IFT209 –Programmation système

Laboratoire 6

Enseignant: Rosa Garcia
Date de remise: indiquée sur **Turnin**
Modalités de remise: **Turnin**
À réaliser: personne seule ou en équipe de deux

Le but de ce laboratoire est de pratiquer la programmation avec les nombres à virgule flottante.

Problème. Un logarithme en base **b** d'un nombre **n** noté $x = \log_b(n)$ est l'exposant auquel on doit élever la base b pour obtenir le nombre n, impliquant donc cette relation inverse: $n = b^x$.

On peut calculer la partie entière d'un logarithme en comptant le nombre de fois qu'on peut successivement diviser le nombre n par la base b avant d'obtenir un quotient inférieur à b.

Exemple 1. $\log_{16}(2048)$:
2048 ÷ 16 = 128 (divisible 1 fois)
128 ÷ 16 = 8 (divisible 2 fois)
8 < 16, fin.
donc, $\log_{16}(2048) = 2 + \log_{16}(8)$
La partie entière est donc 2.

On peut calculer une approximation de la partie fractionnaire du logarithme comme une sommation de fractions de puissances de la forme $1/2^n$. Si une fraction de puissance de b est inférieure à notre quotient actuel, alors elle est présente dans le logarithme et on peut l'ajouter au total. On arrête si le quotient actuel est inférieur à 1 plus la précision désirée.

Exemple 2. $\log_{16}(2048) = 2 + \log_{16}(8)$
$\log_{16}(8)$ (avec précision de 0,01):
$8 >= 16^{1/2}$, donc 8 ÷ 4 = 2 (divisible par $16^{1/2}$, on ajoute ½ au total)
$2 >= 16^{1/4}$, donc 2 ÷ 2 = 1 (divisible par $16^{1/4}$, on ajoute ¼ au total)
1 < 1 + précision (1.01), on peut donc arrêter.
$\log_{16}(8)$ = ½ + ¼ = ¾ =0.75
$\log_{16}(2048)$ = 2,75

Vous devez écrire un programme, en langage d'assemblage `ARMv8`, qui calcule et affiche le logarithme en base b d'un nombre n, la base et le nombre étant lus au clavier. Une partie du code pour la lecture et l'affichage est déjà préparé, vous devez le compléter. Vous devez également écrire votre algorithme de calcul du logarithme. Un sous-programme *`Logarithme`* est prévu à cet effet. Le niveau de précision de votre algorithme doit être fixé à `0.00001` (donc cas d'arrêt à `1.00001`).

Tests. Par exemple, dans un terminal, vous devriez obtenir, pour un calcul de $\log_b(n)$:

Entrées (n,b)	Sortie
32 4	2.5
2048 16	2.75
100 5	2.86135
1000 7	3.54988
375 3.5	4.73108

Rappels.

- La fonction `scanf` lit des nombres en simple précision (format `%f`);
- La fonction `printf` affiche des nombres en double précision (format `%f`);
- Il existe une instruction `fsqrt` qui calcule la racine carrée d'un nombre;
- La commande `info float` dans gdb vous permet de voir l'état des registres virgule flottante.

Directives.

- Votre programme doit être obtenu en complétant le code fourni sur **Turnin**;
- Votre programme doit être remis dans un seul fichier nommé `labo6.as`;
- Ne modifiez pas le point d'entrée ainsi que le format des entrées et sorties;
- Supposez que les entrées seront valides, entre autres, que n sera positif et que b sera entier.

Pointage. Vous pouvez obtenir jusqu'à 10 points répartis ainsi:

- 5 points si votre programme passe les cinq tests ci-dessus;
- 3 points si votre programme affiche la bonne sortie sur d'autres tests choisis à la correction
- 2 points pour la lisibilité du code (indentation, commentaires, conventions).

Voici un exemple complet. Utilisez ses valeurs intermédiaires pour déverminer votre programme.

Exemple 3.

$\log_5(200)$: 200 ÷ 5 = 40 (divisible 1 fois)
40 ÷ 5 = 8 (divisible 2 fois)
8 ÷ 5 = 1,6 (divisible 3 fois)
1,6 < 5: fin.
donc, $\log_5(200) = 3 + \log_5(1,6)$

$\log_5(1,6)$ (avec précision de 0,01):
1,6 < $5^{1/2}$ (2,2361) donc on n'ajoute pas ½ au total.
1,6 >= $5^{1/4}$ (1,4953) donc 1,6 ÷ 1,4953 = 1.0700, on ajoute **1/4** au total.
1,07 < $5^{1/8}$ (1,2228), donc on n'ajoute pas 1/8 au total.
1,07 < $5^{1/16}$ (1,1058), donc on n'ajoute pas 1/16 au total.
1,07 >= $5^{1/32}$ (1,0516), donc 1.07 ÷ 1.0516 = 1.0175, on ajoute **1/32** au total.
1.0175 < $5^{1/64}$ (1,0255), donc on n'ajoute pas 1/64 au total.
1.0175 >= $5^{1/128}$ (1.0127), donc 1.0175 ÷ 1.0127 = 1.0047, on ajoute **1/128** au total.
1.0047 < 1.01 (précision), fin.

$\log_5(1,6) = 1/4 + 1/32 + 1/128 = 0.25 + 0.03125 + 0.0078125 = 0.2890625$
log5(200) = 3,2890625