Génie Logiciel Avancé

Année 2019-2020

Prof. Burkhart Wolff
wolff@lri.fr

T. Balabonski, F. Gara, W. Mebrek
blsk@lri.fr, wafaa.mebrek@telecom-sudparis.eu,
fatmagara@gmail.com

EXAMEN

Date : 19 Décembre 2019, Durée : 2 heures

Vous pouvez utiliser les transparents du cours, et rien d'autre. Il est recommandé de lire chaque exercice complètement avant de commencer.

Exercice 1 (barème indicatif : 10 points)
La fonction suivante calcule la multiplication de deux nombres entiers a et b avec b positif ou nul.

```
int mult(int a, int b) {
    int res = 0;
    while(b > 0) {
        if(b mod 2 != 0) {
            res = res + a;
            b = b - 1;
        }
        a = 2 * a;
        b = b/2;
    }
    return res;
}
```

1. Formaliser la specification de la fonction `mult`.
2. Uniquement à partir de la spécification, donner 3 tests pour la fonction `mult`. Pour chacun des tests, donner son objectif, des données d'entrée concrètes et le résultat attendu.
3. Donner le graphe de flot de contrôle de la fonction `mult`.
4. Donner une expression régulière représentant l'ensemble des chemins du graphe.

Toutes les instructions / AllInstructions.

5. Donner le chemin le plus court permettant de satisfaire le critère « toutes les instructions ». On notera ce chemin `ch1`.
6. Par exécution symbolique, calculer la condition associée au chemin `ch1`.
7. Le chemin `ch1` est-il faisable ? Si oui, donner un test concret pour ce chemin. Si non, expliquer pourquoi.

Toutes les chemins / Allpaths$_k$.

On note Allpaths$_k$ l'ensemble des chemins passant au maximum k fois par la boucle.

8. Donner l'ensemble des chemins Allpaths$_k$ pour $k = 3$.
9. Donner une fonction $f(k)$ exprimant la taille de l'ensemble Allpaths$_k$ en fonction de k.
10. Choisir deux chemins ch2 et ch3 de Allpaths$_k$ différents de ch1 et de longueurs différentes. Par exécution symbolique, calculer les conditions associées à ces nouveaux chemins. Sont-ils faisables ? Si oui, donner des tests concrets.
11. Exécuter tous les tests définis dans cet exercice. Quel est votre verdict ? Le programme est-il correct ?

Exercice 2 (Preuve)
[barème indicatif : 6+4 points]

1. Si possible, dériver les triplets de Hoare suivants en utilisant les règles d'inférence introduites dans le cours et rappelées page suivante. Si l'un des triplets est incorrect, donner un contre-example. Vous pouvez utiliser tous les résultats utiles d'arithmétique entière.
   (a) $\vdash \{x^3 < x\}$ `x:==x*x` $\{0 < x\}$
   (b) $\vdash \{x > 1\}$ `IF x < 8 THEN x :== x*x ELSE x :== 33` $\{x < 53\}$
   (c) $\vdash \{X = i! \wedge i > 0\}$ `i :== i+1; X :== X*i` $\{X = i! \wedge i > 0\}$
   (d) $\vdash \{x \leq 0\}$ `WHILE x ≤ 0 DO x :== x+3` $\{1 \leq x \wedge x \leq 4\}$
   (e) $\vdash \{1 \leq x\}$ `WHILE x < 1 DO x :== x+1` $\{x = 1\}$
   (f) $\vdash \{a \geq 0 \wedge a < b \wedge b^2 < c \wedge c \leq a\}$ `c :== b; c :== b*c` $\{c > 12\}$

2. (a) Donner une spécification formelle du programme C ci-dessous sous forme de pré- et post-conditions. Attention à bien prendre en compte que le programme se base sur des entiers non signés (`unsigned`), donc des nombres entre 0 et $2^{32} - 1$.

```
#define SQRT_UINT_MAX 65536

unsigned is_prime_linear(unsigned n)
{
    /* Numbers less than 2 are not prime. */
    if (n < 2)
        return 0;

    /* Find the first non-trivial factor of 'n'. */
    for (unsigned i = 2; i < SQRT_UINT_MAX && i * i <= n; i++) {
        if (n % i == 0)
            return 0;
    }

    /* No factors. */
    return 1;
}
```

   (b) Donner l'invariant de la boucle.
   (c) Une fois l'invariant établi, quel argument (informel) permet de déduire la post-condition ?

Calcul de Hoare

$$\frac{}{\vdash \{P\}\ \texttt{SKIP}\ \{P\}}\ skip \qquad \frac{}{\vdash \{P[x \mapsto E]\}\ \texttt{x :== } E\ \{P\}}\ assignment$$

$$\frac{\vdash \{P \wedge cond\}\ c\ \{Q\} \qquad \vdash \{P \wedge \neg cond\}\ d\ \{Q\}}{\vdash \{P\}\ \texttt{IF}\ cond\ \texttt{THEN}\ c\ \texttt{ELSE}\ d\ \{Q\}}\ ifthenelse$$

$$\frac{\vdash \{P \wedge cond\}\ c\ \{P\}}{\vdash \{P\}\ \texttt{WHILE}\ cond\ \texttt{DO}\ c\ \{P \wedge \neg cond\}}\ while$$

$$\frac{P \to P' \qquad \vdash \{P'\}\ c\ \{Q'\} \qquad Q' \to Q}{\vdash \{P\}\ c\ \{Q\}}\ consequence$$

$$\frac{}{\vdash \{false\}\ c\ \{P\}}\ falseE \qquad \frac{\vdash \{P\}\ c\ \{Q\} \qquad \vdash \{Q\}\ d\ \{R\}}{\vdash \{P\}\ c; d\ \{R\}}\ sequence$$