Licence de Sciences et Technologies — Université de Perpignan Via Domitia
S6 – mention Informatique, Mathématiques — Ph. LANGLOIS

Algorithmes et calcul scientifique

Première session 2019

Durée : 2 heures. Aucun document ni machine autorisé.

Barème indicatif sur 30 : Ex. 1 : 24 (5+5+5+3+6) ; Ex. 2 : 5 (1+1+3) ; qualité de la rédaction = ± 1.
Les questions plus difficiles sont signalées avec une ou deux ($\star$).

Exercice 1.
Soient les nombres réels $a = 2^{23} + 17$ et $b = 2^{-1} - 2^{-31}$. On note c le nombre réel $a + b$.

1. Rappels du cours et applications simples.
   (a) Rappeler de façon synthétique les caractéristiques principales des formats de représentation `binary32` et `binary64` de la norme IEEE-754.
   (b) Comment se comparent les ensembles des flottants `binary32` et `binary64` ? Justifier votre réponse.
   (c) Rappeler les notions suivantes : arrondi correct, mode d'arrondi au plus près, stratégie de l'arrondi pair.
   (d) a et b sont-ils des nombres flottants `binary64` ? Justifier votre réponse.
   (e) a et b sont-ils des nombres flottants `binary32` ? Justifier votre réponse.
2. Dans cette question, on considère **uniquement le format `binary64` en mode d'arrondi au plus près avec stratégie de l'arrondi pair** .
   (a) Justifier que $c = a + b$ n'est pas un flottant `binary64`.
   (b) Expliciter $c64-$ et $c64+$, les deux flottants `binary64` consécutifs qui encadrent c.
   (c) En déduire la valeur du milieu $c64m$ de $[c64-, c64+]$.
   (d) En déduire la valeur de $\widehat{c64}$, arrondi du nombre réel c (en `binary64` pour le mode d'arrondi au plus près avec stratégie de l'arrondi pair) ? Justifier votre réponse.
   (e) ($\star$) Est-il possible de calculer un tel $\widehat{c64}$ à partir de a et b dans le cadre de la norme IEEE-754, version 1985 ? Justifier votre réponse.
3. Dans cette question, on considère **uniquement le format `binary32` en mode d'arrondi au plus près avec stratégie de l'arrondi pair** .
   (a) Justifier que $c = a + b$ n'est pas un flottant `binary32`.
   (b) Expliciter $c32-$ et $c32+$, les deux flottants `binary32` consécutifs qui encadrent c.
   (c) En déduire la valeur du milieu $c32m$ de $[c32-, c32+]$.
   (d) En déduire la valeur de $\widehat{c32}$, arrondi du nombre réel c (en `binary32` pour le mode d'arrondi au plus près avec stratégie de l'arrondi pair) ? Justifier votre réponse.
   (e) ($\star$) Est-il possible de calculer un tel $\widehat{c32}$ à partir de a et b dans le cadre de la norme IEEE-754, version 1985 ?
4. Quelle est la valeur $\widehat{c64\_32}$ de l'arrondi de $\widehat{c64}$ en `binary32` pour le mode d'arrondi au plus près avec stratégie de l'arrondi pair ? Justifier votre réponse.

5. ($\star$) On considère toujours le mode d'arrondi au plus près avec stratégie de l'arrondi pair. On imagine l'existence d'un opérateur d'addition qui calcule l'arrondi correct en `binary32` de la somme de deux opérandes `binary64`.
   (a) Proposer un algorithme simple qui réalise un tel traitement *dans la très grande majorité des cas*. Dans un premier temps, on oubliera les dépassements de capacités (*overflow* et *underflow*). Les principes de cet algorithme seront d'abord décrits en français, puis une version algorithmique ou en langage C complètera cette description.
   (b) Déduire des questions 2, 3 et 4, quelle est la difficulté majeure pour que cet algorithme soit correct pour toute paire d'opérandes `binary64`. On ne demande pas comment résoudre cette difficulté mais de la décrire précisément.
   (c) ($\star\star$) Décrire les principes des traitements dans les deux cas de dépassement de capacité. On distiguera les cas d'utilisation ou non des flottants `binary32` dénormalisés après avoir rappelé les traitements associés.

Exercice 2.

1. Donner un exemple de *catastrophic cancellation* en `binary32`.
2. Que peut-on dire du problème, de l'algorithme, de la solution calculée (en `binary32`) ?
3. Proposer des solutions pour limiter les effets de tels cas.