Práctico 4: Introducción al Cálculo de Programas Imperativos

Algoritmos y Estructuras de Datos I
2^{do} cuatrimestre 2017

1. Para cada uno de los siguientes programas, calcule la precondición más débil y las anotaciones intermedias. Para ello utilice el transformador de predicados *wp*.

   a) $\mathsf{Var}\ x, y : Num;$
   $\{\qquad\qquad\}$
   $x := x + y$
   $\{x = 6 \land y = 5\}$

   b) $\mathsf{Var}\ x : Num;$
   $\{\qquad\}$
   $x := 8$
   $\{x = 8\}$

   c) $\mathsf{Var}\ x : Num;$
   $\{\qquad\}$
   $x := 8$
   $\{x = 7\}$

   d) $\mathsf{Var}\ x, y : Num;$
   $\{\qquad\}$
   $x, y := y, x$
   $\{x = B \land y = A\}$

   e) $\mathsf{Var}\ x, y, a : Num;$
   $\{\qquad\qquad\}$
   $a, x := x, y;$
   $\{\qquad\qquad\}$
   $y := a$
   $\{x = B \land y = A\}$

   f) $\mathsf{Var}\ x, y : Num;$
   $\{\qquad\}$
   $\textbf{if}\ x \geq y \rightarrow$
   $\quad\{\qquad\}$
   $\quad x := 0$
   $\quad\{\qquad\qquad\}$
   $\square\ x \leq y \rightarrow$
   $\quad\{\qquad\}$
   $\quad x := 2$
   $\quad\{\qquad\qquad\}$
   $\textbf{fi}$
   $\{(x = 0 \lor x = 2) \land y = 1\}$

2. Demuestre que las siguientes ternas de Hoare son correctas. En todos los casos las variables x, y son de tipo Int, y a, b de tipo $Bool$.

   a) $\{True\}$
   $\textbf{if}\ x \geq 1 \rightarrow x := x + 1$
   $\square\ x \leq 1 \rightarrow x := x - 1$
   $\textbf{fi}$
   $\{x \neq 1\}$

   b) $\{x \neq y\}$
   $\textbf{if}\ x > y \rightarrow \textbf{skip}$
   $\square\ x < y \rightarrow x, y := y, x$
   $\textbf{fi}$
   $\{x > y\}$

   c) $\{True\}$
   $x, y := y * y, x * x;$
   $\textbf{if}\ x \geq y \rightarrow x := x - y$
   $\square\ x \leq y \rightarrow y := y - x$
   $\textbf{fi}$
   $\{x \geq 0 \land y \geq 0\}$

   d) $\{True\}$
   $\textbf{if}\ \neg a \lor b \rightarrow a := \neg a$
   $\square\ a \lor \neg b \rightarrow b := \neg b$
   $\textbf{fi}$
   $\{a \lor b\}$

   e) $\{N \geq 0\}$
   $x := 0;$
   $\textbf{do}\ x \neq N \rightarrow x := x + 1$
   $\textbf{od}$
   $\{x = N\}$

   f) $\{True\}$
   $r := N;$
   $\textbf{do}\ r \neq 0 \rightarrow$
   $\quad\textbf{if}\ r < 0 \rightarrow r := r + 1$
   $\quad\square\ r > 0 \rightarrow r := r - 1$
   $\quad\textbf{fi}$
   $\textbf{od}$
   $\{r = 0\}$

3. Para cada uno de los siguientes programas, elija valores para las expresiones **E** y **F** de modo que las ternas de Hoare sean correctas.

   a) Var $x, y : Nat$;
   $\{True\}$
   $x, y := x + 1, \mathbf{E}$
   $\{y = x + 1\}$

   b) Var $a, q, c, w : Num$;
   $\{q = a * c \wedge w = c^2\}$
   $a, q := a + c, \mathbf{E}$
   $\{q = a * c\}$

   c) Const $A, B : Nat$;
   Var $q, r : Nat$;
   $\{A = q * B + r\}$
   $q := \mathbf{E};\ r := r - B$
   $\{A = q * B + r\}$

   d) Const $N : Num$;
   Var $x, y, p, q : Num$;
   $\{x * y + p * q = N\}$
   $x := x - p$;
   $q := \mathbf{F}$
   $\{x * y + p * q = N\}$

4. Especifique los siguientes problemas, enunciando pre y postcondición, y luego derive programas imperativos a partir las especificaciones.

   a) Calcular el mínimo de dos valores.

   b) Calcular el valor absoluto de un número.

5. Demuestre que si la terna de Hoare (a) es correcta, entonces la terna (b) también lo es:

   a) $\{P\}$
   if $B_0 \to S_0$
   $\square\ B_1 \to S_1$
   fi
   $\{Q\}$

   b) $\{P\}$
   if $B_0 \to S_0$
   $\square\ \neg B_0 \to S_1$
   fi
   $\{Q\}$

   ¿Qué utilidad tiene esta propiedad cuando se programa en lenguaje C?

6. Analice los siguientes programas anotados. En cada caso, describa en lenguaje natural la postcondición, y decida si el programa efectivamente valida las anotaciones.

   a) Const $N : Int,\ A : array[0, N)\ of Num$;
   Var $s : Num,\ i : Int$;
   $\{N \geq 0\}$
   $i, s := 0, 0$
   do $i \neq N \to$
   $\quad s := s + A.i$
   od
   $\{s = \langle \sum k\ : 0 \leq k < N :\ A.k \rangle\}$

   b) Const $N : Int,\ A : array[0, N)\ of Num$;
   Var $s : Num,\ i : Int$;
   $\{N \geq 0\}$
   $i, s := 0, 0$
   do $i \neq N \to$
   $\quad i := i + 1$
   $\quad s := s + A.i$
   od
   $\{s = \langle \sum k\ : 0 \leq k < N :\ A.k \rangle\}$

   c) Const $N : Int,\ A : array[0, N)\ of Num$;
   Var $s : Num,\ i : Int$;
   $\{N \geq 0\}$
   $i, s := -1, 0$
   do $i \neq N \to$
   $\quad i := i + 1$
   $\quad s := s + A.i$
   od
   $\{s = \langle \sum k\ : 0 \leq k < N :\ A.k \rangle\}$

   d) Const $E : Num,\ N : Int,\ A : array[0, N)\ of Num$;
   Var $i : Int,\ r : Bool$;
   $\{N \geq 0\}$
   $i, r := 0, False$
   do $i \neq N \wedge \neg r \to$
   $\quad$ **if** $A.i = E \to r := True$
   $\quad \square\ A.i \neq E \to$ **skip**
   $\quad$ **fi**
   $\quad i := i + 1$
   od
   $\{\langle \exists k\ : 0 \leq k < N :\ A.k = E \rangle \Rightarrow A.i = E\}$

7. Decida si los siguientes predicados son invariantes válidos del ciclo del programa 6(b). Justifique.

   a) $\{1 \leq i\}$
   b) $\{0 \leq i\}$
   c) $\{0 \leq i \leq N\}$
   d) $\{s = \langle \sum k\ : 0 \leq k < N :\ A.i\,\rangle\}$
   e) $\{0 \leq s \leq \langle \sum k\ : 0 \leq k < N :\ A.i\,\rangle\}$

8. *Swap:* Considere los siguientes programas que intercambian los valores de dos variables x e y de tipo Int:

$x, y := y, x$	$z := x;$	$x := x - y;$
	$x := y;$	$y := x + y;$
	$y := z$	$x := y - x$

   Especifique el *swap* (con pre y postcondición), y verifique que los programas satisfacen la especificación.

9. Derive un programa para calcular el máximo común divisor entre dos enteros positivos. Utilice la siguiente especificación:

   Const $X, Y : Int;$
   Var $x, y : Int;$
   $\{X > 0 \wedge Y > 0 \wedge x = X \wedge y = Y\}$
   S
   $\{x = mcd.X.Y\}$

   Utilice como invariante $\{I : x > 0 \wedge y > 0 \wedge mcd.x.y = mcd.X.Y\}$.

   Para la derivación serán de utilidad las siguientes propiedades del *mcd*:

   a) $mcd.x.x = x$
   b) $mcd.x.y = mcd.y.x$
   c) $x > y \Rightarrow mcd.x.y = mcd.(x - y).y$
   d) $y > x \Rightarrow mcd.x.y = mcd.x.(y - x)$

10. Considere las siguientes definiciones recursivas de la función de exponenciación $exp.x.y$, especificada como $exp.x.y = x^y$:

   a) Definición de complejidad lineal:
   $$\begin{array}{lll} exp.x.y & = & (\ \ y = 0 \rightarrow 1 \\ & & \ \Box\ y \neq 0 \rightarrow x * exp.x.(y-1) \\ & &) \end{array}$$
   b) Definición de complejidad logarítmica:
   $$\begin{array}{lll} exp.x.y & = & (\ \ y = 0 \rightarrow 1 \\ & & \ \Box\ y \neq 0 \rightarrow (\ \ y \mod 2 = 0 \rightarrow exp.(x * x).(y \div 2) \\ & & \qquad\qquad\ \ \Box\ y \mod 2 = 1 \rightarrow x * exp.x.(y-1) \\ & & \qquad\qquad\ \ ) \\ & &) \end{array}$$

   Derive **dos** programas imperativos que calculen la exponenciación, cada uno utilizando una de las definiciones recursivas. Utilice la siguiente especificación:

   Const $X, Y : Int;$
   Var $x, y, r : Int;$
   $\{x = X \wedge y = Y \wedge x \geq 0 \wedge y \geq 0\}$
   S
   $\{r = X^Y\}$

   Utilice como invariante $\{I : y \geq 0 \wedge r * x^y = X^Y\}$.

Ejercicios extra

5. Dado $n > 0$, derive un programa que devuelva en la variable k la mayor potencia de 2 menor o igual que n, utilizando el invariante dado.

   Precondición R: $\{n > 0\}$

   Postcondición Q: $\{0 < k \leq n \wedge n < 2 * k \wedge \langle \exists j \ : 0 \leq j : \ k = 2^j \rangle\}$

   Invariante I: $\{0 < k \leq n \wedge \langle \exists j \ : 0 \leq j : \ k = 2^j \rangle\}$

6. Calcule las precondiciones más débiles (*weakest preconditions*) P de modo que sean correctos los siguientes programas anotados. Agregue además las anotaciones intermedias en caso que haya sentencias compuestas con ";". Asuma que las variables x, y, z, q, r son de tipo Int, las variables i, j de tipo Nat y las variables a, b de tipo $Bool$:

   a) $\{P\}\ x := 8\ \{x = 8\}$
   b) $\{P\}\ x := 8\ \{x \neq 8\}$
   c) $\{P\}\ x := 9\ \{x = 7\}$
   d) $\{P\}\ x := x + 1;\ y := y - 2\ \{x + y = 0\}$
   e) $\{P\}\ x := x + 1;\ y := y - 1\ \{x * y = 0\}$
   f) $\{P\}\ x := x + 1;\ y := y - 1\ \{x + y + 10 = 0\}$
   g) $\{P\}\ z := z * y;\ x := x - 1\ \{z * y^x = C\}$
   h) $\{P\}\ x, y, z := 1, d, c\ \{x * x^y = c^d\}$
   i) $\{P\}\ i, j := i + i, j;\ j := j + i\ \{i = j\}$
   j) $\{P\}\ x := (x - y) * (x + y)\ \{x + y^2 = 0\}$
   k) $\{P\}\ q, r := q + 1, r - y\ \{q * y + r = x\}$
   l) $\{P\}\ a := a \equiv b;\ b := a \equiv b;\ a := a \equiv b\ \{(a \equiv B) \wedge (b \equiv A)\}$

7. Demuestre que si el programa

   $\{P\}$
   if $B \to S$
   fi
   $\{Q\}$

   es correcto, entonces también lo es

   $\{P\}$
   S
   $\{Q\}$

8. Sean S, S_0, S_1, T programas cualesquiera, B_0, B_1 guardas cualesquiera, E, F expresiones cualesquiera. En cada caso, ¿son equivalentes los programas i, ii e iii? En caso afirmativo demostralo, en caso negativo dá un contraejemplo (instanciando los programas y las guardas).

   a) *i*)
   $x := E;$
   $y := F;$

   ii)
   $y := F;$
   $x := E;$

   b) *i*)
   if $B_0 \to S$
   $\Box\ B_1 \to S$
   fi

   ii) S

   c) *i*)
   if $B_0 \to S; S_0; T$
   $\Box\ B_1 \to S; S_1; T$
   fi

   ii)
   if $B_0 \to S; S_0$
   $\Box\ B_1 \to S; S_1$
   fi;
   T

   iii)
   $S;$
   if $B_0 \to S_0; T$
   $\Box\ B_1 \to S_1; T$
   fi