Les algorithmes d'approximations

I. Problèmes d'optimisation

Exercice :

Une entreprise fabrique des boites d'allumettes ayant la forme d'un parallélépipède carrée de largeur 10 cm. Pour fabriquer cette forme à partir d'une surface du papier carrée, on découpe aux quatre coins quatre petits carrés de coté x. L'entreprise vous demande de déterminer la valeur x qui permet de consommer le moins du papier possible et en fabriquant des boites de volume maximum.

Questions :

Exprimer le volume de la boite en fonction de x

V = surface de base * hauteur

Surface de base = (10-2*x)*(10-2*x)*x

➔ $V = 4x^3 - 40x^2 + 100x$

- apparition d'une fonction numérique
- la fonction trouvée admet un maximum

$V'(x) = 12x^2 - 80x + 100$

$V'(x) = 0$ ➔ calculer delta

Delta = 80*80 – 4*12*100 = 1600

Racine (delta) = 40

$X_1 = (80 + 40)/(24) = 5$

$X_2 = (80 - 40)/24 = 10/6 = 1.666666666.....$

Donc la valeur (maximale) de x qui correspond à un volume maximum est égale à $X_2 = 10/6$

D'où le tableau de variation de V (x) :

X	0	10/6	5
V'(x)			
V (x)	↗	V max	↘

V max = V (10/6) = 16000/216 = 74.074....

V (0) = 0

V (5) = - 600

Ce type de problème est appelé problème d'optimisation : il s'agit de trouver la valeur de x qui donne un volume maximum de la boite.

On utilise une fonction valeur_optimal qui permet de déterminer la valeur maximale de x qui donne un volume maximum.

Le domaine de variation de x est de 0 à 5 (moitié de la largeur). On utilise alors une structure itérative à condition d'arrêt.

Pour cela, pour chaque itération :

On avance x d'un pas donné,

On calcul le volume

On le compare avec Vmax, si elle supérieure alors on échange la valeur du volume et on mémorise xmax.

Analyse du problème :

Résultat = Afficher la valeur de x

Données =pas (la valeur de variation de x)

Traitement =

    Ecrire ("x = ", valeur_optimal(pas))
    Répéter
    Pas = données
    Jusqu'à pas >=0.00001 et pas <=0.1

Algorithme principal

0) début volume_max
1) Ecrire ("pas ="), Lire (pas)
2) Ecrire ("x = ", valeur_optimal (pas))
3) Fin volum_max

Algorithme de la fonction valeur_optimal

0) **Fonction valeur_optimal (pas : réel) : réel**
1) Xmax ← 0, Vmax ← 0, X ← 0
2) Répéter
    X ← X + pas
    V ← 4*x*x*x – 40*x*x + 100
    Si Vmax < V Alors
        Xmax ← X
        Vmax ← V
    Fin Si
    Jusqu'à >5
3) Valeur_optimal ← Xmax
4) **Fin valeur_optimal**

```
program test;
uses wincrt;
var pas,x:real;
function calcul(pas:real):real;
var xmax,vmax,x,v:real;
begin
xmax:=0;vmax:=0;  x:=0;
repeat
x:=x+pas;
v:=4*x*x*x-40*x*x+100*x;
if vmax<v then
   begin
     xmax:=x;
     vmax:=v;   end;until x>5;
calcul:=xmax;
end;
begin
repeat
write('pas=');read(pas);until (pas>=0.00001) and (pas <=0.1);x:=calcul(pas);
write('valeur approx  de x = ',x:3:10);
end.
```

II. Problème d'approximations

Résolution de l'équation f(x) = 0

Soit f une fonction continue monotone sur l'intervalle [a, b]. Cette fonction ne s'annule qu'une seule fois dans [a, b]. On veut écrire un programme qui permet de chercher et d'afficher le zéro de cette fonction (f(x) = 0) avec une précision epsilon donnée.

On utilise la méthode de recherche par dichotomie :

- On divise l'intervalle [a, b] par 2
- Soit m le milieu de cet intervalle. Si f(m) et f(a) sont de même signe, le zéro recherché est dans [m, b], sinon il est dans [a, m].
- Répéter les étapes précédentes jusqu'à (b-a) devient inférieure ou égale à epsilon, dans ce cas, la valeur de m correspond à la valeur approchée de la solution de l'équation f(x)=0.

Exemple : $f(x) = 1/2 - x^2$, avec $x \in [0,1]$

On f(0) = 1/2, f(1) = -1/2 d'où : f(a).f(b) = f(0).f(1) <0

Donc on peut appliquer la méthode dichotomique sur [0, 1]

Pour ce la :

- Diviser [0, 1] par 2 ➔ m = (0+1)/2 = 1/2
- f(1/2) = 1/4
- f(m).f(0) = 1/2 * 1/4 >0 (sont de même signe)

➔ Le zéro est dans [m, b]= [1/2, 1], D'où :

Résultat = Afficher le zéro de f

Données = a, b, eps

Traitement =

Saisie (a,b,eps)

Écrire ("Le zéro de f est = ", **zéro (a, b, eps)**)

Analyse de la fonction zéro

Résultat = m

Traitement =

Zéro ← m

m ← (a+b)/2

Tant Que (b-a) >eps et f(m) <>0 **Faire**

Si f(a)*f(m)>0 **Alors**

a ← m

Sinon b ← m

Fin Si

m ← (a+b)/2

Fin Tant Que

Algorithme de la fonction zéro :

0) **Fonction zéro (a, b, eps : réel) : réel**
1) m ← (a+b)/2
2) Tant Que (b-a) >eps et f(m) <>0 Faire

   Si f(a)*f(m)>0 Alors

   a ← m

   Sinon b ← m, Fin Si

   m ← (a+b)/2

   Fin Tant Que
3) Zéro ← m
4) **Fin zéro**

Algorithme de la fonction f

0) **Fonction f (x : réel) : réel**
1) f ← 1/2 – x*x
2) **Fin f**