

Division entière euclidienne vs non-euclidienne

Il existe 2 sortes de division entière :

- La division entière **euclidienne**
 - correspond à une **troncature vers zéro**
 - revient à conserver la partie entière du résultat
 - Exemples :
 - $7 / 4 =$ partie entière de $1.75 = 1$
 - $-7 / 4 =$ partie entière de $-1.75 = -1$
- La division entière **non-euclidienne**
 - correspond à une **troncature vers $-\infty$**
 - Exemples :
 - $7 / 4 = 1$
 - $-7 / 4 = -2$

Ces deux divisions (voir exemples ci-dessus)

- donnent le même résultat si les 2 opérandes sont de même signe
- donnent des résultats différents si les 2 opérandes sont de signe opposé

Dans le cas d'une expression (p.ex. $n = 7 / 4$), c'est toujours la division entière euclidienne qui est mise en oeuvre (quel que soit le compilateur C utilisé).

Question : Qu'en est-il dans le cas d'un décalage à droite $n \gg m$?

Rappel :

- Si les bits insérés à gauche valent 0, on parle de **décalage à droite logique**
- Si les bits insérés à gauche correspondent au bit de signe de n , on parle de **décalage à droite arithmétique**
- Attention ! La nature du décalage à droite (logique ou arithmétique) peut varier d'un compilateur à l'autre

Réponse : Dans le cas d'un décalage à droite, c'est toujours la division entière **non-euclidienne** qui est mise en oeuvre, sauf dans le cas où n est signé, $n < 0$ et que l'on a affaire à un décalage à droite logique.

Exemple :

- $-1 \gg 1 = -1$, si décalage à droite arithmétique
 - correspond bien à la division entière non-euclidienne : $-1 / 2 = -1$
- $-1 \gg 1 = \text{INT_MAX}$, si décalage à droite logique
 - pas équivalent à la division entière non-euclidienne : $-1 / 2 = -1$