

1 Algorithme de division

Le but de cet algorithme c'est de pouvoir effectuer la division tout en échappant au multiples cas d'exceptions ou réduire d'énorme écart d'erreur qu'il peut y avoir au cours de l'opération. Pour ce faire nous voudrions stocké les bits dans un Tableau ou nous pourrions accéder bit par bit. Il s'agira de faire une soustraction successive du diviseur de la dividende . Le Q format ici est le nombre de bits fractionnaire.

Algorithme :

Variable : numérateur [nombre de bits] : Entier,
Resultat[taille numérateur + dénominateur] : Entier;

Variable : dénominateur[nomre de bits] : Entier, a : reel, b : reel,

$Q_{numérateur}, Q_{dénominateur};$

Debut :

$a \leftarrow a * 2^{(Q_{numérateur}+Q_{dénominateur})};$

$b \leftarrow b * 2^{(Q_{numérateur}+Q_{dénominateur})};$

Si (ab<b) Alors

Tant-que (a<b) Faire

$a \leftarrow a * 2$

Resultat[i] = 0;

i++;

FinTantQue

SiNon

TantQue a-b > 0 Alors

Resulta[i] = 1 ;

FinTantQue

FinSi

Si Resultat[0] = 0;

Pour i \in [0, *tailleBit*];

```

        Resultat = 0. Resulta[i];
SiNon
    Pour i ∈ [tailleBit, 0];
        Resultat = Resultat[i];
FinSi
Lire(resultat);
Fin

```