Conception Formelle
TD1 : Weakest Precondition simple

Vincent Penelle

2022-2023

Copie de VOTRE NOM

Note : Comment rédiger ?

Regardez le fichier source fourni, et notamment l'exercice 1 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, {φ}P{ψ} pour les triplets, 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 horiblement 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 : Un exemple rédigé : swap

On considère la fonction suivante :

```
1 /*@ ensures Psi: *a == \old(*b) && *b == \old(*a)
2 */
3 void swap(int* a, int* b){
4 int aux = *a;
5 *a = *b;
6 *b = aux;
7             }
```

(a) Calculer WP(swap, ψ), avec $\psi = *a == \text{old}(*b) \wedge *b == \text{old}(*a)$.

Réponse :

$$\begin{aligned}
\mathsf{WP}(\texttt{swap}, \psi) &= \mathsf{WP}(4, \mathsf{WP}(5, \mathsf{WP}(6, \psi))) \\
&= \mathsf{WP}(4, \mathsf{WP}(5, *a == \mathsf{old}(*b) \wedge aux == \mathsf{old}(*a))) \\
&= \mathsf{WP}(4, *b == \mathsf{old}(*b) \wedge aux == \mathsf{old}(*a)) \\
&= (*b == \mathsf{old}(*b) \wedge *a == \mathsf{old}(*a))[\mathsf{old}(t) \leftarrow t \in \mathsf{Arith}] \\
&= *b == *b \wedge *a == *a \\
&= \top
\end{aligned}$$

Note pour la rédaction : comme dit dans le cours on a ici omis les substitutions dans le cas où il n'y en a pas (i.e., les at pour les lignes 5 et 6). Vous pourrez même ne pas écrire ces lignes en appliquant directement leur effet (ici, ça reviendrait à ne pas mettre la ligne 4).

(b) En déduire un triplet de Hoare valide.

Réponse :

$\{\top\}\texttt{swap}\{\psi\}$ est un triplet de Hoare valide.

Exercice 2 : Maximum

On considère l'implémentation suivante de la fonction maximum :

```
1  /*@ ensures OneOfTwo: (\result == a || \result == b);
2      ensures Bigger: (\result >= a && \result >= b);
3  */
4  int maximum(int a, int b){
5      if(a >= b) {
6          res = a;
7      }
8      else {
9          res = b;
10     }
11     return res;
12 }
```

(a) Calculez $\mathsf{WP}(\texttt{maximum}, OneOfTwo)$.

Réponse :

METTEZ VOTRE RÉPONSE ICI.

(b) Calculez $\mathsf{WP}(\texttt{maximum}, Bigger)$.

Réponse :

METTEZ VOTRE RÉPONSE ICI.

(c) En déduire un triplet de Hoare valide pour `maximum`.

Réponse :

METTEZ VOTRE RÉPONSE ICI.

Exercice 3 : SortPointer

:

On considère la fonction suivante :

```
1  /*@ ensures Sorted: *a <= *b;
2      ensures Res0: \old(*a) < \old(*b) ==> \result == 0;
3      ensures Res1: \old(*b) <= \old(*a) ==> \result == 1;
4  */
5  int sortPtr(int* a, int* b){
6      if(*b <= *a){
7          aux = *a;
8          *a = *b;
9          *b = aux;
10         res = 1;
11     }
12     else{
13         res = 0;
14     }
15     return res;
16 }
```

(a) Calculez WP($\texttt{sortPtr}, Sorted$).

Réponse :

METTEZ VOTRE RÉPONSE ICI.

(b) Calculez WP($\texttt{sortPtr}, Res0$).

Réponse :

METTEZ VOTRE RÉPONSE ICI.

(c) Calculez WP($\texttt{sortPtr}, Res1$).

Réponse :

METTEZ VOTRE RÉPONSE ICI.

(d) Déduisez-en un triplet de Hoare valide pour `sortPtr`.

Réponse :

METTEZ VOTRE RÉPONSE ICI.