

# Conception Formelle

## TD3: Une boucle, mais on ne donne plus les invariants

Vincent Penelle

2019-2020

### Copie de VOTRE NOM

Note : Comment rédiger ?

Regardez le fichier source fourni, et notamment l'exercice 1 du TD1 qui est rédigé. Vous y trouverez toutes les commandes latex dont vous devriez avoir besoin (c'est-à-dire essentiellement WP et `code` pour la syntaxe C, ainsi que l'environnement `align*`), ainsi qu'un exemple de comment rédiger (qui est déjà dans le cours). Rédigez vos réponses dans les cadres prévus à cet effet (ceux où il y a écrit "METTEZ VOTRE RÉPONSE ICI"). Et n'oubliez pas de renseigner "VOTRE NOM" dans le bloc ci-dessus.

Si vous ne pouvez pas compiler du latex, rendez-moi simplement le fichier `.tex`, et je ne râlerais pas si j'ai à faire quelques modifs pour le compiler et que la présentation est bizarre. Sinon, rendez-moi le pdf, et essayez de rendre ça "pas horriblement moche".

Pour l'environnement `align*` : l'utiliser est assez simple.

- Il est par défaut en mathmode (donc pas besoin de `$`, mais si vous voulez du texte, il faut utiliser la commande `\text`).
- `&` permet de couper l'endroit où aligner (je le met toujours avant le `=` dans mes exemples).
- `\\` termine une ligne. Ne l'oubliez pas, sinon vous aurez des résultats surprenants. On se contentera d'un alignement (un `&`) par ligne affichée.

#### Exercice 1 : Somme de tableau peu orthodoxe

On considère la fonction suivante :

```
1 /*@ ensures Psi: \result == SUM(t,0,n);
2 */
3 int StupidSumTab(int *t, int n)
4 {
5     int low = 0;
6     int high = n - 1;
7     int res = 0;
8     /*@ loop invariant I1:
9         loop invariant I2:
10        loop variant V:
11    */
12    while (low <= high)
13    {
14        if (t[low] < t[high])
15        {
```

```

16         res = res + t[low];
17         low++;
18     }
19     else
20     {
21         res = res + t[high];
22         high--;
23     }
24 }
25 return res;
26 }

```

Cette fonction réalise la somme d'un tableau, avec un parcours dépendant du contenu du tableau. Dans les annotations,  $SUM(t, i, n)$  est un raccourci pour la somme des cases du tableau entre  $i$  et  $n - 1$ . Par exemple  $SUM(t, 2, 5) = t[2] + t[3] + t[4]$ . Vous pouvez voir cela comme une macro. Il est ceci dit tout à fait possible de définir de telles fonctions logiques en frama-c (via des axiomes décrivant le comportement de la fonction, on verra ça plus tard).

Le but de l'exercice est évidemment de déterminer la weakest liberal precondition de la fonction, de montrer qu'elle est bien définie, et de prouver qu'elle termine, mais il va, pour cela, falloir déterminer des invariants de boucles.

La première étape est de déterminer quelle formule doit être vraie en fin de boucle.

- (a) Calculer  $WLP(25, \psi)$ .

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.

On va maintenant déterminer des invariants de boucle.

- (b) Déterminez deux invariants de boucle qui doivent permettre de démontrer que  $WLP(25, \psi)$  est vrai en fin de boucle. Vous devez avoir un invariant de boucle qui permet de prouver qu'en fin de boucle,  $low - 1$  et  $high$  sont égaux (mettons  $I_1$ ). L'autre invariant ( $I_2$ ) doit déterminer ce que vaut  $res$ , et notamment, en fin de boucle (c'est à dire quand  $low > high$  et  $I_1$  sont vrais) impliquer  $WLP(25, \psi)$ . Bien évidemment, vous pouvez déterminer si vos solutions sont correctes en réussissant les 3 questions d'après.

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.

- (c) Démontrez que  $\neg(low \leq high) \wedge I_1 \wedge I_2 \Rightarrow WLP(25, \psi)$ .

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.

- (d) Démontrez que  $low \leq high \wedge I_1 \wedge I_2 \Rightarrow WLP(13 - 24, I_1)$ .

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.

- (e) Démontrez que  $low \leq high \wedge I_1 \wedge I_2 \Rightarrow \text{WLP}(13 - 24, I_2)$ .

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.

- (f) Déduisez-en que  $\text{WLP}(\text{StupidSumTab}, \psi)$  est bien défini et donnez un triplet de HOARE valide pour cette fonction.

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.

On va maintenant démontrer que cette fonction termine.

- (g) Déterminer une valeur pour le variant  $V$ . La justification de cette valeur sera la réussite des deux questions suivantes.

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.

- (h) Démontrez que  $low \leq high \wedge I_1 \wedge I_2 \Rightarrow V \geq 0$ .

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.

- (i) Démontrez que  $low \leq high \wedge I_1 \wedge I_2 \wedge I_3 \Rightarrow \text{WLP}(13 - 24, V < \text{at}(V, 13))$ .

**Réponse:**

METTEZ VOTRE RÉPONSE ICI.