeur initiale  de x";x0
60 INPUT"valeur initiale  de y";y0
70 INPUT"valeur initiale  de vx";dx0
80 INPUT"valeur initiale  de vy";dy0
90 CLS
100 n=10
110 dt=0.1/n
120 t=t0:x=x0:dx=dx0
130 y=y0:dy=dy0
140 FOR i=1 TO n
150 k1=dt*FN f(t,x,dx)
160 k2=dt*FN f(t+dt/2,x+dt*dx/2,dx+k1/2)
170 k3=dt*FN f(t+dt/2,x+dt*dx/2+dt*k1/4,dx+k2/2)
180 k4=dt*FN f(t+dt,x+dt*dx+dt*k2/2,dx+k3)
190 x=x+dt*(dx+(k1+k2+k3)/6)
200 dx=dx+(k1+2*k2+2*k3+k4)/6
210 f1=dt*FN g(t,y,dy)
220 f2=dt*FN g(t+dt/2,y+dt*dy/2,dy+f1/2)
230 f3=dt*FN g(t+dt/2,y+dt*dy/2+dt*f1/4,dy+f2/2)
240 f4=dt*FN g(t+dt,y+dt*dy+dt*f2/2,dy+f3)
250 y=y+dt*(dy+(f1+f2+f3)/6)
260 dy=dy+(f1+2*f2+2*f3+f4)/6
270 t=t+dt
280 NEXT i
290 PLOT 300+5*x0,200+y0
300 DRAW 300+5*x,200+y
310 t0=t:x0=x:dx0=dx
320 y0=y:dy0=dy
330 IF t<tf THEN 120
```

4.2. Résultats

Prenons $x_0 = 0$, $y_0 = 100$, $v_x(0) = v_y(0) = 10$ et faisons varier t de 0 à 20 avec un pas $\Delta t = 0{,}01$.

La trajectoire de M est une **ellipse.**

Remarque : les instructions PLOT $300 + 5x_0$, $200 + y_0$ et DRAW $300 + 5x$, $200 + y$ ont été choisies à posteriori pour donner à la courbe des dimensions convenables sur l'écran.



Chapitre 13

Éléments de calcul vectoriel et de calcul matriciel

Le calcul vectoriel et le calcul matriciel jouent un rôle important dans de nombreux domaines : algèbre linéaire, géométrie, mécanique,... etc.

Dans ce chapitre, nous examinerons les opérations les plus usuelles, fastidieuses à exécuter à la main.

1. Calcul vectoriel

Nous nous limiterons à l'ensemble des vecteurs de R^3. Dans tout le paragraphe, $\vec{u}$, $\vec{v}$ et $\vec{w}$ désignerons des vecteurs de coordonnées respectives $u_1, u_2, u_3, v_1, v_2, v_3$ et w_1, w_2, w_3 relativement à une **base** $(\vec{i}, \vec{j}, \vec{k})$ **orthonormée** et **positive.**

1.1. Produit scalaire et produit vectoriel

Le produit scalaire des vecteurs $\vec{u}$ et $\vec{v}$ est le nombre $\vec{u} \cdot \vec{v} = u_1v_1 + u_2v_2 + u_3v_3$.

Le produit vectoriel de ces mêmes vecteurs est le vecteur $\vec{u} \wedge \vec{v}$ de coordonnées $u_2v_3 - u_3v_2$, $u_3v_1 - u_1v_3$ et $u_1v_2 - u_2v_1$.

```
10 DIM u(3):DIM v(3)
20 FOR i=1 TO 3
30 PRINT "u";i;"=";
```

```
40 INPUT u(i)
50 NEXT
60 FOR i=1 TO 3
70 PRINT "v";i;"=";
80 INPUT v(i)
90 NEXT i
100 ps=u(1)*v(1)+u(2)*v(2)+u(3)*v(3)
110 pv1=u(2)*v(3)-u(3)*v(2)
120 pv2=u(3)*v(1)-u(1)*v(3)
130 pv3=u(1)*v(2)-u(2)*v(1)
140 PRINT "produit scalaire=";ps
150 PRINT "coordonnees du produit vectoriel:"
160 PRINT pv1,pv2,pv3
```

Exemple : Soit $\vec{u} = \vec{i} - 2\vec{j} + 3\vec{k}$ et soit $\vec{v} = -\vec{i} + \vec{j} + 5\vec{k}$.

On a $\vec{u} \,.\, \vec{v} = 12$ et $\vec{u} \wedge \vec{v} = -13\vec{i} - 8\vec{j} - \vec{k}$.

1.2. Produit mixte

Le produit mixte des vecteurs $\vec{u}$, $\vec{v}$ et $\vec{w}$ est le nombre $\vec{u} \,.\, (\vec{v} \wedge \vec{w})$.

```
10 DIM u(3):DIM v(3):DIM w(3)
20 FOR i=1 TO 3
30 PRINT "u";i;"=";
40 INPUT u(i)
50 NEXT
60 FOR i=1 TO 3
70 PRINT "v";i;"=";
80 INPUT v(i)
90 NEXT i
100 FOR i=1 TO 3
110 PRINT "w";i;"=";
120 INPUT w(i)
130 NEXT i
140 a=u(1)*(v(2)*w(3)-v(3)*w(2))
150 b=u(2)*(v(3)*w(1)-v(1)*w(3))
160 c=u(3)*(v(1)*w(2)-v(2)*w(1))
170 PRINT "produit mixte=";a+b+c
```

Exemple : $\vec{u} = 2\vec{i} + \vec{j} + \vec{k}$, $\vec{v} = -\vec{i} + 5\vec{j} + 3\vec{k}$, $\vec{w} = 4\vec{i} - 3\vec{j} + \vec{k}$

On a $\vec{u} \,.\, (\vec{v} \wedge \vec{w}) = 24$.

1.3. Double produit vectoriel

Le double produit vectoriel des vecteurs $\vec{u}$, $\vec{v}$, $\vec{w}$, pris dans cet ordre, est le vecteur $\vec{u} \wedge (\vec{v} \wedge \vec{w})$.

```
10 DIM u(3):DIM v(3):DIM w(3)
20 FOR i=1 TO 3
30 PRINT "u";i;"=";
40 INPUT u(i)
```

```
50 NEXT
60 FOR i=1 TO 3
70 PRINT "v";i;"=";
80 INPUT v(i)
90 NEXT i
100 FOR i=1 TO 3
110 PRINT "w";i;"=";
120 INPUT w(i)
130 NEXT i
140 d=u(1)*w(1)+u(2)*w(2)+u(3)*w(3)
150 e=u(1)*v(1)+u(2)*v(2)+u(3)*v(3)
160 PRINT"coordonnees du  double produit:"
170 PRINT d*v(1)-e*w(1)
180 PRINT d*v(2)-e*w(2)
190 PRINT d*v(3)-e*w(3)
```

Exemple : $\vec{u} = \vec{i} + \vec{j} + \vec{k}$, $\vec{v} = \vec{i} - \vec{j} - \vec{k}$, $\vec{w} = -\vec{i} + \vec{j} - \vec{k}$. On a :

$$\vec{u} \wedge (\vec{v} \wedge \vec{w}) = -2\vec{i} + 2\vec{j}.$$

2. Calcul matriciel

Rappelons la définition « naïve » d'une **matrice** : il s'agit d'un ensemble de np nombres disposés en n **lignes** et p **colonnes.**

Si $n = p$, la matrice est dite **carrée.**

On note habituellement a_{ij} le nombre situé sur la $i^{ème}$ ligne et la $j^{ème}$ colonne.

Laissant de côté l'addition des matrices et la multiplication par un scalaire, nous ne traiterons que la **multiplication** des matrices.

2.1. Produit de deux matrices

Soit A une matrice à n lignes et p colonnes et soit B une matrice à q lignes et r colonnes.

Le produit AB n'existe que si $q = p$: dans ce cas, AB est une matrice à n lignes et r colonnes.

Désignons, pour $1 \leqslant i \leqslant n$ et $1 \leqslant j \leqslant r$, par c_{ij} le coefficient de AB situé sur la $i^{ème}$ ligne et la $j^{ème}$ colonne.

On a $c_{ij} = \sum_{k=1}^{k=p} a_{ij} b_{kj}$.

Il en résulte le programme suivant :

```
10 REM donnees
20 INPUT"n=";n
30 INPUT"p=";p
40 INPUT"q=";q
```

```
50 IF q<>p THEN PRINT"p different de q"
60 IF p<>q THEN 10
70 INPUT"r=";r
80 DIM a(n,p):DIM b(q,r):DIM c(n,r)
90 FOR i=1 TO n
100 FOR j=1 TO p
110 PRINT "a(";i;",";j;")=";
120 INPUT a(i,j)
130 NEXT j
140 NEXT i
150 FOR i=1 TO q
160 FOR j=1 TO r
170 PRINT "b(";i;",";j;")=";
180 INPUT b(i,j)
190 NEXT j
200 NEXT i
210 REM calcul du produit
220 FOR i=1 TO n
230 FOR j=1 TO r
240 c(i,j)=0
250 FOR k=1 TO p
260 c(i,j)=c(i,j)+a(i,k)*b(k,j)
270 NEXT k
280 PRINT "c(";i;",";j;")=";c(i,j)
290 NEXT j
300 NEXT i
```

Exemple :

Soit $A = \begin{bmatrix} 3 & 1 & 2 \\ 4 & 0 & 1 \end{bmatrix}$ et soit $B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 2 & 1 \end{bmatrix}$

$$AB = \begin{bmatrix} 7 & 3 \\ 6 & 1 \end{bmatrix}.$$

2.2. Image d'un vecteur par une matrice carrée

Soit A une matrice carrée à n lignes et n colonnes et soit x un vecteur de coordonnées $x_1, x_2, \dots, x_n$.

L'image par A du vecteur x est un vecteur y de coordonnées $y_1, y_2, \dots, y_n$.

Pour $1 \leqslant i \leqslant n$, y_i est donné par $y_i = \sum_{j=1}^{j=n} a_{ij} x_j$.

En modifiant légèrement le programme précédent, on obtient celui-ci :

```
10 INPUT"n=";n
20 DIM a(n,n):DIM x(n):DIM y(n)
30 REM coefficients de la matrice
40 FOR i=1 TO n
50 FOR j=1 TO n
60 PRINT"a(";i;",";j;")=";
70 INPUT a(i,j)
```

```
80 NEXT j
90 NEXT i
100 REM coordonnees du vecteur
110 FOR i=1 TO n
120 PRINT "x(";i;")=";
130 INPUT x(i)
140 NEXT i
150 REM calcul de y
160 FOR i=1 TO n
170 FOR j=1 TO n
180 y(i)=y(i)+a(i,j)*x(j)
190 NEXT j
200 PRINT "y(";i;")=";y(i)
210 NEXT i
```

Exemple : $A = \begin{bmatrix} 3 & 1 & 2 \\ 5 & 0 & 1 \\ -1 & 2 & 4 \end{bmatrix}$, $x = (2, 1, -1)$

On a $y = (5, 9, -4)$.

3. Inverse d'une matrice carrée

Soit A une matrice carrée d'ordre n.

Sous certaines conditions, A est **inversible** : il existe alors une matrice carrée A^{-1} d'ordre n telle que $AA^{-1} = A^{-1}A = I$ (I désigne la matrice unité d'ordre n).

On peut calculer A^{-1} par **la méthode du pivot.**

3.1. Principe de la méthode

Considérons les manipulations matricielles suivantes :

- on échange deux lignes
- on multiplie tous les coefficients d'une ligne par le même nombre non nul
- à la $i^{ème}$ ligne, on ajoute la $k^{ième}$.

On démontre que ces manipulations correspondent à des pré-multiplications de la matrice A par des matrices particulières M_1, M_2,..., M_p.

Si A^{-1} existe, on peut alors transformer la matrice A en la matrice-unité I grâce à des manipulations bien choisies. La **même suite** de manipulations faite sur I transforme celle-ci en A^{-1}.

3.2. Exemple

Prenons $A = \begin{bmatrix} 2 & 2 & 1 \\ 0 & 1 & 3 \\ 1 & 1 & 0 \end{bmatrix}$; $I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

1) Commençons par transformer la 1^re colonne de A.

Pour cela, divisons la 1^re ligne par 2 pour avoir 1 sur la diagonale

$$\begin{bmatrix} 1 & 1 & 0{,}5 \\ 0 & 1 & 3 \\ 1 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0{,}5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Retranchons maintenant la 1^re ligne à la 3^e pour avoir 0 en début de ligne :

$$\begin{bmatrix} 1 & 1 & 0{,}5 \\ 0 & 1 & 3 \\ 0 & 0 & -0{,}5 \end{bmatrix} \quad \begin{bmatrix} 0{,}5 & 0 & 0 \\ 0 & 1 & 0 \\ 0{,}5 & 0 & 1 \end{bmatrix}$$

2) Transformons maintenant la 2^e colonne. La 2^e ligne est conservée telle quelle puisqu'elle possède un 1 sur la diagonale. Retranchons la 2^e ligne à la 1^re :

$$\begin{bmatrix} 1 & 0 & -2{,}5 \\ 0 & 1 & 3 \\ 0 & 0 & -0{,}5 \end{bmatrix} \quad \begin{bmatrix} 0{,}5 & -1 & 0 \\ 0 & 1 & 0 \\ -0{,}5 & 0 & 1 \end{bmatrix}$$

3) Transformons, pour terminer, la 3^e colonne.

Commençons par diviser la 3^e ligne par $-0{,}5$ pour avoir 1 sur la diagonale :

$$\begin{bmatrix} 1 & 0 & -2{,}5 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 0{,}5 & -1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & -2 \end{bmatrix}$$

A la ligne 1, ajoutons la 3^e ligne multipliée par 2,5 puis retranchons la ligne 3 multipliée par 3 à la seconde ligne :

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad A^{-1} = \begin{bmatrix} 3 & -1 & -5 \\ -3 & 1 & 6 \\ 1 & 1 & -2 \end{bmatrix}$$

La transformation est achevée.

3.3. Programme

```
10 INPUT"n=";n
20 REM Entree des coefficients a(i,j) et des
30 REM coefficients de la matrice unite
40 DIM a(n+1,n):DIM b(n+1,n)
```

```
50 FOR i=1 TO n
60 FOR j=1 TO n
70 PRINT"a(";i;",";j;")=";
80 INPUT a(i,j)
90 b(i,j)=0
100 IF i=j THEN b(i,j)=1
110 NEXT j
120 NEXT i
130 FOR k=1 TO n
140 IF ABS(a(k,k))<=0.001 THEN c=0:GOSUB  420
150 REM Si a(k,k) non nul, on divise
160 REM la Kieme ligne par a(k,k)
170 d=a(k,k)
180 FOR j=1 TO n
190 a(k,j)=a(k,j)/d
200 b(k,j)=b(k,j)/d
210 NEXT j
220 REM On transforme a(i,k) en 0
230 REM pour i different de k
240 FOR i=1 TO n
250 IF i=k THEN 310
260 e=a(i,k)
270 FOR j=1 TO n
280 a(i,j)=a(i,j)-e*a(k,j)
290 b(i,j)=b(i,j)-e*b(k,j)
300 NEXT j
310 NEXT i
320 NEXT k
330 REM Affichage de la matrice inverse
340 FOR i=1 TO n
350 FOR j=1 TO n
360 PRINT "b(";i;",";j;")=";b(i,j)
370 NEXT j
380 NEXT i
390 END
400 REM Sous-programme d'echange des
410 REM lignes pour a(k,k)<=0.001
420 c=c+1
430 FOR j=1 TO n
440 a(n+1,j)=a(k,j)
450 b(n+1,j)=b(k,j)
460 NEXT j
470 FOR l=k TO n
480 FOR j=1 TO n
490 a(l,j)=a(l+1,j)
500 b(l,j)=b(l+1,j)
510 NEXT j
520 NEXT l
530 IF c<=n-k THEN RETURN
540 PRINT"matrice singuliere"
```

a_{ij} désigne un coefficient quelconque de A.

b_{ij} désigne d'abord un coefficient de I, puis un coefficient de A^{-1}.

Si le coefficient diagonal $a_{k,k}$ est égal à 0, on procède à l'échange de la $k^{ième}$ ligne avec l'une des lignes suivantes.

Si on a toujours $a_{k,k} = 0$ après avoir tenté tous les échanges possibles, c'est que A n'est pas inversible.

En fait, on procède à l'échange dès que $|a_{k,k}| < 0{,}001$.

Une $n + 1^{ème}$ ligne est créée pour permettre l'échange de 2 lignes.

3.4. Exemple

$$\text{Soit } A = \begin{bmatrix} 1 & 1 & 12 & 3 & 2 \\ -3 & 7 & 11 & 1 & -15 \\ -1 & 2 & 1 & 0 & -5 \\ 2 & 2 & 29 & 7 & 4 \\ 1 & 4 & 34 & 7 & 0 \end{bmatrix}$$

$$\text{Le calcul donne } A^{-1} = \begin{bmatrix} 19 & -43 & 103 & -42 & 40 \\ -1 & 10 & -24 & 8 & -9 \\ 1 & -3 & 7 & -3 & 3 \\ -7 & 15 & -35 & 16 & -15 \\ -4 & 12 & -29 & 11 & -11 \end{bmatrix}$$

Le programme donne de bonne valeurs approchées des coefficients de A^{-1}.

Par exemple $b_{3,2} = -2{,}999\,999\,95$; $b_{4,3} = -34{,}999\,999\,3$; $b_{5,4} = 10{,}999\,999\,8$.

Chapitre 14
Éléments de statistiques

Une série statistique est constituée d'un ensemble de n nombres.

On caractérise habituellement une série statistique par deux types de paramètres : **les paramètres de position** comme la médiane ou la moyenne, **les paramètres de dispersion** comme la variance et l'écart-type.

Si on dispose de deux séries statistiques de mêmes effectifs, on peut essayer de voir si elles sont **corrélées** ou non.

1. *Moyenne, variance et écart-type*

Soit $x_1, x_2, x_3, \dots, x_n$ une série statistique de n nombres.

1.1. Définitions

La série précédente a pour **moyenne** $m = \dfrac{\sum x_i}{n}$, pour **variance** $v = \dfrac{\sum (x_i - m)^2}{n}$ et pour **écart-type** $\sigma = \sqrt{v}$.

(nous avons omis dans ces définitions les indices sous et sur les signes sommes).

1.2. Programme

```
10 INPUT"combien de valeurs";n
20 DIM x(n)
30 FOR i= 1 TO n
40 PRINT "valeur numero";i;":";
50 INPUT x(i)
60 s=s+x(i)
70 p=p+x(i)*x(i)
80 NEXT i
90 m=s/n
100 v=p/n-(m*m)
110 e=SQR(v)
120 PRINT"moyenne=";m
130 PRINT"variance=";v
140 PRINT"ecart-type=";e
```

1.3. Exemple

Considérons la série 3, 7, 15, 20, 4, 4, 11, 19.

On obtient $m = 10{,}375$ et $\sigma = 6{,}479\,535\,09$. (On peut remarquer que la quasi-totalité des éléments de la série sont compris entre $m - 4\sigma$ et $m + 4\sigma$)

2. Covariance et coefficient de corrélation

Soient $x_1, x_2, ..., x_n$ et $y_1, y_2, ..., y_n$ deux séries statistiques de n éléments chacune, de moyennes respectives m_x et m_y, de variances respectives v_x et v_y.

2.1. Définitions

On appelle **covariance** des deux séries le nombre $\text{cov}(x, y) = \dfrac{\sum(x_i - m_x)(y_i - m_y)}{n}$.

On appelle **coefficient de corrélation linéaire** le nombre $r = \dfrac{\text{cov}(x, y)}{\sqrt{v_x v_y}}$ (on a $-1 \leqslant r \leqslant 1$).

2.2. Programme

```
10 INPUT"combien de valeurs";n
20 DIM x(n):DIM y(n)
```

```
30 FOR i= 1 TO n
40 PRINT "valeur de x numero";i;":";
50 INPUT x(i)
60 PRINT "valeur de y numero";i;":";
70 INPUT y(i)
80 s=s+x(i):u=u+y(i)
90 t=t+x(i)*x(i):v=v+y(i)*y(i)
100 w=w+x(i)*y(i)
110 NEXT i
120 mx=s/n:my=u/n
130 vx=t/n-(mx*mx):vy=v/n-(my*my)
140 cov=w/n-(s/n)*(u/n)
150 PRINT"variance des x=";vx
160 PRINT"variance des y=";vy
170 PRINT"covariance=";cov
180 PRINT"cefficient de correlation=";
190 PRINT cov/(SQR(vx*vy))
```

Le nombre $\text{cov}(x, y)$ est calculé à l'aide de la formule $\text{cov}(x, y) = \frac{\sum x_i y_i}{n} - m_x m_y$;

2.3. Exemples

1) série des x : 2, 3, 3, 5, 7, 15, 21
   série des y : 1, 4, 3, 7, 10, 15, 8

On obtient $\text{cov}(x, y) = 19{,}571\,428\,6$ et $r = 0{,}665\,761\,637$

2) série des x : 4, 8, 3, 7, 5, 6
   série des y : 12, 8, 20, 100, 24, 32

covariance $= 17{,}666\,666\,6$, $r = 0{,}332\,562\,487$.

3. Ajustements

Soient $x_1, x_2, \ldots, x_n$ et $y_1, y_2, \ldots, y_n$ deux séries statistiques de n nombres chacunes.

Considérons les points $P_1 = (x_1, y_1)$, $P_2 = (x_2, _2), \ldots, P_n = (x_n, y_n)$.

La méthode des moindres carrés permet de trouver une fonction f de type donné dont la courbe représentative s'ajuste le mieux possible » aux points $P_1, P_2, \ldots, P_n$.

Les caractéristiques de f sont déterminées en cherchant le minimum du nombre :

$$\sum [y_i - f(x_i)]^2.$$

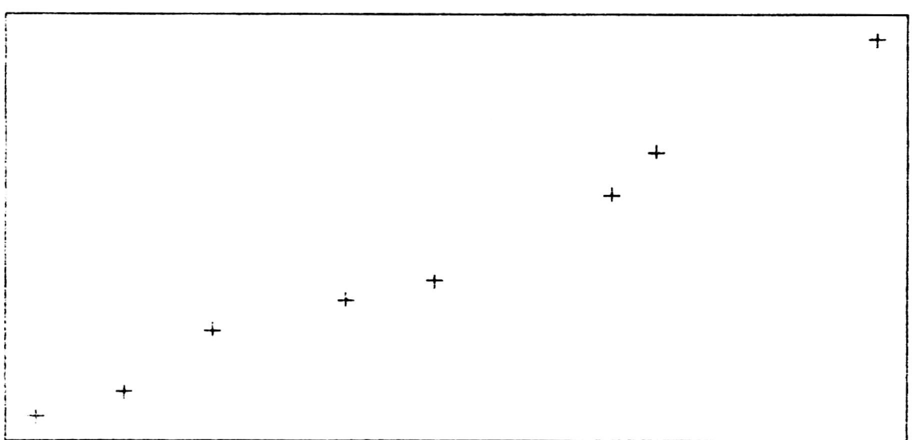
3.1. Visualisation des points $P_1, P_2, \ldots, P_n$

Avant de calculer un ajustement, il peut être intéressant de visualiser le « nuage » des points $P_1, P_2, \ldots P_n$. C'est là l'objet du programme qui suit :

```
10 REM donnees
20 INPUT"combien de points";n
30 DIM x(n):DIM y(n)
40 xmin=1E+30:xmax=-1E+30
50 ymin=1E+30:ymax=-1E+30
60 FOR i=1 TO n
70 PRINT "point numero ";i
80 INPUT "abscisse=";x(i)
90 INPUT"ordonnee =";y(i)
100 IF xmax<x(i) THEN xmax=x(i)
110 IF xmin>x(i) THEN xmin=x(i)
120 IF ymax<y(i) THEN ymax=y(i)
130 IF ymin>y(i) THEN ymin=y(i)
140 NEXT i
150 REM caracteristiques de la fenetre
160 umin=50:vmin=50
170 umax=600:vmax=350
180 CLS
190 REM dessin du cadre
200 MOVE umin-20,vmin-20:DRAW umax+20,vmin-20
210 DRAW umax+20,vmax+20:DRAW umin-20,vmax+20
220 DRAW umin-20,vmin-20
230 REM mise au format et dessin des points
240 k1=(umax-umin)/(xmax-xmin)
250 l1=umax-k1*xmax
260 k2=(vmax-vmin)/(ymax-ymin)
270 l2=vmax-k2*ymax
280 FOR i=1 TO n
290 x(i)=k1*x(i)+l1
300 y(i)=k2*y(i)+l2
310 MOVE x(i)-5,y(i)
320 DRAW x(i)+5,y(i)
330 MOVE x(i),y(i)-5
340 DRAW x(i),y(i)+5
350 NEXT i
```

Exemple :

La série des x est 1, 3, 5, 8, 10, 14, 15, 20. Celles des y est 4, 8, 18, 23, 26, 40, 47, 65. On obtient la représentation suivante :



3.2. Ajustement linéaire

L'ajustement est réalisé par une droite d'équation $y = ax + b$. On démontre que :

$$a = \frac{\text{cov}(x, y)}{v_x} \quad \text{et} \quad b = m_y - am_x.$$

Il s'agit alors de la **droite de régression de y par rapport à x.** (En échangeant x et y on obtient la droite de régression de x par rapport à y).

Le programme donnant a et b est le suivant :

```
10 INPUT"combien de valeurs";n
20 DIM x(n):DIM y(n)
30 FOR i= 1 TO n
40 PRINT "valeur de x numero";i;":";
50 INPUT x(i)
60 PRINT "valeur de y numero";i;":";
70 INPUT y(i)
80 s=s+x(i):u=u+y(i)
90 t=t+x(i)*x(i):v=v+y(i)*y(i)
100 w=w+x(i)*y(i)
110 NEXT i
120 mx=s/n:my=u/n
130 vx=t/n-(mx*mx):vy=v/n-(my*my)
140 cov=w/n-(s/n)*(u/n)
150 a=cov/vx
160 b=my-a*mx
170 PRINT"equation cherchee:";
180 PRINT a;"*x + ";b
190 c=INT(ABS(100*cov/SQR(vx*vy)))/100
200 PRINT"coefficient d'ajustement:";c
```

On a utilisé le coefficient de corrélation comme mesure de l'ajustement : plus la valeur absolue de ce coefficient est proche de 1, meilleur est l'ajustement.

En combinant les 2 programmes qui précèdent, on peut obtenir une représentation graphique de la droite d'ajustement :

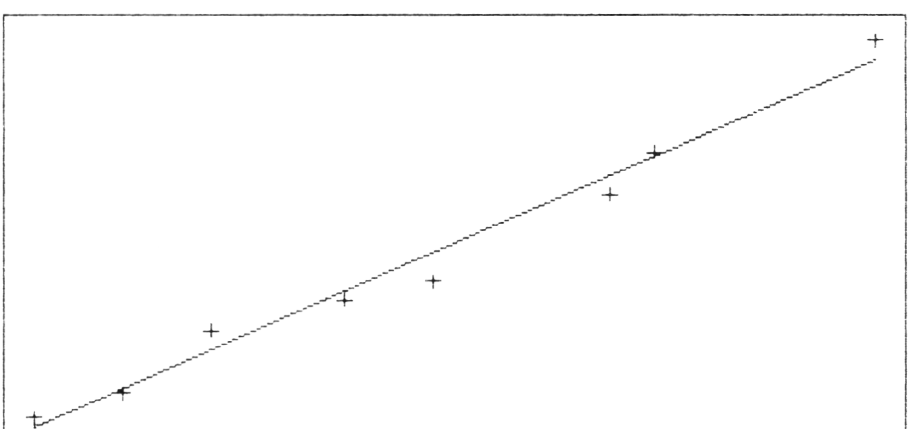
```
10 REM donnees
20 INPUT"combien de points";n
30 DIM x(n):DIM y(n)
40 xmin=1E+30:xmax=-1E+30
50 ymin=1E+30:ymax=-1E+30
60 REM calculs des differents parametres
70 FOR i=1 TO n
80 PRINT "point numero ";i
90 INPUT "abscisse=";x(i)
100 INPUT"ordonnee =";y(i)
110 s=s+x(i):u=u+y(i)
120 t=t+x(i)*x(i):v=v+y(i)*y(i)
130 w=w+x(i)*y(i)
140 IF xmax<x(i) THEN xmax=x(i)
150 IF xmin>x(i) THEN xmin=x(i)
160 IF ymax<y(i) THEN ymax=y(i)
170 IF ymin>y(i) THEN ymin=y(i)
180 NEXT i
```

```
190 mx=s/n:my=u/n
200 vx=t/n-(mx*mx):vy=v/n-(my*my)
210 cov=w/n-s*u/n/n
220 a=cov/vx:b=my-a*mx
230 REM caracteristiques de la fenetre
240 umin=50:vmin=50
250 umax=600:vmax=350
260 CLS
270 REM dessin du cadre
280 MOVE umin-20,vmin-20
290 DRAW umax+20,vmin-20
300 DRAW umax+20,vmax+20
310 DRAW umin-20,vmax+20
320 DRAW umin-20,vmin-20
330 REM mise au format et dessin des points
340 k1=(umax-umin)/(xmax-xmin)
350 l1=umax-k1*xmax
360 k2=(vmax-vmin)/(ymax-ymin)
370 l2=vmax-k2*ymax
380 FOR i=1 TO n
390 x(i)=k1*x(i)+l1
400 y(i)=k2*y(i)+l2
410 MOVE x(i)-5,y(i)
420 DRAW x(i)+5,y(i)
430 MOVE x(i),y(i)-5
440 DRAW x(i),y(i)+5
450 NEXT i
460 REM dessin de la droite
470 y1=a*xmin+b:y2=a*xmax+b
480 x1=k1*xmin+l1:x2=k1*xmax+l1
490 y1=k2*y1+l2:y2=k2*y2+l2
500 MOVE x1,y1:DRAW x2,y2
```

Exemple :

série des *x* : 1, 3, 5, 8, 10, 14, 15, 20

série des *y* : 4, 8, 18, 23, 26, 40, 47, 65



3.3. Autres ajustements

On se ramène au cas précédent pour chacun des ajustements suivants :

- ajustement par une **exponentielle** $y = a\,e^{bx}$
- ajustement par une **fonction puissance** $y = ax^b$
- ajustement de **type logarithmique** $y = a \log x + b$

1) Pour l'ajustement exponentiel $y = a\,e^{bx}$ on a, pour $y > 0$, $\log y = bx + \log \boldsymbol{a}$, ce qui ramène à un ajustement linéaire.

On apportera les modifications suivantes au premier programme du paragraphe 3.2 :

```
75 y(i)=LOG(y(i))
180 PRINT "y=";EXP(b);"EXP(";a;"*x"
```

Exemple :

série des x : 2, 4, 6, 10, 15

série des y : 5, 8, 12, 22, 54

On obtient $y = 3{,}786\,307\,48 \exp(0{,}178\,293\,413x)$ avec un coefficient d'ajustement égal à 0,99.

2) Pour un ajustement par $y = ax^b$, on doit avoir $x_i > 0$ et $y_i > 0$.

On a alors $\log y = b \log x + \log a$.

On apporte les modifications suivantes au premier programme du paragraphe 3.2 :

```
55 x(i)=LOG(x(i))
75 y(i)=LOG(y(i))
180 PRINT "y=";EXP(b);"*x↑";a
```

Exemple :

série des x : 2, 4, 5, 10, 12
série des y : 1, 3, 9, 20, 22

On obtient $y = 0{,}320\,752\,047x^{1{,}77765293}$ avec 0,97 comme coefficient d'ajustement.

3) Pour un ajustement de type logarithmique $y = a \log x + b$, on doit avoir $x_i > 0$.

n appportera les modifications suivantes au premier programme du paragraphe 3.2 :

```
55 x(i)=LOG(x(i))
180 PRINT "y=";a;"LOG(x)+";b
```

Exemple :

série des x : 6, 9, 16, 22, 40
série des y : 1, 2, 3, 4, 5

On obtient $y = 2{,}118\,029\,35 \log(x) + 2{,}736\,256\,65$ avec un coefficient d'ajustement égal à 0,99.

4. Histogrammes

Soit $x_1, x_2, ..., x_n$ une série de n nombres : cette série peut être représentée par un histogramme de k classes ayant toutes la même amplitude.

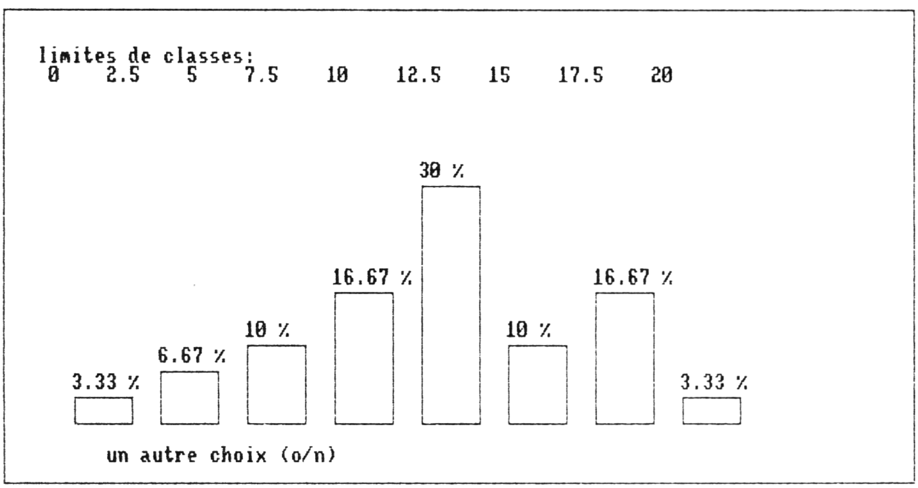
4.1. Programme

Le nombre des classes peut être choisi entre 2 et 12.

```
10 REM entree des x(i);recherche du maximum
20 REM  et du minimum des x(i)
30 INPUT "quel est l'effectif de la serie ?",n
40 DIM x(n)
50 xmax=-1E+30:xmin=1E+30
60 FOR i=1 TO n
70 PRINT " nombre numero ";i;"= ";
80 INPUT x(i)
90 IF x(i)>xmax THEN xmax=x(i)
100 IF x(i)<xmin THEN xmin=x(i)
110 NEXT i
120 REM choix du nombre k des classes
130 INPUT "Combien faut-il de classes";k
140 IF k>10 THEN la=30 ELSE la=40
150 IF k>10 THEN ecart=15 ELSE ecart=20
160 REM calcul des limites l(i) et des
170 REM effectifs e(i) des classes
180 DIM l(k+1):DIM e(k)
190 FOR i=1 TO k
200 l(i)=xmin+(i-1)*(xmax-xmin)/k
210 NEXT i
220 l(k+1)=xmax+1
230 FOR j=1 TO n
240 FOR i=1 TO k
250 IF x(j)>=l(i) AND x(j)<l(i+1) THEN e(i)=e(i)+1
260 NEXT i
270 NEXT j
280 REM calcul des frequences de classe
290 REM en pourcentage
300 DIM f(k)
310 fmax=0
320 FOR i=1 TO k
330 f(i)=100*e(i)/n
340 IF f(i)>fmax THEN fmax=f(i)
350 NEXT i
360 REM calcul des hauteurs des rectangles
370 DIM h(k)
380 FOR i=1 TO k
390 h(i)=200*f(i)/fmax
400 f(i)=INT (0.5+100*f(i))/100
410 NEXT i
420 CLS
430 REM dessin
```

```
440 x=50
450 FOR i=1 TO k
460 MOVE x,50:DRAW x+la,50
470 DRAW x+la,50+h(i):DRAW x,50+h(i)
480 DRAW x,50
490 MOVE x-10,70+h(i)
500 TAG:PRINT f(i);"%";
510 TAGOFF
520 x=x+la+ecart
530 NEXT i
540 PRINT"limites de classes:"
550 FOR i=1 TO k+1
560 IF i<>k+1 THEN PRINT l(i),
565 IF i=k+1 THEN PRINT l(i)-1
570 NEXT i
580 REM on peut changer le nombre des classes
590 LOCATE 10,24
600 INPUT "un autre choix (o/n)";a$
610 IF a$="n" THEN END
620 ERASE f
630 ERASE h
640 ERASE l
650 ERASE e
660 GOTO 130
```

4.2. Exemple



Chapitre 15
Lois usuelles de probabilité

Le calcul des probabilités utilise différents modèles mathématiques pour relier des événements à des probabilités.

La **loi binomiale,** la **loi de Poisson** permettent de traiter le cas des événements discrets tels que le nombre des pièces défectueuses d'un échantillon, le nombre des accidents du travail qui se produisent en un lieu et en un jour donnés.

Au contraire, la **loi de Gauss** concerne des variables aléatoires à variation continue.

1. Factorielles et coefficients binominaux

On utilise constamment, en calcul des probabilités, des factorielles ou des coefficients binomiaux. Nous pensons utile d'en rappeler le calcul.

1.1. Factorielles

Le programme qui suit permet de calculer $n!$ pour $n \leqslant 33$:

```
10 INPUT "valeur de n:";n
20 f=1
30 FOR i=1 TO n
```

```
40 f=f*i
50 NEXT i
60 PRINT n;"!=";f
```

On peut néanmoins dépasser cette limitation en remarquant que le logarithme décimal de $n!$ est égal à $\sum_{1}^{n} \log_{10} i$.

Soit y ce logarithme. On peut alors écrire $n! = 10^{y - \text{int}(y)} \times 10^{\text{int}(y)}$:

```
10 INPUT "valeur de n";n
20 IF n<34 THEN GOSUB 40  ELSE GOSUB 100
30 END
40 f=1
50 FOR i=1 TO n
60 f=f*i
70 NEXT
80 PRINT n;"!=";f
90 RETURN
100 FOR i=1 TO n
110 y=y+LOG10(i)
120 NEXT
130 PRINT n;"!=";(10^(y-INT(y)));"E+";INT(y)
140 RETURN
```

Exemples :

$8! = 40\,320$

$10! = 3\,628\,800$

$100! = 9{,}332\,624 \times 10^{157}$

1.2. Coefficients binomiaux

Pour $0 \leqslant k \leqslant n$, on a par définition :

$$C_n^k = \frac{n!}{k!(n-k)!}$$

On peut calculer C_n^k par récurrence en remarquant que $C_n^k = C_n^{k-1} \times \dfrac{n-k+1}{k}$ pour $k \geqslant 1$.

```
10 INPUT "valeur de k";k
20 INPUT "valeur de n";n
30 IF k>n THEN 10
35 c=1
40 IF k=0 THEN 80
50 FOR j=1 TO k
60 c=c*(n-j+1)/j
70 NEXT
80 PRINT "Coefficient binomial=";c
```

Comme pour les factorielles, on arrive assez vite à un dépassement de capacité.

Posons donc $y = \log_{10}(C_n^k)$. On a alors $y = \sum_1^k \log_{10}\left(\frac{n-i+1}{i}\right)$ d'où :

$$C_n^k = 10^{y - \mathrm{int}(y)} \times 10^{\mathrm{int}(y)} :$$

```
10 INPUT "valeur de k";k
20 INPUT "valeur de n";n
30 IF k>n THEN 10
40 IF k=0 THEN PRINT"coefficient binomial=1"
50 IF k=0 THEN END
60 FOR j=1 TO k
70 y=y+LOG10((n-j+1)/j )
80 NEXT
90 IF y>8 THEN 130
100 PRINT "coefficient binomial=";
110 PRINT INT(0.5+10^y)
120 END
130 PRINT "Coefficient binomial=";
140 PRINT 10^(y-INT(y));"E+";INT(y)
```

Exemples :

1) $C_5^0 = 1$, $C_9^1 = 9$, $C_{10}^4 = 210$, $C_{20}^{15} = 15\,504$, $C_{30}^{15} = 1{,}551\,175 \times 10^8$

2) La probabilité de tirer quatre as en choisissant au hasard quatre cartes parmi trente-deux est $\frac{1}{C_{32}^4} = \frac{1}{35\,960} = 0{,}000\,028$.

2. *Loi binomiale*

Cette loi est également connue sous le nom de **loi de l'alternative répétée.**

2.1. Définition

Considérons une série d'épreuves : chaque fois, un événement E peut se réaliser avec une probabilité q. Cet événement est distribué selon une loi binomiale si la probabilité p de le voir réalisé k fois exactement en n épreuves est :

$$p = C_n^k q^k (1-q)^{n-k}.$$

2.2. Programme

Nous calculerons p par récurrence sur k. Pour $k \geqslant 1$, $C_n^k = C_n^{k-1} \times \frac{n-k+1}{k}$.

On peut donc écrire, pour $q \neq 1$:

$$C_n^k q^k (1-q)^{n-k} = C_n^{k-1} q^{k-1} (1-q)^{n-k+1} \times \frac{n-k+1}{k} \times \frac{q}{1-q}.$$

Le programme est le suivant :

```
10 INPUT"q=";q
20 INPUT"k=";k
30 INPUT"n=";n
40 p=(1-q)^n
50 IF k=0 THEN 90
60 FOR i=1 TO k
70 p=p*q*(n-i+1)/i/(1-q)
80 NEXT
90 PRINT "p=";p
```

2.3. Exemples

1) On lance un dé dix fois de suite. La probabilité d'avoir cinq fois le six est égale à $C_{10}^5 \left(\frac{1}{6}\right)^5 \left(\frac{5}{6}\right)^5$ soit 0,013.

2) La probabilité d'obtenir dix fois de suite un six en dix lancers est égale à $C_{10}^{10} \left(\frac{1}{6}\right)^{10} \left(\frac{5}{6}\right)^0$ soit 0,000 000 016.

3. Loi de Poisson

Considérons un événement qui peut survenir 0, 1, 2,..., n fois dans l'intervalle de temps Δt et supposons qu'il se réalise en moyenne m fois pendant cet intervalle de temps. La loi de Poisson permet de calculer la probabilité de voir l'événement se réaliser k fois exactement pendant le temps Δt.

La loi de Poisson permet également d'étudier la dispersion aléatoire de points sur une droite, dans un plan, dans l'espace.

3.1. Définition

Soit x une variable aléatoire ne pouvant prendre que les valeurs entières 0, 1, 2,..., n. x est distribuée selon une loi de Poisson **de paramètre *m*** si la probabilité p qu'elle prenne la valeur k est donnée par $p = \frac{m^k}{k!} e^{-m}$.

(m est un nombre positif).

3.2. Calcul de *p*

Remarquons que $\frac{m^k}{k!} e^{-m} = e^{-m} \prod_{j=1}^{j=k} \frac{m}{j}$. Le programme est le suivant :

```
10 INPUT " valeur du parametre m ";m
20 INPUT "valeur de l'entier k";k
30 p=EXP(-m)
40 IF k=0 THEN 80
50 FOR j=1 TO k
60 p=p*m/j
70 NEXT
80 PRINT"p=";p
```

3.3. Exemples

1) Un appareil a, en moyenne, une panne tous les deux ans. Quelle est la probabilité qu'il n'ait aucune panne en deux mois de fonctionnement?

$$m = \frac{1}{12} = 0{,}083 \text{ et } k = 0 : p = 0{,}920.$$

2) le fil d'un métier à tisser se casse en moyenne 0,3 fois par heure de fonctionnement du métier. Quelle est la probabilité p d'avoir moins de 2 cassures en huit heures de travail? On doit prendre ici $m = 8 \times 0{,}3 = 2{,}4$. La probabilité de n'avoir aucune cassure en huit heures est $p_0 = 0{,}0907$.

Celle d'avoir une cassure est $p_1 = 0{,}2177$.

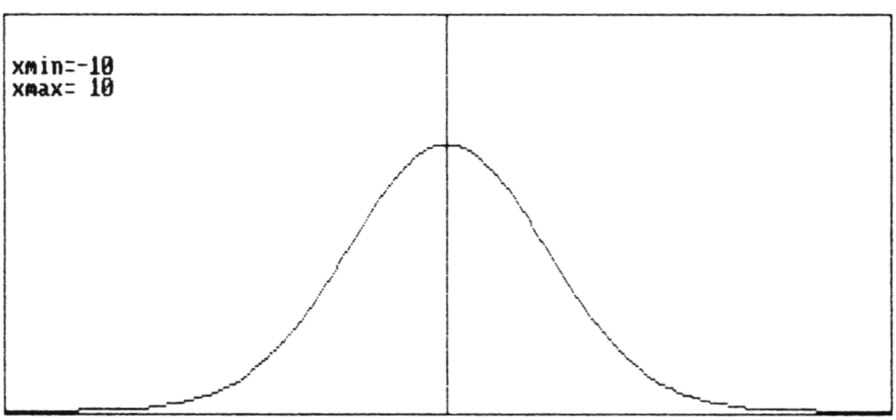
On a donc $p = p_0 + p_1 = 0{,}3084$.

3) La densité moyenne des microbes nocifs est 200 par mètre cube d'air. Quelle est la probabilité p d'avoir un microbe exactement dans un échantillon de 1 litre d'air?

Avec $m = \frac{200}{1\,000}$ et $k = 1$, on obtient $p = 0{,}303$.

4. *Loi de Gauss*

Cette loi, connue sous le nom de **loi normale** est représentée graphiquement à l'aide de la célèbre « **courbe en cloche** ».



4.1. Définition

Soit x une variable aléatoire continue et soit a un nombre réel quelconque.

x est distribuée selon une **loi normale** de **moyenne m** et **d'écart-type σ** si la probabilité p d'avoir $x \leqslant a$ est donnée par $p = \dfrac{1}{\sigma\sqrt{2\pi}} \displaystyle\int_{-\infty}^{a} \exp\left[-\frac{(x-m)^2}{2\sigma^2}\right] dx$.

4.2. Calcul de p

Faisons le changement de variable $u = \dfrac{x-m}{\sigma}$ et posons $z = \dfrac{a-m}{\sigma}$.

On obtient $p = \dfrac{1}{\sqrt{2\pi}} \displaystyle\int_{-\infty}^{z} \exp\left(-\frac{u^2}{2}\right) \mathrm{d}u$.

Comme $\dfrac{1}{\sqrt{2\pi}} \displaystyle\int_{-\infty}^{0} \exp\left(-\frac{u^2}{2}\right) \mathrm{d}u = 0{,}5$, il vient finalement :

$p = 0{,}5 + \dfrac{1}{\sqrt{2\pi}} \displaystyle\int_{0}^{z} \exp(-u^2/2)\, \mathrm{d}u$.

Au lieu de calculer l'intégrale globalement, on peut utiliser un développement en série de $e^{-u^2/2}$ et intégrer terme à terme ce développement.

On obtient alors $p = 0{,}5 + \dfrac{1}{\sqrt{2\pi}} \displaystyle\sum_{0}^{\infty} (-1)^k \frac{z^{2k+1}}{k!2^k(2k+1)}$.

Posons maintenant $c_k = (-1)^k \dfrac{z^{2k+1}}{k!2^k(2k+1)}$. Pour $k \geqslant 1$, on peut écrire :

$$c_k = -c_{k-1} \frac{(2k-1)z^2}{2k(2k+1)}$$

avec $c_0 = \dfrac{z}{\sqrt{2\pi}}$.

Le programme qui en résulte est le suivant :

```
10 INPUT "moyenne=";m
20 INPUT "ecart-type=";s
30 INPUT "valeur de a=";a
40 z=(a-m)/s
50 IF ABS(z)>4 THEN 150
60 n=10
70 IF ABS(z)>=2 THEN n=25
80 c=z:t=z
90 FOR k=1 TO n
100 c=-c*(2*k-1)*z*z/(2*k*(2*k+1))
110 t=t+c
120 NEXT k
130 p=0.5+t/SQR(2*PI)
140 GOTO 170
150 IF SGN(z)=-1 THEN p=0
160 IF SGN(z)=1 THEN p=1
170 PRINT "probabilite d'avoir x<=";a;" =";p
```

Dans ce programme on a fait varier k de 0 à 10 quand $|z| < 2$ et de 0 à 25 quand $2 \leqslant |z| \leqslant 4$.

Pour $|z| > 4$, on a pratiquement $p = 0$.

4.3. Calcul inverse

Soit p un nombre compris entre 0,000 1 et 0,999 9. Pour quelle valeur de a la probabilité d'avoir $x \leqslant a$ vaut-elle p?

```
10 INPUT "moyenne=";m
20 INPUT "ecart-type=";s
30 INPUT "valeur de p=";p
40 h=-4:b=4
50 FOR i=1 TO 100
60 z=(h+b)/2
70 n=10
80 IF ABS(z)>-2 THEN n=25
90 c=z:t=z
100 FOR k=1 TO n
110 c=-c*(2*k-1)*z*z/(2*k*(2*k+1))
120 t=t+c
130 NEXT k
140 q=0.5+t/SQR(2*PI)
150 IF ABS(p-q)<=0.000001 THEN 190
160 IF q>p THEN b=z
170 IF q<p THEN h=z
180 NEXT i
190 a=z*s+m
200 PRINT "a=";a
```

Dans ce programme, on a utilisé la propriété suivante : on a sensiblement $p = 0$ pour $x < m - 4\sigma$ et $p = 1$ pour $x > m + 4\sigma$. Il suffit donc, entre ces deux valeurs extrêmes, de calculer a **par dichotomie.**

Ainsi, pour $m = 2$, $s = 6$ et $p = 0{,}975$ on obtient $a = 13{,}76$.

4.4. Exemples

1) On installe 5 000 lampadaires neufs dans une ville. Les lampes ont une durée de fonctionnement distribuée selon une loi normale de moyenne $m = 2\,000$ heures, d'écart-type $\sigma = 300$ heures.

A combien peut-on estimer le nombre des lampes qui fonctionneront moins de 1 500 heures?

Prenons $a = 1\,500$. Le programme du paragraphe 4.2 donne $p = 0{,}048$. Ceci correspond à $5\,000 \times 0{,}048 = 239$ lampes.

2) Après combien d'heures de fonctionnement pourrons nous estimer que 25 % des lampes sont hors d'usage?

Le programme du paragraphe 4.3 donne, pour $p = 0{,}25$, $a = 1\,798$ heures.

Chapitre 16

Tests statistiques

Les tests statistiques permettent d'**accepter** ou de **rejeter** certaines hypothèses faites sur une population dont on a pu observer un ou plusieurs échantillons.

Nous examinerons dans ce chapitre deux tests choisis parmi les plus usuels : le test du χ^2 et le test du t de Student-Fischer.

1. Le test du χ^2

Il sert notamment à éprouver la **conformité** d'une distribution expérimentale à une distribution théorique.

1.1. Définitions

Soit x une variable aléatoire positive ou nulle. Cette variable est distribuée selon la **loi du χ^2 à n degrés de liberté** si la probabilité p d'avoir $x \leqslant a$ est donnée par :

$$p = \frac{1}{2^{n/2}\,\Gamma(n/2)} \int_0^a e^{-\frac{x}{2}} x^{\frac{n}{2}-1}\,dx$$

Dans cette formule, n est un entier $\geqslant 1$.

Γ désigne la **fonction gamma,** définie pour $x \geqslant 0$, par $\Gamma(x) = \displaystyle\int_0^{\infty} e^{-t} t^{x-1}\,dt$.

1.2. Calcul de *p*

On démontre que

$$p = \left(\frac{a}{2}\right)^{n/2} \frac{e^{-a/2}}{\frac{n}{2}\Gamma\left(\frac{n}{2}\right)} \left[1 + \sum_{k=1}^{k=\infty} \frac{a^k}{(n+2)(n+4)...(n+2k)}\right]$$

Pour calculer $\Gamma\left(\frac{n}{2}\right)$, nous utiliserons une propriété classique de la fonction Γ :

- $\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}$
- pour n impair et $n \geqslant 3$,

$$\Gamma\left(\frac{n}{2}\right) = \left(\frac{n-2}{2}\right) \times \left(\frac{n-4}{2}\right) ... \times \frac{5}{2} \times \frac{3}{2} \times \Gamma\left(\frac{1}{2}\right)$$

- pour n pair, $\Gamma\left(\frac{n}{2}\right) = \left(\frac{n}{2} - 1\right)!$

Afin de ne pas être limité à des valeurs de n trop faibles, nous utiliserons les logarithmes.

Posons $c_k = \dfrac{a^k}{(n+2)(n+4)...(n+2k)}$. On a $c_k = c_{k-1} \times \dfrac{a}{n+2k}$ pour $k \geqslant 2$, avec $c_1 = \dfrac{a}{n+2}$.

A priori nous ferons varier k de 1 à 1 000, mais en fait le programme sera arrêté dès que c_k sera inférieur à 10^{-8}.

```
10 INPUT "nombre des degres de liberte";n
20 INPUT "valeur de a";a
30 m=n/2
40 IF m=INT(m) THEN 110
45 REM calcul de log gamma(n/2) pour n impair
50 l=LOG(PI)/2
60 IF n=1 THEN 160
70 FOR i=1 TO n-2 STEP 2
80 l=l+LOG(i)-LOG(2)
90 NEXT i
100 GOTO 160
105 REM calcul de log gamma(n/2) pour n pair
110 l=0
120 IF n=2 THEN 160
130 FOR i=1 TO m-1
140 l=l+LOG(i)
150 NEXT i
155 REM calcul des c(k) puis de la somme s
160 c=1:s=1
170 FOR k=1 TO 1000
180 c=c*a/(n+2*k)
190 s=s+c
```

```
200 IF c<=0.000001 THEN 220
210 NEXT k
215 REM affichage du resultat
220 p=s*EXP(m*LOG(a/2)-(a/2)-1-LOG(m))
230 PRINT "probabilite cherchee:";p
```

Prenons par exemple $n = 4$ et $a = 5$: on obtient $p = 0{,}712\,702\,95$.

Pour $n = 3$ et $a = 12$ on obtient $p = 0{,}992\,616\,838$.

1.3. Calcul inverse

Soit p un nombre compris entre 0 et 0,999 : pour quelle valeur de a la probabilité d'avoir $x \leqslant a$ est-elle égale à p?

```
10 INPUT"nombre de degres de liberte";n
20 IF n<1 THEN 10
30 INPUT"valeur de p";p
40 IF p<0 THEN 30
50 IF p>0.999 THEN 30
60 h=0:b=n+8*SQR(2*n)
70 m=n/2
80 IF m=INT(m) THEN 150
90 l=LOG(PI)/2
100 IF n=1 THEN 200
110 FOR i=1 TO n-2 STEP 2
120 l=l+LOG(i)-LOG(2)
130 NEXT i
140 GOTO 200
150 l=0
160 IF n=2 THEN 200
170 FOR i=1 TO m-1
180 l=l+LOG(i)
190 NEXT i
200 a=(b+h)/2
210 c=1:s=1
220 FOR k=1 TO 1000
230 c=c*a/(n+2*k)
240 s=s+c
250 IF c<=0.00000001 THEN 270
260 NEXT k
270 q=s*EXP(m*LOG(a/2)-(a/2)-l-LOG(m))
280 IF ABS(p-q)<=0.000001 THEN 320
290 IF p<q THEN b=a
300 IF p>q THEN h=a
310 GOTO 200
320 PRINT"a=";INT(1000*a)/1000
```

Ce programme est basé sur la remarque suivante : la moyenne et l'écart-type de la distribution sont respectivement n et $\sqrt{2n}$: on a donc pratiquement $p = 1$ pour $x > n + 8\sqrt{2n}$.

Comme $p = 0$ pour $a = 0$, il suffit de déterminer a **par dichotomie** entre ces deux valeurs extrêmes.

2.2. Calcul de *p*

On a $p = 0{,}5 + \frac{1}{\sqrt{n\pi}} \frac{\Gamma\left(\frac{n+1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)} \int_0^a \left(1 + \frac{x^2}{n}\right)^{-\frac{n+1}{2}} \mathrm{d}x.$

En posant $x = \sqrt{n}\ \mathrm{tg}\ u$, il vient $p = 0{,}5 + k \int_0^{\varphi} \cos^{n-1} u\, \mathrm{d}u$ avec $\varphi = \mathrm{Arctg} \frac{a}{\sqrt{n}}$

et $k = \frac{1}{\sqrt{\pi}} \frac{\Gamma\left(\frac{n+1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)}$.

Si J_{n-1} désigne $\int_0^{\varphi} \cos^{n-1} u\, \mathrm{d}u$ on a, par récurrence sur n :

$$J_{n-1} = \frac{n-2}{n-1} J_{n-3} + \frac{\cos^{n-2}\varphi \sin \varphi}{n-1} \quad \text{pour } n \geqslant 3.$$

Le programme est le suivant :

```
380 GOTO 410
390 p=0.5+e*phi/PI
400 GOTO 420
410 p=0.5+e*d*j*EXP(r-l)
420 PRINT"probabilite:";p
10 INPUT "nombre des degres de liberte";n
20 INPUT "valeur de a";a
30 m=n/2
40 phi=ATN(ABS(a/SQR(n)))
50 s=SIN(phi):c=COS(phi)
60 d=1/SQR(PI)
70 e=SGN(a)
80 IF m<>INT(m) THEN 300
90 IF n=2 THEN 270
100 REM 1er cas:n est pair
110 l=0
120 REM calcul de l=log gamma(n/2) et de
130 REM             r=log gamma((n+1)/2)
140 FOR i=1 TO m-1
150 l=l+LOG(i)
160 NEXT i
170 r=0.5*LOG(PI)
180 FOR i=1 TO n-1 STEP 2
190 r=r+LOG(i)-LOG(2)
200 NEXT i
210 REM calcul de l'integrale J(n-1)
220 j=s
230 FOR i=4 TO n STEP 2
240 j=((i-2)*j+s*(c^(i-2)))/(i-1)
250 NEXT i
260 GOTO 480
270 p=0.5+e*s/2
```

Ainsi, on obtient $a = 14{,}86$ pour $n = 4$ et $p = 0{,}995$.

Pour $n = 5$ et $p = 0{,}99$ on obtient $a = 15{,}086$.

1.4. Application

Pour étudier la conformité de N valeurs expérimentales $v_1, v_2, \ldots, v_N$ à N valeurs théoriques $t_1, t_2, \ldots, t_N$ on calcule le nombre $\chi^2 = \sum_{k=1}^{k=N} \frac{(v_k - t_k)^2}{t_k}$.

Une fois choisi un **seuil de confiance** p, on peut utiliser le programme du paragraphe 1.3 pour déterminer le nombre a tel que la probabilité d'avoir $\chi^2 \leqslant a$ soit p.

On peut alors accepter ou rejeter la conformité des deux séries de valeurs.

On doit prendre $n = N - 1$ comme nombre des degrés de liberté.

Exemple : On lance 40 fois de suite une pièce de monnaie et on obtient 10 fois pile et 30 fois face. Peut-on considérer que la pièce est truquée?

Les valeurs observées sont respectivement 10 et 30; les valeurs théoriques étant 20 et 20.

Soit $\chi^2 = \frac{(10-20)^2}{20} + \frac{(30-20)^2}{20} = 10$.

Choisissons $p = 0{,}99$. Comme $n = 2 - 1 = 1$, on obtient alors $a = 6{,}634$. Il y a donc seulement 1 chance sur cent pour que χ^2 dépasse 6,634 : on peut accepter l'idée que la pièce est truquée.

2. Le test du t de Student

En général, on utilise ce test pour vérifier l'**homogénéité** de deux échantillons d'effectifs faibles (moins de 60).

2.1. Définitions

Soit x une variable aléatoire. Cette variable est distribuée selon la **loi de Student-Fischer à *n* degrés de liberté** lorsque la probabilité p d'avoir $x \leqslant a$ est donnée par :

$$p = \frac{1}{\sqrt{n\pi}} \frac{\Gamma\left(\frac{n+1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)} \int_{-\infty}^{a} \left(1 + \frac{x^2}{n}\right)^{-\frac{n+1}{2}} \mathrm{d}x.$$

Dans cette formule, n est un entier $\geqslant 1$.

```
280 GOTO 490
290 REM 2eme cas: n est impair
300 IF n=1 THEN 460
310 l=0.5*LOG(PI)
320 FOR i=1 TO n-2 STEP 2
330 l=l+LOG(i)-LOG(2)
340 NEXT i
350 r=0
360 FOR i=1 TO (n-1)/2
370 r=r+LOG(i)
380 NEXT i
390 REM calcul de l'integrale J(n-1)
400 j=phi
410 FOR i=3 TO n STEP 2
420 j=((i-2)*j+s*(c^(i-2)))/(i-1)
430 NEXT i
440 REM affichage des resultats
450 GOTO 480
460 p=0.5+e*phi/PI
470 GOTO 490
480 p=0.5+e*d*j*EXP(r-l)
490 PRINT"probabilite:";p
```

Pour $n = 3$ et $a = 7$, on obtient $p = 0{,}997\,006\,872$.

Avec $n = 8$ et $a = 3$, $p = 0{,}991\,464\,16$.

2.3. Calcul inverse

Soit p un nombre compris entre 0,000 1 et 0,999 9. Quelle est la valeur de a pour laquelle la probabilité d'avoir $x \leqslant a$ est p?

```
10 INPUT"nombre des degres de liberte:";n
20 INPUT"probabilite:";p
30 IF p<0.0001 OR p>0.9999 THEN 20
40 IF n>=3 THEN h=100*SQR(n/(n-2))
50 IF n>=3 THEN g=-100*SQR(n/(n-2))
60 IF n=1 OR n=2 THEN h=10000
70 IF n=1 OR n=2 THEN g=-10000
80 FOR k=1 TO 1000
90 a=(h+g)/2
100 m=n/2
110 phi=ATN(ABS(a/SQR(n)))
120 s=SIN(phi):c=COS(phi)
130 d=1/SQR(PI):e=SGN(a)
140 IF m<>INT(m) THEN 310
150 IF n=2 THEN 290
160 l=0
170 FOR i=1 TO m-1
180 l=l+LOG(i)
190 NEXT i
200 r=0.5*LOG(PI)
210 FOR i=1 TO n-1 STEP 2
220 r=r+LOG(i)-LOG(2)
230 NEXT i
```

```
240 j=s
250 FOR i=4 TO n STEP 2
260 j=((i-2)*j+s*(c^(i-2)))/(i-1)
270 NEXT i
280 GOTO 470
290 q=0.5+e*s/2
300 GOTO 480
310 IF n=1 THEN 450
320 l=0.5*LOG(PI)
330 FOR i=1 TO n-2 STEP 2
340 l=l+LOG(i)-LOG(2)
350 NEXT i
360 r=0
370 FOR i=1 TO (n-1)/2
380 r=r+LOG(i)
390 NEXT i
400 j=phi
410 FOR i=3 TO n STEP 2
420 j=((i-2)*j+s*(c^(i-2)))/(i-1)
430 NEXT i
440 GOTO 470
450 q=0.5+e*phi/PI
460 GOTO 480
470 q=0.5+e*d*j*EXP(r-l)
480 IF q>p THEN h=a
490 IF q<p THEN g=a
500 IF ABS(p-q)<0.0001 THEN 520
510 NEXT k
520 PRINT "a=";SGN(a)*INT(0.5+1000*ABS(a))/1000
```

Dans ce programme, on a utilisé le fait que la distribution de Student a pour moyenne 0 et pour écart-type $\sqrt{\dfrac{n}{n-2}}$ pour $n \geqslant 3$.

On a donc sensiblement $p = 0$ pour $x < -100\sqrt{\dfrac{n}{n-2}}$ et $p = 1$ pour $x > 100\sqrt{\dfrac{n}{n-2}}$. On calcule a **par dichotomie** entre ces deux valeurs.

Pour $n = 1$ ou $n = 2$, on calcule a entre 10^{-5} et 10^{5}.

Soient, par exemple, $n = 1$ et $p = 0{,}95$: on a $a = 6{,}313$.

Pour $n = 15$ et $p = 0{,}99$, on obtient $a = 2{,}603$.

2.4. Application

Soient deux échantillons de faibles effectifs, n_1 et n_2 ayant pour moyennes respectives m_1 et m_2, pour écarts-types respectifs σ_1 et σ_2.

Pour tester l'homogénéité de ces deux échantillons, on calcule le nombre :

$$t = \frac{|m_1 - m_2|}{s\sqrt{\dfrac{1}{n_1} + \dfrac{1}{n_2}}} \quad \text{avec} \quad s^2 = \frac{(n_1 - 1)\sigma_1^2 + (n_2 - 1)\sigma_2^2}{n_1 + n_2 - 2}.$$

Une fois choisi un **seuil de confiance** p, on peut utiliser le programme du paragraphe 2.3 pour déterminer le nombre a tel que la probabilité d'avoir $t \leqslant a$ soit p. On peut alors accepter ou rejeter l'hypothèse de conformité des échantillons.

On doit prendre un nombre de degré de liberté égal à $n_1 + n_2 - 2$.

Exemple : le tableau suivant indique les résultats de deux groupes d'élèves à une même série d'épreuves.

	groupe A	groupe B
effectif	25	22
note moyenne	14	12
écart-type	4	5

Peut-on considérer ces deux groupes comme homogènes?

Calculons t. On a $s^2 = \dfrac{24 \times 4^2 + 21 \times 5^2}{25 + 22 - 2} = 20{,}2$.

D'où $t = \dfrac{2}{s\sqrt{\dfrac{1}{25} + \dfrac{1}{22}}} = 1{,}522$.

Choisissons $p = 0{,}95$. Avec 45 degrés de liberté, on obtient alors $a = 1{,}68$: a est supérieur à t.

Avec 5 % de risque d'erreur, on peut refuser de considérer ces deux groupes comme homogènes.

Chapitre 17
Méthodes diverses

On peut quelquefois résoudre des problèmes de calcul à l'aide de méthodes assez rudimentaires, basées sur l'exploitation de la puissance de calcul d'un ordinateur. On peut par exemple **explorer la totalité des cas possibles** ou bien encore utiliser une **simulation.**

Dans ce dernier chapitre nous donnerons des indications sur ce genre de méthode en traitant quelques problèmes simples relevant de la théorie des nombres, du calcul des probabilités, du calcul différentiel.

1. Décomposition d'un entier en une somme de deux carrés

Proposons-nous de chercher les entiers naturels inférieurs à 1 000 qui s'écrivent de deux manières au moins comme somme de deux carrés non nuls.

1.1. Méthode

Elle est excessivement simple : il nous suffit d'examiner le cas de chaque entier compris entre 1 et 1 000.

Soit n un tel entier : posons $n = y^2 + z^2$. Pour tout entier y compris entre 1 et $\sqrt{n}$, on calculera $c = n - y^2$. S'il existe un entier z tel que $c = z^2$, on aura trouvé une décomposition du nombre n en une somme de deux carrés.

1.2. Programme et résultats

```
10 FOR n=1 TO 1000
20 x=SQR(n)
30 FOR y=1 TO x
40 c=n-y*y
50 d=SQR(c)
60 z=INT(d)
70 IF z*z=c AND z<>0 AND y<z THEN PRINT n
80 NEXT y
90 NEXT n
```

On obtient :

$145 = 1^2 + 12^2 = 8^2 + 19^2$

$185 = 4^2 + 13^2 = 8^2 + 11^2$

$260 = 2^2 + 16^2 = 8^2 + 14^2$

$265 = 3^2 + 16^2 = 11^2 + 12^2$... etc.

2. *Les nombres de Pythagore*

Cherchons les entiers x, y, z qui vérifient la relation $x^2 = y^2 + z^2$.

2.1. Méthode

Il suffit de modifier légèrement le programme précédent : on cherche les entiers n qui s'écrivent $n = y^2 + z^2$ et $n = x^2$.

2.2. Programmes et résultats

Limitons-nous à l'examen des entiers naturels compris entre 1 et 100 :

```
10 FOR x=1 TO 100
20 n=x*x
30 FOR y=1 TO x
40 e=0
50 c=n-y*y
60 d=SQR(c)
70 z=INT(d)
```

```
80 IF z=d AND z<>0 THEN PRINT x,y,z:e=1
90 IF e=1 THEN y=x
100 NEXT y
110 NEXT x
```

Ce programme donne 16 nombres.

En fait, il « oublie » des nombres tels que $x = 13$, $y = 12$ et $z = 5$.

Il s'agit du même phénomène que celui qui a été décrit chapitre 1, paragraphe 3. Modifions le programme en conséquence :

```
10 FOR x=1 TO 100
20 n=x*x
30 FOR y=1 TO x
40 e=0
50 c=n-y*y
60 IF c<>0 THEN GOSUB 130
70 z=INT(d)
80 IF z=d AND z<>0 THEN PRINT x,y,z:e=1
90 IF e=1 THEN y=x
100 NEXT y
110 NEXT x
120 END
130 d=SQR(c)
140 FOR i=1 TO 6
150 d=0.5*(d+c/d)
160 NEXT
170 RETURN
```

Nous obtenons cette fois 38 nombres vérifiant $x^2 = y^2 + z^2$.

3. Équations diophantiennes

Soient a, b et c des entiers relatifs non nuls. On appelle équation diophantienne du premier degré une équation de la forme $ax + by = c$, avec x et y entiers relatifs.

3.1. Méthode

Supposons que l'équation $ax + by = c$ possède une solution particulière (x_0, y_0) : il est facile de voir que toute autre solution (x, y) doit vérifier $x = x_0 + zb$ et $y = y_0 - za$, z étant un entier relatif quelconque.

On peut alors remarquer que b divise $c - ax_0$ puisque $y_0 = \dfrac{c - ax_0}{b}$.

Il existe alors un entier relatif k tel que $c - ax_0 = kb$. On en tire alors :

$$x_0 = \frac{c - kb}{a}.$$

Pour trouver x_0, il nous suffit donc de trouver un entier relatif k tel que a divise $c - kb$.

On peut également observer que l'équation $ax + by = c$ ne peut pas avoir de solution si le p.g.c.d. de a et de b ne divise pas c.

3.2. Programme et exemples

```
10 INPUT"a=";a
20 INPUT "b=";b
30 INPUT"c=";c
40 CLS
50 GOSUB 230
60 pgcd=e
70 q=ABS(c)/pgcd
80 IF q<>INT(q) THEN PRINT "pas de solution":END
90 REM Recherche d'une solution particuliere
100 a=a/pgcd:b=b/pgcd:c=c/pgcd
110 k=0
120 x=(c-k*b)/a
130 IF x<>SGN(x)*INT(ABS(x)) THEN k=k+1:GOTO 120
140 REM Affichage de la solution
150 s$="+":t$="+"
160 IF SGN(b)=-1 THEN s$="-"
170 IF SGN(a)=1 THEN t$="-"
180 PRINT "x=";x;s$;ABS(b);"*z"
190 PRINT "y=";(c-a*x)/b;t$;ABS(a);"*z"
200 END
210 REM Sous-programme de calcul du pgcd
220 REM                  des nombres a et b
230 f=MAX(ABS(a),ABS(b)):e=MIN(ABS(a),ABS(b))
240 r=f-e*INT(f/e)
250 IF r=0 THEN 280
260 f=e:e=r
270 GOTO 240
280 RETURN
```

Pour $a = 2$, $b = 3$, $c = 5$ on obtient $x = 1 + 3z$ et $y = 1 - 2z$.

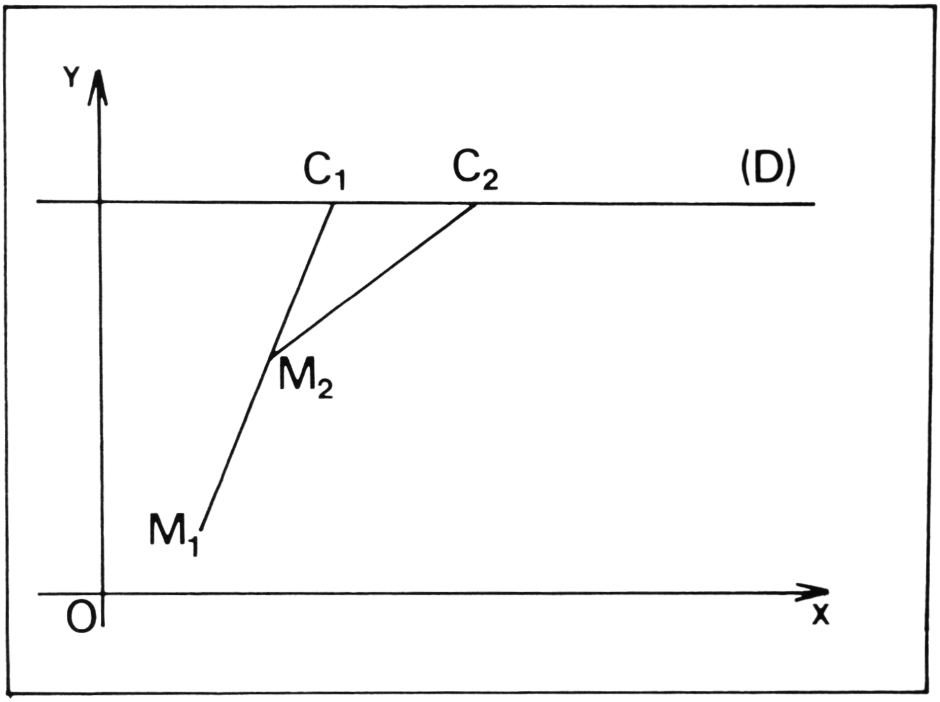
Pour $a = 129$, $b = 37$ et $c = 208$, on obtient $x = -9 + 37z$, $y = 37 - 129z$.

4. Recherche d'un minimum

Il s'agit là d'un exemple de discrétisation d'un problème : au lieu de rechercher le minimum d'une fonction f sur un intervalle I, on se contente de comparer des nombres $f(x_1)$, $f(x_2)$,..., $f(x_n)$.

1. Énoncé du problème

Une personne doit aller de M en N. Dans la région 1, elle avance à la vitesse constante de 2 km/h.



Dans la région 2, elle avance à la vitesse constante de 4 km/h.

Où placer le point I pour que le trajet ait la durée la plus courte possible?

4.2. Méthode et programme

Partageons $[A, B]$ en n parties égales et déterminons pour chacun des points de la subdivision obtenue la durée du trajet.

```
10 INPUT"n=";n
20 DIM x(n+1)
30 tmin=10000
40 FOR i=0 TO n
50 x(i)=2*i/n
60 d1=SQR(0.64+x(i)^2)
70 d2=SQR(1.44+(2-x(i))^2)
80 t=d1/2+d2/4
90 IF t<tmin THEN k=i
100 IF t<tmin THEN tmin=t
110 NEXT i
120 PRINT "AI=";x(k)
```

Le tableau suivant donne la valeur de AI obtenue pour diverses valeurs de n :

n	longueur de $[A, I]$ (en km)
20	0,4
40	0,35
100	0,36
500	0,352
1 000	0,354
2 000	0,353

On peut vérifier que 4 sin $\alpha = 2$ sin β : on a l'analogue d'une loi de réfraction en optique.

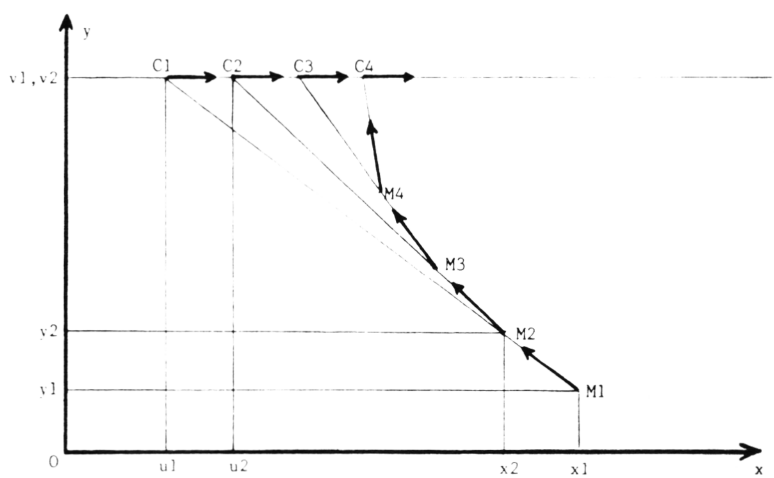
5. Courbes de poursuite

Un char parcourt une droite D avec une vitesse constante $\vec{v}_c$.

Au temps $t = 0$, on tire un missile dans la direction du char.

La vitesse $\vec{v}_M$ du missile est constante en module. On suppose également qu'à chaque instant le missile est « informé » de la position du char et peut modifier sa direction en conséquence. Quelle sera la trajectoire du missile?

5.1. Méthode



Pour obtenir l'équation de cette trajectoire, il faudrait résoudre un système différentiel. On peut se contenter d'une simulation. Choisissons un repère ainsi qu'il est indiqué sur la figure ci-dessous et considérons des instants successifs séparés par le temps Δt.

A l'instant t_1, le missile est en $M_1 = (x_1, y_1)$, le char en $C_1 = (u_1, v_1)$.

Pendant un temps Δt, le missile avance en ligne droite jusqu'en $M_2 = (x_2, y_2)$ et le char jusqu'en $C_2 = (u_2, v_2)$.

Arrivé en M_2, le missile est informé de la nouvelle position du char. Il se dirige alors vers C_2.
Le processus continue de la même manière jusqu'à la rencontre éventuelle.

5.2. Calculs

Soit d la distance de M_1 à C_1 et soit $\vec{k}$ le vecteur unitaire de la droite $(M_1\ C_1)$:

$$\vec{k} = \frac{\overrightarrow{M_1\ C_1}}{d}.$$

Nous désignerons par v_M et v_C les modules respectifs de $\vec{v}_M$ et $\vec{v}_C$.

On a $\overrightarrow{M_1M_2} = \vec{v}_M\,\Delta t$ et $\overrightarrow{C_1C_2} = \overrightarrow{v_C}\,\Delta t$.

On obtient successivement à partir de la première relation $\overrightarrow{M_1M_2} = v_M \Delta t\,\vec{k}$, $\overrightarrow{M_1M_2} = v_M \dfrac{\overrightarrow{M_1C_1}}{\mathrm{d}}\,\Delta t$.

Par projection sur les axes Ox et Oy, on obtient :

$$x_2 - x_1 = v_M\,\frac{u_1 - x_1}{d}\,\Delta t, \quad y_2 - y_1 = v_M\,\frac{v_1 - y_1}{d}\,\Delta t$$

et $\qquad u_2 = u_1 + v_C \Delta t, \qquad v_2 = v_1.$

5.3. Programme

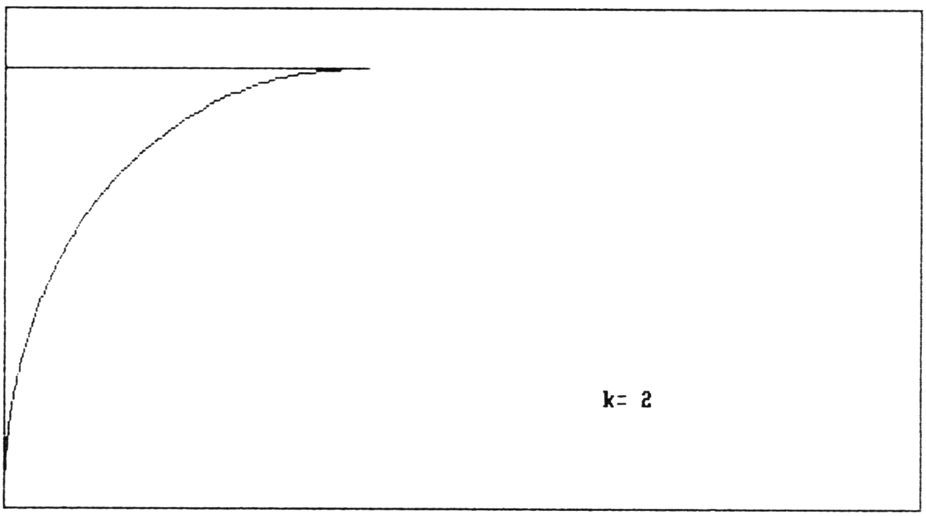
Prenons $\Delta t = 1$, $v_C = 2$ et posons $v_M = kv_C$.

Plaçons le missile en (0,0) et le char en (0,350) à l'instant initial. La rencontre est considérée comme faite pour $\mathrm{d} \leqslant 2$.

```
10 x1=0:y1=0
30 u1=0:v1=350
```

```
40 vc=2
50 INPUT"k=";k
60 vm=k*vc
70 CLS
80 d=SQR((x1-u1)↑2+(y1-v1)↑2)
90 x2=x1+vm*(u1-x1)/d
100 y2=y1+vm*(v1-y1)/d
110 u2=u1+vc
120 v2=v1
130 PLOT u1,v1:DRAW u2,v2
140 PLOT x1,y1:DRAW x2,y2
150 u1=u2:v1=v2
160 x1=x2:y1=y2
170 IF d>2 THEN 80
180 FOR n=15 TO 1 STEP-1
190 SOUND 1,426,40,n,,,1
200 NEXT
```

Pour $k = 2$, on obtient les trajectoires suivantes :



6. Les tablettes de chocolat

Il s'agit là d'un problème de probabilités très classique. On le traite par **simulation** grâce à l'existence de la **fonction RND.**

6.1. Problème

Quand on achète une tablette de chocolat X on trouve une image. S'il existe n images différentes également réparties dans les tablettes, quel nombre N de tablettes faudra-t-il acheter pour espérer avoir la collection complète des images?

6.2. Méthode

Les images sont numérotées de 1 à n. On tire au sort les numéros jusqu'à ce qu'ils soient tous sortis et on compte le nombre t des tirages nécessaires.

Chaque numéro x est accompagné d'un indicateur $c(x)$ qui vaut 0 si x n'est pas sorti et 1 si x est sorti.

L'arrêt des tirages intervient lorsque la somme des indicateurs est n.

6.3. Programme et résultats

```
10 INPUT"n=";n
20 DIM c(n)
30 FOR i=1 TO n
40 c(i)=0
50 NEXT i
60 x=1+INT(n*RND):c(x)=1
70 t=t+1
80 s=0
90 FOR j=1 TO n
100 s=s+c(j)
110 NEXT j
120 IF s<>n THEN 60
130 PRINT"il faut acheter ";t;" tablettes"
```

Le tableau suivant indique quelques valeurs de N en fonction de n :

n	5	8	15	20	30	100
N	12	21	47	72	126	532

Il s'agit, chaque fois, de moyennes faites sur des séries de 50 essais.

Indications bibliographiques

Les ouvrages cités sont précédés des signes *, ** ou *** selon leur degré de difficulté.

1. **Ouvrages de mathématiques générales**

**	J. Dixmier	Mathématiques du premier cycle Paris 1968 Gauthiers-Villars
***	J. Dieudonné	Éléments d'analyse Paris 1969 Gauthiers-Villars
**	J. Dieudonné	Calcul infinitésimal Paris 1968 Hermann
*	Gauthier/Royer/Thierce	Mathématiques (terminales C et E) Paris 1983 Hachette
*	M. Gourion	Mathématiques (terminales C et E) Paris 1983 Nathan

2. **Méthodes et algorithmes du calcul numérique**

**	Voiëvodine	Principes numériques d'algèbre linéaire Éditions Mir 1980
**	J. Baranger	Introduction à l'analyse numérique Paris 1977 Hermann
***	M. Sibony	Analyse numérique. Tome 1 : systèmes linéaires et non linéaires Tome 2 : Approximation et équations différentielles Paris 1982 Hermann
**	J. P. Nougier	Méthodes du calcul numérique Paris 1982 Masson
*	Vilenkine/Chilov	Quelques applications des mathématiques Éditions Mir 1975
**	H. Lehning/D. Jakubowicz	Mathématiques par l'informatique individuelle Paris 1982. Masson tome 1 : équations tome 2 : approximation, sommation
*	A. Engel	Mathématiques élémentaires d'un point de vue algorithmique. Paris 1979. Cédic/Nathan

3. **Statistiques et probabilités**

**	S. Aivazian	Étude statistique des dépendances Éditions Mir 1978
**	H. Ventzel	Théorie des probabilités Éditions Mir 1973
**	B. Vauquois	Probabilités Paris 1969 Hermann
**	E. Amzallag	Introduction à la statistique Paris 1978 Hermann

** E. B. Manouchian — Guide de statistiques appliquées Paris 1986 Hermann
* M. J. Moroney — Comprendre la statistique Paris 1970 Éditions Marabout-Université

4. **Graphisme mathématique**

* M. Rousselet — Graphismes en kits Paris 1984 E.T.S.F.
* M. Rousselet — Graphisme 3D Paris 1985 E.T.S.F.
** M. Lartigue/V. Gautheron — Systèmes différentiels : étude graphique. Paris 1983. Cédic/Nathan
** H. Lehning/D. Jakubowicz — Mathématiques par l'informatique individuelle
Paris 1984 Masson tome 4 : graphisme
* G. Grandpierre/R. Cotté — Mathématiques et graphismes Éditions du P.S.I Paris 1985.

Imprimé en France - JOUVE, 18, rue Saint-Denis, 75001 Paris
Dépôt légal : Septembre 1986. N° 16183. N° d'Editeur : 4535

CARTE POSTALE

Si vous désirez être informé régulièrement de nos différentes parutions, en vente chez votre libraire habituel, il vous suffit de mettre une croix dans la case correspondant à la rubrique qui vous intéresse (voir au verso) et de nous retourner cette carte après avoir inscrit très lisiblement - EN CAPITALES - votre nom et votre adresse.

Attention : si vous recevez déjà nos avis de parution, il est inutile de nous retourner ce formulaire. En cas de changement de coordonnées, prière de nous indiquer ancienne et nouvelle adresses.

Timbre

EDITIONS EYROLLES

61, boulevard Saint-Germain,

Service « Lecteurs »

75240 PARIS Cedex 05

POUR BÉNÉFICIER D'UN SERVICE GRATUIT D'INFORMATIONS

Société (le cas échéant)

NOM, PRENOM

PROFESSION

ADRESSE

CODE POSTAL VILLE

DISCIPLINES

- ☐ 326 INFORMATIQUE
- ☐ 327 MICRO-INFORMATIQUE
- ☐ 290 AUTOMATIQUE (y compris microprocesseurs)
- ☐ 111 ELECTRONIQUE

Les règles à calcul, les tables de racines carrées ou de logarithmes ont fait leur temps. Aujourd'hui l'**ordinateur** permet, en un temps raisonnable, de mener à bien les calculs les plus ardus.

Cet ouvrage met à la portée de tous de nombreux outils de calcul : ils pourront être utilisés au **lycée**, à l'**université**, dans l'**entreprise**.

Les principaux domaines du calcul numérique sont abordés : résolution des équations et des systèmes, intégration, statistiques et probabilités, calcul différentiel, . . . etc.

Chaque méthode est exposée simplement, sans formalisme inutile, avec un souci constant : **faire comprendre**.

couverture JÉRÔME LO MONACO

EYROLLES

8350

EYROLLES

CALCUL NUMERIQUE SUR AMSTRAD

M. ROUSSELET

Document numérisé avec amour par

AMSTRAD CPC

MÉMOIRE ÉCRITE

https://acpc.me/