2
INFORMATIQUE

Tout petit programme pour rechercher un nombre dans un catalogue (*)

par Guy DESENFANT, Niort

I. Présentation

La fiche technique d'un produit X est classée selon le diamètre croissant (caractéristique principale du produit).

Les calculs techniques nous amènent à considérer un diamètre ϕ que nous appellerons ϕ_{th} (diamètre technique ou théorique).

Ce diamètre ϕ_{th} ne correspond pas en général à un diamètre du catalogue. Ce que nous savons, c'est qu'il existe ϕ_0 et ϕ_1, deux diamètres consécutifs du catalogue, tels que ϕ_{th} soit compris entre eux deux. Dans ce cas, le technicien choisit en général le diamètre du catalogue immédiatement supérieur : c'est-à-dire :

$$\text{si} \qquad \phi_0 < \phi_{th} \leqslant \phi_1$$

il choisira ϕ_1 que nous noterons ϕ_{st} (diamètre standard). Le problème consiste en l'automatisation de ce choix.

II. Mathématisation

Soit $X_1,\ldots,X_n$, n réels formant une suite strictement croissante.

En posant

$$X_0 = -\infty \quad \text{et} \quad \langle 1,n\rangle = [1,n] \cap \mathbb{N}$$

---

* La méthode décrite ci-dessous peut être utilisée pour rechercher un titre d'article dans un sommaire informatique.

nous obtenons

$$X \in]-\infty, X_n] \Rightarrow \exists ! i \in \langle 1,n \rangle : X_{i-1} < X \leq X_i$$

et nous pouvons définir la fonction C

$$C \left| \begin{array}{rcl}]-\infty, X_n] & \rightarrow & \langle 1,n \rangle \\ X & \rightarrow & C(X) = i \quad \text{tel que } X_{i-1} < X \leq X_i \end{array} \right.$$

Nous voulons trouver un algorithme tel que :

si $X > X_n$, il indique que X ne convient pas
si $X \leq X_n$, il indique $C(X)$

(il est intéressant de considérer X plus grand que X_n parce que dans la pratique, cela signifie que X est une valeur correspond à un dépassement des possibilités du produit).

Notons $I_i =]X_{i-1}, X_i]$ $i = 1$ à n
et $I_d =]X_n, +\infty[$

Nous avons pour tout x réel

soit $x \in I_{C(x)}$ $(x \leq X_n)$
soit $x \in I_d$ $(x > X_n)$

I_d sera dit zone de "dépassement de capacité".

A - n est une puissance de 2

III. Etude du cas $n = 4$

Nous avons 4 nombres réels formant une suite croissante x_1, x_2, x_3, x_4.
Soit x un autre nombre.

a) Si x est supérieur à x_4, alors x appartient à la zone "dépassement de capacité".

b) *Supposons maintenant que* x *est inférieur ou égal à* x_4

Comparons x *à* x_2 :

si x est plus petit que x_2, il faudra le comparer à x_1 et on pourra donner ensuite, selon le cas, la valeur 1 ou 2 à $C(x)$;

si x est plus grand que x_2, il faudra le comparer à x_3 et on pourra donner ensuite, selon le cas, la valeur 3 ou 4 à $C(x)$.

IV. Premier découpage du problème

Le problème peut se décomposer en 2 phases.

Phase 1 :

Nous comparons x à X_n :

— dans le cas où x est plus grand que X_n, le processus est terminé ; nous savons que la valeur x correspond à un dépassement de capacité ;

— dans le cas où x est inférieur ou égal à X_n, il faut faire la seconde phase.

Phase 2 :

Calcul de $C(x)$.

V. Principe de la phase 2

5.1.) *Exemple :* revoyons comment se décompose le cas où $n = 4$; supposons $x \leqslant x_4$;

x appartient au groupe de $\ell_2 (=4)$ intervalles d'élément central $\ell_1 (=2)$

si $x \leqslant x_2$, alors x appartient au groupe de $\ell_2 (=2)$ intervalles d'élément central $\ell_1 (=1)$; aller à (1)

sinon, x appartient au groupe de $\ell_2 (=2)$ intervalles d'élément central $\ell_1 (=3)$; aller à (2)

(1) si $x \leqslant x_1$, alors x appartient à I_1
sinon x appartient à I_2

(2) si $x \leqslant x_3$, alors x appartient à I_3
sinon x appartient à I_4

Reprenons le calcul de ℓ_1 et ℓ_2, en supposant que lorsque le travail est terminé, x doit appartenir à I_{ℓ_1}

$$\ell_1 = 2$$
$$\ell_2 = 4$$
$$\ell_2 = \ell_2/2 \ (=2)$$

Si $x \leqslant x_{\ell_1}$, alors $\ell_1 = \ell_1 - \ell_2/2 \ (=1)$; aller à (1)
sinon $\ell_1 = \ell_1 + \ell_2/2 \ (=3)$; aller à (1)

(1) $\ell_2 = \ell_2/2 \ (=1)$

Si $x \leqslant x_{\ell_1}$, alors fin
sinon $\ell_1 = \ell_1 + \ell_2$; fin.

5.2.) Le principe est fondé sur la dichotomie

Nous sommes en présence d'une suite de 2^p intervalles successifs.

Il existe 2 groupes d'intervalles successifs contenant chacun 2^{p-1} intervalles :

le premier est formé de tous les intervalles dont tous les éléments sont inférieurs ou égaux à $X_{2^{p-1}}$

le second est formé de tous les intervalles dont tous les éléments sont plus grands que $X_{2^{p-1}}$.

Présenté comme cela, nous voyons que nous sommes passés du problème d'ordre p à celui d'ordre p − 1, à une seule condition c'est que nous sachions repérer un groupe d'intervalles successifs ayant 2^k intervalles.

Appelons élément central, l'élément X_j qui partage le groupe d'intervalles successifs ayant 2^k intervalles en deux groupes d'intervalles successifs ayant 2^{k-1} intervalles.

Nous voyons que nous pouvons déterminer chaque groupe d'intervalles par son élément central (ou l'indice correspondant de son élément central : ℓ_1) et le nombre ℓ_2 d'intervalles.

5.3.) *Si ℓ_2 est une puissance de 2*

Supposons que nous sachions que x appartient à l'un des intervalles d'un groupe d'intervalles d'élément central x_{ℓ_1}.

Alors, en comparant x à x_{ℓ}, on sait déterminer dans quel groupe de $\ell_2/2$ intervalles se trouve x. De manière algorithmique, supposons ℓ_1 et ℓ_2 définis

$\ell_2 = \ell_2/2$

si $x \leqslant x_{\ell_1}$, alors $\ell_1 \leftarrow \ell_1 - \ell_2/2$
sinon $\ell_1 \leftarrow \ell_1 + \ell_2/2$

5.4.) *Premier algorithme de la phase 2*

$\ell_2 \leftarrow n$
$\ell_1 \leftarrow n/2$

tant que $\ell_2 > 2$, faire
début tq
$\ell_2 \leftarrow \ell_2/2$
si $x \leqslant x_{\ell_1}$ alors $\ell_1 \leftarrow \ell_1 - \ell_2/2$
sinon $\ell_1 \leftarrow \ell_1 + \ell_2/2$
fin tq

si $x \leqslant x_{\ell_1}$ alors (1)
sinon $\ell_1 \leftarrow \ell_1 + 1$, (1)

(1) écrire ℓ_1
fin.

5.5.) *Deuxième algorithme de la phase 2*

Dans l'algorithme précédent, nous avons dû exclure de la partie "tant que" le cas $\ell_2 = 2$, car on aurait obtenu :

si $x \leqslant x_{\ell_1}$ alors $\ell_1 \leftarrow \ell_1 - 0{,}5$
sinon $\ell_1 \leftarrow \ell_1 + 0{,}5$

dans les deux cas, nous obtenons une valeur de ℓ_1 inférieure de 0,5. Nous pouvons facilement remédier à cet inconvénient. D'où l'algorithme :

$\ell_2 \leftarrow n$
$\ell_1 \leftarrow n/2$
tant que $\ell_2 > 1$, faire
 début tq
 $\ell_2 \leftarrow \ell_2/2$
 si $x \leqslant x_{\ell_1}$ alors $\ell_1 \leftarrow \ell_1 - \ell_2/2$
 sinon $\ell_1 \leftarrow \ell_1 + \ell_2/2$
 fin tq
$\ell_1 \leftarrow \ell_1 + 0{,}5$
écrire ℓ_1
fin.

5.6.) *Troisième algorithme de la phase 2*

Une toute petite amélioration ; prenons $\ell_2' = \ell_2/2$. Nous obtenons alors :

$\ell_2' = \ell_1 \leftarrow n/2$
tant que $\ell_2' > 0{,}5$, faire
 début tq
 $\ell_2' \leftarrow \ell_2'/2$
 si $x \leqslant x_{\ell_1}$, alors $\ell_1 \leftarrow \ell_1 - \ell_2'$
 sinon $\ell_1 \leftarrow \ell_1 + \ell_2'$
 fin tq
$\ell_1 \leftarrow \ell_1 + 0{,}5$
écrire ℓ_1
fin.

VI. Algorithme dans le cas où n est une puissance de 2

Nous supposons les x_i déjà en mémoire sur un disque ou une bande ou autre support.

début
appeler le tableau X
entrer x

si $x \leqslant x_n$ alors (1)

sinon écrire "dépassement de capacité", aller à (2)

(1) $\ell_2 = \ell_1 \leftarrow n/2$

tant que $\ell_2 > 0{,}5$ | faire début tq

$\ell_2 \leftarrow \ell_2/2$

si $x \leqslant x_{\ell_1}$, alors $\ell_1 \leftarrow \ell_1 - \ell_2$

sinon $\ell_1 \leftarrow \ell_1 + \ell_2$

fin tq

$\ell_1 \leftarrow \ell_1 + 0{,}5$

écrire ℓ_1

(2) fin.

B - n est un nombre quelconque

VII. Calcul de n'

7.1.) Soit n' la puissance de 2 qui majore n

$$n' = 2^{p+1} \quad \text{avec} \quad 2^p < n \leqslant 2^{p+1}$$

d'où $p \operatorname{Log} 2 < \operatorname{Log} n \leqslant (p+1) \operatorname{Log} 2$

ou $p < \dfrac{\operatorname{Log} n}{\operatorname{Log} 2} \leqslant p + 1$

ce qui donne immédiatement

a) Si $\dfrac{\operatorname{Log} n}{\operatorname{Log} 2}$ est un entier, $n' = 2^{\frac{\operatorname{Log} n}{\operatorname{Log} 2}} = n$

b) Si $\dfrac{\operatorname{Log} n}{\operatorname{Log} 2}$ n'est pas entier, $n' = 2^{1 + E\left(\frac{\operatorname{Log} n}{\operatorname{Log} 2}\right)}$

où E est l'application partie entière.

7.2.) *Algorithme du calcul*

entrer n

$p \leftarrow \operatorname{Log} n/\operatorname{Log} 2$

si $E(p) \neq p$, alors $p \leftarrow E(p) + 1$

$n' \leftarrow 2^p$

fin.

VIII. Comment utiliser l'algorithme précédent

Seule la phase 2 est à modifier ; nous savons donc que $x \leqslant X_n$.

Posons $X_p = X_n + 1$ si $p \in \langle n+1, n' \rangle$;

alors $x \leqslant X_p \qquad \forall p \in \langle n+1, n' \rangle$

Ainsi nous créons des intervalles factices qui nous permettent d'utiliser l'algorithme précédent.

IX. Algorithme définitif

Début

appeler le tableau X et n

$p \leftarrow \text{Log } n / \text{Log } 2$

Si $E(p) \neq p$, alors $p \leftarrow E(p+1)$

sinon $n' \leftarrow 2^p$

Pour $p = n+1$ à n', faire $(x_p \leftarrow x_n + 1)$

entrer x

Si $x \leqslant x_n$, alors

- début
- $\ell_2 \leftarrow n'/2$
- $\ell_1 \leftarrow n'/2$
- tant que $\ell_2 > 0{,}5$ faire
  - début tq
  - $\ell_2 \leftarrow \ell_2/2$
  - Si $x \leqslant x_{\ell_1}$
  - alors $\ell_1 \leftarrow \ell_1 - \ell_2$
  - sinon $\ell_1 \leftarrow \ell_1 + \ell_2$
  - fin tq
- $\ell_1 \leftarrow \ell_1 + 0{,}5$
- écrire ℓ_1
- fin

sinon écrire "dépassement de capacité"

fin.