Demostraciones de corrección: Precondición más Débil

Algoritmos y Estructuras de Datos

30 de agosto de 2023

Repaso: Triplas de Hoare

$\{P\}$ **codigo** $\{Q\}$
¿Es la siguiente tripla válida?

$$\{x \geq 4\}$$
$$\mathbf{x := x + 2}$$
$$\{x \geq 5\}$$

¿Es $\{x \geq 4\}$ la precondición más débil para el programa **x := x + 2** y la postcondición $\{x \geq 5\}$? Intuitivamente no!

Repaso: Precondición más débil (WP)

Definición. La precondición más débil de un programa **S** respecto de una postcondición Q es el predicado P más débil posible tal que $\{P\}\mathbf{S}\{Q\}$. **Notación.** $wp(\mathbf{S}, Q)$.

Ejemplo S: **x := x + 2** y $Q : x \geq 5$
$wp(\mathbf{S}, Q) = x \geq 3$

Repaso: Lenguaje *SmallLang*

- Variables
- Instrucciones
  1. Nada: Instrucción **skip** que no hace nada.
  2. Asignación: Instrucción **x := E**.
- Estructuras de control:
  1. Secuencia: **S1; S2** es un programa, si **S1** y **S2** son dos programas.
  2. Condicional: **if B then S1 else S2 endif** es un programa, si **B** es una expresión lógica y **S1** y **S2** son dos programas.
  3. Ciclo: **while B do S endwhile** es un programa, si **B** es una expresión lógica y **S** es un programa.

Repaso: Axiomas

- **Axioma 1.** $wp(\mathbf{x := E}, Q) \equiv \text{def}(E) \wedge_L Q^x_E$.
- **Axioma 2.** $wp(\mathbf{skip}, Q) \equiv Q$.
- **Axioma 3.** $wp(\mathbf{S1; S2}, Q) \equiv wp(\mathbf{S1}, wp(\mathbf{S2}, Q))$.
- **Axioma 4.** Si $\mathbf{S} = \mathbf{if\ B\ then\ S1\ else\ S2\ endif}$, entonces

$$wp(\mathbf{S}, Q) \equiv def(B) \wedge_L \Big((B \wedge wp(\mathbf{S1}, Q)) \vee (\neg B \wedge wp(\mathbf{S2}, Q)) \Big)$$

Ejercicio 1

proc `transformarEnPar` (inout n: $\mathbb{Z}$) {
  `requiere` $\{n = N_0 \wedge ??\}$
  `asegura` $\{esPar(n) \wedge n > N_0\}$
}

Programa 1	**Programa 1**
S1: `n := 2*n`	**S2:** `n := n + 1`

Ejercicio 2

proc swap (inout a: $\mathbb{Z}$, inout b: $\mathbb{Z}$) $\{$
  requiere $\{a = A_0 \wedge b = B_0 \wedge a \neq 0 \wedge b \neq 0\}$
  asegura $\{a = B_0 \wedge b = A_0\}$
$\}$

S1: `a := a*b`
S2: `b := a/b`
S3: `a := a/b`

Ejercicio 3

`proc diferenciaPositiva` (in a: $\mathbb{Z}$, in b: $\mathbb{Z}$, out res: $\mathbb{Z}$) $\{$

`requiere` $\{??\}$

`asegura` $\{res = |a - b|\}$

$\}$

- Programa 1
  S1: `if (a > b) then res := a - b else res := b - a endif`
- Programa 2
  S2: `res := a-b`

Repaso: Asignación en secuencias

- El programa `b[i]:=E` se reescribe como $b := setAt(b, i, E)$.
- Recordando,

$$\begin{array}{rcl} \text{def}(setAt(b, i, E)) & = & (\text{def}(E) \ \wedge \text{def}(b) \ \wedge \ \text{def}(i)) \\ & \wedge_L & \big(0 \leq i < |b|\big). \end{array}$$

- Aplicando el Axioma 1, tenemos que:

$wp(b[i] := E, Q)$

$$\equiv \quad \big((def\,(b) \wedge def\,(i)) \wedge_L 0 \leq i < |b|\big) \wedge \ \text{def}(E)) \wedge_L Q^{b}_{setAt(b,i,E)}$$

- Dados $0 \leq i, j < |b|$ sabemos que:

$$setAt(b, i, E)[j] \ = \ \begin{cases} E & \text{si } i = j \\ b[j] & \text{si } i \neq j \end{cases}$$

Ejercicio 4

Sea $Q \equiv (\forall j : \mathbb{Z})(0 \leq j < |A| \rightarrow_L A[j] \geq 0)$, i es una variable entera y A es una secuencia de reales.
Calcular $wp(\textbf{A[i+2] := 0}, Q)$.

Ejercicio 5

`proc sumarTodos` (in s: $seq\langle\mathbb{Z}\rangle$, in n: $\mathbb{Z}$, inout suma: $\mathbb{Z}$) $\{$

`requiere`

$$\{suma = suma_0 \land |s| > 0 \land n = |s| \land suma = \sum_{i=0}^{|s|-2} s[i]\}$$

`asegura` $\{suma = \sum_{i_0}^{|s|-1} s[i]\}$

$\}$

S: `suma := suma + s[n-1]`