L’histoire de 0x5f3759df

Clément Villain

E.I.S.T.I.

December 2, 2015

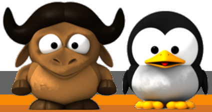
Fast inverse square root

```
float Q_rsqrt( float number )
{
    long i;
    float x2, y;
    const float threehalfs = 1.5F;

    x2 = number * 0.5F;
    y  = number;
    i  = * ( long * ) &y;                       // evil floating point bit level hacking
    i  = 0x5f3759df - ( i >> 1 );               // what the fuck?
    y  = * ( float * ) &i;
    y  = y * ( threehalfs - ( x2 * y * y ) );   // 1st iteration
//      y  = y * ( threehalfs - ( x2 * y * y ) );   // 2nd iteration, this can be removed

    return y;
}
```

Figure: Code source de *Quake III Arena*



Prérequis: Représentation des nombres

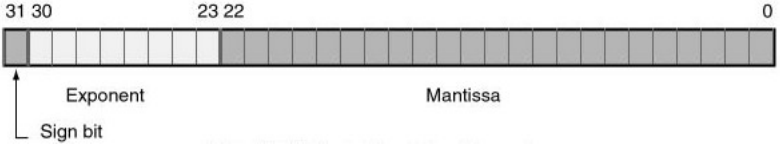


Figure: Réprésentation IEEE 754: float

- Valeur d'un float $x = 2^e(1 + m)$ avec $e = E - B$ et $B = 127$
- Réprésentation entière: $X = M + LE$ avec $m = \frac{M}{L}$ et $L = 2^{23}$

Calcul de $\frac{1}{\sqrt{x}}$

1. Passage au $\log_2$:
$$y = \tfrac{1}{\sqrt{x}} \text{ donc } \log_2(y) = -\tfrac{1}{2}\log_2(x)$$
2. Réprésentation flottante:
$$\log_2(1+m_y) + e_y = -\frac{1}{2}(\log_2(1+m_x) + e_x)$$
3. DL du log_2:
$$m_y + \sigma + e_y \simeq -\frac{1}{2}(m_x + \sigma + e_x)$$
4. Réprésentation entière:
$$m = \tfrac{M}{L} \text{ et } e = E - B$$
5. Finalement:
$$I_y \simeq \frac{3}{2}L(B-\sigma) - \frac{1}{2}I_x$$

Comment choisir σ

Pour $\sigma = 0.0450465$

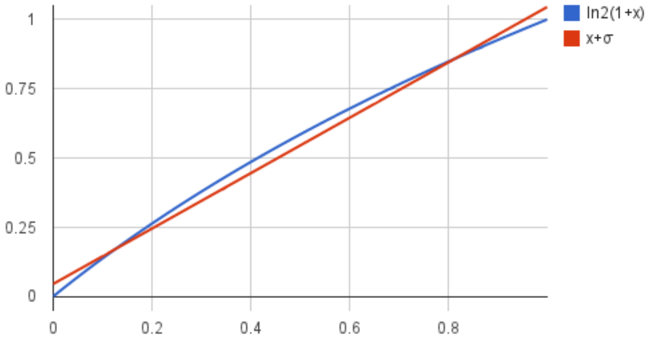
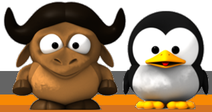


Figure: Approximation de $\log_2(1+x)$



Notre nombre magique

Rappel:

$$I_y \simeq \frac{3}{2} L(B - \sigma) - \frac{1}{2} I_x$$

$\frac{3}{2} 2^{23}(127 - 0.0450465) = 0x5f3759df$

```
i  = 0x5f3759df - ( i >> 1 );
```

Figure: Application de la formule

Performance

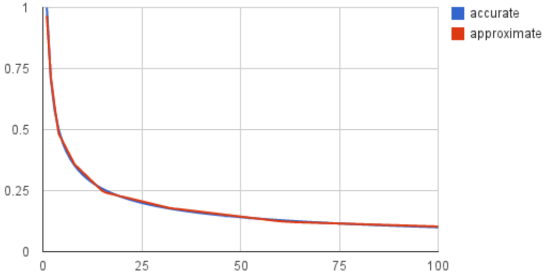
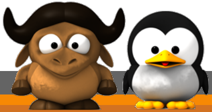


Figure: Qualité de l'approximation

Efficace, rapide et généralisable



C'est fini !

En espérant vous voir nombreux pour les talks fin janvier !

<?="Atilla";?>