La division euclidienne implémentée avec une division flottante et une partie entière

Vincent Lefèvre

Loria / INRIA Lorraine

Journées Arinews

7–8 mars 2005

Introduction

Problème : calculer $\lfloor x/y \rfloor$ (*division euclidienne*), où $x \geqslant 0$ et $y > 0$.

En général, x et y sont de type entier (division entière). Ici, x et y sont des flottants $\rightarrow$ décomposition en deux opérations (/ et `floor`), avec un arrondi intermédiaire. Résultat final correct ?

Applications :

– Historiquement : implémentation en virgule flottante de mon algorithme de minoration de la distance entre un segment de droite et $\mathbb{Z}^2$.

– ECMAScript n'a pas de type entier, ni de division entière. `Math.floor(x/y)` est recommandé *sans justification*.

Le système à virgule flottante

Système en base 2, n bits de mantisse, division correctement arrondie dans le système cible (suivant l'un des modes d'arrondi, noté $\diamond$).

Par exemple : arithmétique IEEE 754 sans précision étendue intermédiaire (pour le moment).

Exposant d'un flottant $x \neq 0$: $e_x = \lfloor \log_2(|x|) \rfloor$, à une constante près.

$$x = \pm 1, b_2 b_3 b_4 \ldots b_n \times 2^{e_x}, \quad b_i \in \{0, 1\}.$$

Si les exposants sont bornés ($E_{\min} \leqslant e_x \leqslant E_{\max}$), on suppose que :

1. $E_{\max}$ est assez grand pour que x/y ne génère pas d'*overflow* ;
2. Le réel $1 - 2^{-n}$ est exactement représentable (dénormalisés OK).

$\lfloor x/y \rfloor$ représentable ?

On pose $k = \lfloor x/y \rfloor$ (sur les réels, sans arrondi).

Si $x/y > 2^n$, alors k n'est pas forcément représentable dans le système à virgule flottante, et le problème est tout autre.

Dans la suite, on suppose que :

$$x/y \leqslant 2^n.$$

Le problème est maintenant ($\diamond$ désigne le mode d'arrondi actif) :

Quelles sont les conditions sous lesquelles $\lfloor \diamond(x/y) \rfloor = k$?

En arrondi vers le bas

Pour $u \geqslant 0$, $\diamond(u)$ est le plus grand flottant $\leqslant u$.
Modes d'arrondi concernés : vers $-\infty$ et vers 0.

D'une part, $\diamond(x/y) \leqslant x/y$. D'autre part, $k \leqslant x/y$ et est exactement représentable. Donc $k \leqslant \diamond(x/y)$. Puisque $k+1 > x/y \geqslant \diamond(x/y)$, l'égalité $\lfloor \diamond(x/y) \rfloor = k$ est toujours vérifiée.

Sélectionner le mode d'arrondi vers le bas est une solution. Mais :

- Perte de performance possible si la sélection du mode d'arrondi est dynamique (ce qui est généralement le cas aujourd'hui).
- Certains langages (Java, ECMAScript, XPath 1.0) ne supportent que le mode d'arrondi au plus près.

En arrondi au plus près

$\diamond(u)$ est le flottant le plus proche de u.

Cas au milieu de deux flottants représentables consécutifs : le flottant choisi pour $\diamond(u)$ n'aura ici aucune importance (cas impossibles avec une division, sauf si le résultat est un dénormalisé).

Exemples pour lesquels $\lfloor \diamond(x/y) \rfloor \neq k$ avec $n \geqslant 2$:

$$x = 3 \times 2^{n-1} + 2 \quad \text{et} \quad y = 3.$$

Mais si x et y sont deux entiers sur n bits, on peut alors prouver que $\lfloor \diamond(x/y) \rfloor = \lfloor x/y \rfloor$.

Mieux...

Théorème (prouvé plus loin). Soit un système à virgule flottante en base 2, avec ou sans dénormalisés, tel que $1/2$ est représentable et normal. Si u est un réel, on note $\diamond(u)$ le nombre flottant le plus près de u (n'importe lequel s'il y en a deux). Soient x et y des flottants tels que $x \geqslant 0$, $y > 0$, et $x - y$ est exactement représentable. Supposons que x/y ne provoque pas d'*overflow*. Alors $\lfloor \diamond(x/y) \rfloor = \lfloor x/y \rfloor$, et cette valeur est exactement représentable.

Les conditions du théorème sont assez fortes.

Par exemple, si on demande juste que $x - y$ soit représentable sur $n + 1$ bits, alors on peut trouver des contre-exemples. Pour $n \geqslant 2$:

$$y = 2^n - 1 \quad \text{et} \quad x = 3y - 1 = 3 \times 2^n - 4.$$

En arrondi vers le haut

Pour $u \geqslant 0$, $\diamond(u)$ est le plus petit flottant $\geqslant u$.
Modes d'arrondi concernés : vers $+\infty$ et *away from* 0.

- Si $n \geqslant 4$, alors on a un exemple d'entiers x et y sur n bits pour lesquels $\lfloor \diamond(x/y) \rfloor \neq \lfloor x/y \rfloor$: $x = 3 \times 2^{n-2} + 2$ et $y = 3$.
- Possibilité de réutiliser les résultats de l'arrondi au plus près en précision $n - 1$.
- En particulier, si x et y sont deux entiers positifs $< 2^{n-1}$, alors $\lfloor \diamond(x/y) \rfloor = \lfloor x/y \rfloor$ dans n'importe quel mode d'arrondi.

Preuve — $\lfloor x/y \rfloor$ est représentable

Note : cas $n = 1$ trivial ($x = 0$, $\frac{1}{2}y$, y ou $2y$). Supposons $n \geqslant 2$.

Si $x = 0$, alors $\lfloor x/y \rfloor = 0$, qui est représentable. Sinon, $\mathrm{ulp}(x)$ désigne le poids du n-ième bit de la mantisse de x. Deux cas :

- Cas $y \geqslant \mathrm{ulp}(x)$: alors $x/y \leqslant x/\mathrm{ulp}(x) < 2^n$.
- Cas $y < \mathrm{ulp}(x)$: $x - y$ est représentable et son dernier bit non nul est $< \mathrm{ulp}(x)$, donc $e_{x-y} < e_x$.
  Puisque $n \geqslant 2$, $y < \mathrm{ulp}(x) \leqslant x/2^{n-1} \leqslant x/2$, et $x - y > x/2$.
  Donc $e_{x-y} = e_x - 1$.
  Seule possibilité : x puissance de 2 et $y = \mathrm{ulp}(x)/2$.
  Donc $\lfloor x/y \rfloor = 2^n$, qui est exactement représentable.

Preuve — cas $x \leqslant y$

– Si $x < y$, alors $\lfloor x/y \rfloor = 0$. Prouvons que $\diamond(x/y) < 1$.

Soit y^- le plus grand flottant $< y$. On a : $x/y \leqslant y^-/y$, donc $\diamond(x/y) \leqslant \diamond(y^-/y)$. Il suffit de faire la preuve pour $x = y^-$.

Multiplication de x et y par une puissance de $2 \rightarrow 1/2 < y \leqslant 1$.

Alors $y^- = y - 2^{-n}$, et $y^-/y = 1 - 2^{-n}/y \leqslant 1 - 2^{-n}$, qui est exactement représentable.

Par conséquent, $\diamond(y^-/y) \leqslant 1 - 2^{-n} < 1$, et $\lfloor \diamond(x/y) \rfloor = 0 = \lfloor x/y \rfloor$.

– Si $x = y$, alors $\diamond(x/y) = x/y = 1$, donc $\lfloor \diamond(x/y) \rfloor = \lfloor x/y \rfloor$.

Preuve — cas $x > y$

Si $\diamond(x/y) \leqslant x/y$, alors $\lfloor \diamond(x/y) \rfloor = k$ (cf arrondi vers le bas).
$\rightarrow$ On suppose que $\diamond(x/y) > x/y$.

On a $\lfloor \diamond(x/y) \rfloor \geqslant \lfloor x/y \rfloor$.

Il suffit de prouver que $\lfloor \diamond(x/y) \rfloor \leqslant \lfloor x/y \rfloor$, i.e. $\diamond(x/y) < k + 1$.

Noter que $\diamond(x/y)$ ne peut pas être un dénormalisé, car $x/y > 1$.
L'arrondi au plus près donne : $\diamond(x/y) - x/y \leqslant \mathrm{ulp}(x/y)/2$.

On définit $d = e_x - e_y \geqslant 0$.
Alors $x/y < 2^{d+1}$, et $\mathrm{ulp}(x/y) \leqslant \mathrm{ulp}(2^d) = 2^{d-n+1}$. Donc

$$\diamond(x/y) - x/y \leqslant 2^{d-n}.$$

Preuve — cas $x > y$ (suite)

Trouver un "bon" entier E tel que :

- on ait $x = X.2^E$, où X est un entier ;
- on ait $y = Y.2^E$, où Y est un entier ;

et on aura alors : $X/Y = x/y$.

Si y ou $x - y$ a le même exposant que x, on prendra $E = \log_2(\mathrm{ulp}(x))$, sinon $E = \log_2(\mathrm{ulp}(x)) - 1$.

(Explications sur ce choix plus loin, dans chaque sous-cas.)

Preuve — cas $x > y$, $E = \log_2(\mathrm{ulp}(x))$

Choix de E : X est entier. Concernant Y :

- Si $e_y = e_x$, alors $\mathrm{ulp}(y) = \mathrm{ulp}(x)$ et Y est entier.
- Sinon, $X - Y$ est entier. Donc $X - (X - Y) = Y$ aussi.

D'une part, $X < 2^n$, donc $Y < 2^{n-d}$.

D'autre part, $X/Y = x/y < k + 1$, donc $(k + 1)Y - X > 0$, i.e. $(k + 1)Y - X \geqslant 1$.

D'où $\diamond(x/y) - x/y \leqslant 2^{d-n} < 1/Y \leqslant k + 1 - x/y$, i.e.

$$\diamond(x/y) < k + 1.$$

Preuve — cas $x > y$, $E = \log_2(\mathrm{ulp}(x)) - 1$

- Le choix d'un E plus petit est nécessaire dans certains cas.
- Ce choix est suffisant. D'abord, X est entier.
  Ensuite, y ou $x - y$ a un exposant $\geqslant e_x - 1$ (donc $= e_x - 1$).
  Donc Y ou $X - Y$ est entier. $\rightarrow$ Y est entier.

Ce qui change : $X < 2^{n+1}$, donc $Y < 2^{n+1-d}$, et $\diamond(x/y) - x/y < 2/Y$.
La borne $2/Y$ est 2 fois plus grande que dans le premier cas.

Exemple où $\diamond(x/y) - x/y > 1/Y$: $n = 5$, $x = X = 44$, $y = Y = 19$.
Mais x/y n'est pas suffisamment proche d'un entier. Donc, malgré l'erreur importante, pas de problème d'arrondi : $\diamond(44/19) < 3$.

On considère l'entier positif $r = (k+1)Y - X$.

– Si $r \geqslant 2$, on termine la preuve comme dans le premier cas.

– Si $r = 1$, on remarque que sur des exemples généraux, $x/y < 2^d$.
On gagne le facteur 2 ici : $\mathrm{ulp}(x/y) \leqslant \mathrm{ulp}(2^{d-1}) = 2^{d-n}$, et
$\diamond(x/y) - x/y \leqslant \mathrm{ulp}(x/y)/2 \leqslant 2^{d-n-1} < 1/Y = k + 1 - x/y$.

– Sous-cas restant : $r = 1$ et $x/y \geqslant 2^d$.
On a : $2^d \leqslant x/y < k+1$, donc $2^d \leqslant k$.
Alors $2^d Y \leqslant kY = X - Y + 1 < 2^n + 1$, i.e. $2^d Y \leqslant 2^n$.
Mais par définition de d, $2^n \leqslant 2^d Y$. Donc $Y = 2^{n-d}$.
X est pair et r impair, donc $Y = 1$ et $X = 2^n$.
$X/Y = 2^n$, qui est représentable, d'où $\lfloor \diamond(x/y) \rfloor = \lfloor x/y \rfloor$.

$\rightarrow$ Tous les cas ont été prouvés.

L'effet de la précision étendue intermédiaire

Problème : possible double arrondi avant le `floor`, dont l'effet peut être visible en arrondi au plus près.

Exemple : $n = 5$, $p = 8$, $x = 44$ et $y = 15$, où p est la taille de mantisse en précision étendue. $44/15$ en binaire : 10.11101110111..., si bien que $\diamond_p(44/15) = 10.111100_2$ et $\diamond_n(\diamond_p(44/15)) = 11.000_2 = 3$, avec la règle de l'arrondi pair, au lieu de 2.

Pour $n \geqslant 3$, soit $y = 2^{n-1} - 1$ et $x = 3y - 1$. Alors $x/y = 3 - 1/y$, avec $2^{1-n} < 1/y < 2^{1-n} + 2^{3-2n}$. Si $p \leqslant 2n - 2$, alors $\diamond_p(x/y) = 3 - 2^{1-n}$ et $\diamond_n(\diamond_p(x/y)) = 3$, alors que $\lfloor x/y \rfloor = 2$.

$\longrightarrow$ Problème possible en C sous Linux/x86 pour `floor(x/y)` avec $x = 3 \times 2^{52} - 4$ et $y = 2^{52} - 1$ de type `double`.

Théorème pour la précision étendue :

- Deux entiers $n \geqslant 3$ et $p \geqslant 2n-1$.
- Système flottant en base 2, en précision interne p.
- Avec ou sans dénormalisés, mais $1/2$ doit être normal.
- $\diamond_n$ et $\diamond_p$: arrondi au plus près en précisions respectives n et p (cas au milieu : le choix de l'arrondi n'a aucune importance).
- Deux flottants x et y en précision n, tels que $x \geqslant 0$, $y > 0$ et $x - y$ est exactement représentable en précision n.
- Pas d'*overflow* lors du calcul de x/y.

Alors $\lfloor \diamond_n(\diamond_p(x/y)) \rfloor = \lfloor \diamond_n(x/y) \rfloor = \lfloor x/y \rfloor$, et cette valeur est exactement représentable en précision n.

Conclusion

Quelques suppositions sur le système flottant pour que `floor(x/y)` donne $\lfloor x/y \rfloor$ comme résultat en précision n :

- En arrondi au plus près, s'il y a une précision étendue p, alors on impose $p \geqslant 2n - 1$. Ne convient pas à la précision étendue x86, mais OK pour la quadruple ($p = 113$) avec la double ($n = 53$). Certains langages (Java, etc.) interdisent la précision étendue.
- La division doit être correctement arrondie (IEEE 754 OK).
- La plage d'exposants doit être assez grande (IEEE 754 OK).

Modes d'arrondi convenables :

- Au plus près : OK dans des conditions raisonnables connues.
- Vers le bas : toujours OK.