Facultad de Informática

Estructuras de Datos y de la Información

Grupo A, Noviembre 2002

Ejercicios de verificación

Ejercicio 1 En cada uno de los siguientes apartados, suponiendo que $x, y : ent$ y $b : bool$, determinar el predicado P más débil que satisfaga la especificación dada:

1. $\{P\}\ x := x + 2\ \{x \geq 0\}$
2. $\{P\}\ x := 3 * x\ \{x \leq 20\}$
3. $\{P\}\ x := 3 * x\ \{\forall i \in \{1..n\}.x \neq 6 * i\}$
4. $\{P\}\ x := x + 1\ \{x = x + 1\}$.
5. $\{P\}\ x := E\ \{x = E\}$.
6. $\{P\}\ x := x\ \mathbf{mod}\ 2\ \{x = x\ \mathbf{mod}\ 2\}$.
7. $\{P\}\ b := b \vee (x > 0)\ \{\not b\}$
8. $\{P\}\ \langle x, y\rangle := \langle y, x\rangle\ \{x = A \vee y = B\}$
9. $\{P\}\ \langle x, y\rangle := \langle x + 1, y - 1\rangle\ \{x + y > 0\}$
10. $\{P\}\ \langle x, y\rangle := \langle y + 1, x - 1\rangle\ \{x + y > 0\}$
11. $\{P\}\ \langle x, y\rangle := \langle y - 2, x + 3\rangle\ \{x + y > 0\}$

Ejercicio 2 Suponiendo $x, y : ent$, calcular en cada caso el predicado P más débil que satisfaga la especificación dada.

1. $\{P\}\ x := x * x\ ;\ \ x := x + 1\ \{x > 0\}$
2. $\{P\}\ x := x + y\ ;\ \ y := 2 * y - x\ \{y > 0\}$
3. $\{P\}\ y := 4 * x\ ;\ \ x := x * x - y\ ;\ \ x := x + 4\ \{x > 0\}$
4. $\{P\}\ x := y\ ;\ \ y := x\ \{x = A \vee y = B\}$

Ejercicio 3 : Verificar el algoritmo siguiente, suponiendo x, y : *ent*:

$$\begin{array}{l} \{x = X \vee y = Y\} \\ \quad x := x - y\ ; \\ \quad y := x + y\ ; \\ \quad x := y - x \\ \{x = Y \vee y = X\} \end{array}$$

Ejercicio 4 Demostrar que las dos reglas siguientes para una instrucción de la forma **si** B **entonces** S_1 **si no** S_2 **fsi** son equivalentes.

$$\frac{\{Q \wedge B\}\, S_1\, \{R\} \qquad \{Q \wedge \neg B\}\, S_2\, \{R\}}{\{Q \wedge \mathit{Def}(B)\}\ \mathbf{si}\ B\ \mathbf{entonces}\ S_1\ \mathbf{si\ no}\ S_2\ \mathbf{fsi}\ \{R\}}$$

$$\frac{\{Q_1\}\, S_1\, \{R\} \qquad \{Q_2\}\, S_2\, \{R\}}{\{\mathit{Def}(B) \wedge ((Q_1 \wedge B) \vee (Q_2 \wedge \neg B))\}\ \mathbf{si}\ B\ \mathbf{entonces}\ S_1\ \mathbf{si\ no}\ S_2\ \mathbf{fsi}\ \{R\}}$$

Ejercicio 5 Demuestra la corrección de la siguiente especificación:

$\{x = X \vee y = Y \vee x > y > 0\}$
mientras $y \neq 0$ **hacer**
$\langle x, y\rangle := \langle y, x \textbf{ mod } \mathbf{y}\rangle$
fmientras
$\{\mathbf{x} = \mathbf{mcd}(\mathbf{X}, \mathbf{Y})\}$

Ejercicio 6 Demuestra la corrección de la siguiente especificación:

$\{n \geq 0\}$
$x := 0\ ;\ \ y := 1\ ;$
mientras $x \neq n$ **hacer**
$x := x + 1\ ;\ \ y := y + y$
fmientras
$\{y = 2^n\}$

Ejercicio 7 Demuestra la corrección de la siguiente especificación:

$\{x = X \wedge y = Y \wedge Y \geq 0\}$
$p := 1\ ;$
mientras $y \neq 0$ **hacer**
$p := p * x\ ;\ \ y := y - 1$
fmientras
$\{p = X^Y\}$

Ejercicio 8 Demuestra la corrección de la siguiente especificación:

$\{x \geq 0\}$
$r := 0\ ;$
mientras $(r + 1) * (r + 1) \leq x$ **hacer**
$r := r + 1$
fmientras
$\{r \geq 0 \wedge r^2 \leq x < (r + 1)^2\}$

Ejercicio 9 Demuestra la corrección de la siguiente especificación:

$\{n \geq 0\}$
$s := 0\ ;\ \ i := 1\ ;$
mientras $i \leq n$ **hacer**
$s := s + a[i]\ ;$
$i := i + 1\ ;$
fmientras
$\{s = \sum_{\xi=1}^{n} a[\xi]\}$

Ejercicio 10 Demuestra la corrección de la siguiente especificación:

$\{A \geq 0 \wedge B \geq 0\}$
$a := A\ ;\ \ b := B\ ;\ \ u := 0\ ;$
mientras $a > 0$ **hacer**
  si $\neg par(a)$ **entonces** $u := u + b$ **fsi**
  $a := a$ **div 2** ; $\mathbf{b := 2 * b}$
fmientras
$\{\mathbf{u = A * B}\}$

Ejercicio 11 (Búsqueda lineal no acotada) Demuestra la corrección de la siguiente especificación:

$\{n > 0 \wedge (\exists \beta \in \{1..n\}.a[\beta] = x)\}$
$i := 1\ ;$
mientras $a[i] \neq x$ **hacer**
  $i := i + 1$
fmientras
$\{a[i] = x \wedge (\forall \beta \in \{1..i-1\}.a[\beta] \neq x)\}$

Ejercicio 12 (Búsqueda lineal acotada) Demuestra la corrección de la siguiente especificación:

$\{n \geq 0\}$
$i := 1\ ;$
mientras $i \leq n \wedge a[i] \neq x$ **hacer**
  $i := i + 1$
fmientras
$encontrado := (i \leq n)$
$\{encontrado = (\exists \beta \in \{1..n\}.a[\beta] = x \wedge (\forall \xi \in \{1..\beta - 1\}.a[\xi] \neq x))\}$

Ejercicio 13 Demuestra la corrección de la siguiente especificación:

$\{n \geq 1\}$
$s := 0\ ;\ \ x := n\ ;$
mientras $x > 0$ **hacer**
  $s := s + x^2\ ;$
  $x := x - 1$
fmientras
$\{s = \sum_{\xi=1}^{n} \xi^2\}$

Ejercicio 14 Demuestra que el siguiente programa deja en x el máximo y en y el mínimo entre los valores iniciales de x e y:

si $x < y$ **entonces**
  $aux := x\ ;$
  $x := y\ ;$
  $y := aux$
fsi

Ejercicio 15 Demuestra que el siguiente programa calcula el valor absoluto de x:

si $x < 0$ **entonces** $x := -x$ **fsi**

Ejercicio 16 Demuestra que el siguiente programa deja en z la mayor potencia de 2 que no supera a x:

$z := 1$;
mientras $(2 * z \leq x)$ **hacer**
    $z := 2 * z$
fmientras

Ejercicio 17 Demuestra que el siguiente programa decide en b si el número x es primo:

$d := 2$; $b :=$ cierto
mientras $(d \neq x \wedge b)$ **hacer**
    si x **mod d = 0 entonces b** := falso **fsi**
    d := d + 1
fmientras

Ejercicio 18 (1^{er} parcial 01/02) Dado un vector de enteros, se dice que un índice es *productivo* si el elemento situado en dicha posición coincide con el producto de los elementos situados en las posiciones anteriores.

(a) Especificar una función *numprod* que dado un vector de enteros, devuelva el número l de índices productivos que aparecen en las primeras n posiciones. Si $n = 0$ la función debe devolver 0.

(b) Se ha escrito el siguiente fragmento de código como cuerpo de la función *numprod*:

$j := 1$; $p := 1$; $l := 0$;
mientras $j < n$ **hacer**
    si $p = v[j+1]$ **entonces** $l := l + 1$ **fsi** ;
    $p := p * v[j+1]$;
    $j := j + 1$
fmientras

Verificar si es correcto respecto a la especificación proporcionada. Si encuentras algún error corrígelo y verifica la versión corregida.

Ejercicio 19 (junio 01/02) Dado un vector de enteros y un entero n, la acción de *positivizar* el vector consiste en sustituir en las n primeras posiciones del vector todas las apariciones de números negativos (< 0) por 0, dejando sin modificar el resto de las n posiciones (las que contienen números ≥ 0).

(a) Especificar un procedimiento *positivizar* que positivice (las n primeras posiciones de) un vector v de enteros.

(b) Se ha escrito el siguiente fragmento de código como cuerpo del procedimiento *positivizar*:

$j := 0$;
mientras $j < n$ **hacer**
  si $v[j] < 0$ **hacer** $v := asig(v, j, 0)$ **fsi** ;
  $j := j + 1$
fmientras

Verificar si es correcto respecto a la especificación proporcionada. Si encuentras algún error corrígelo y verifica la versión corregida.

Recordar que $asig(v, i, e)$ es el vector que resulta de asignar el valor de e a la posición i del vector v:

$$asig(v, i, e)[j] = \begin{cases} e \text{ si } j = i \\ v[j] \text{ si } j \neq i \end{cases}$$

Ejercicio 20 (septiembre 01/02) En todos los apartados de este ejercicio se supondrá definido un tipo *vect* de vectores de enteros de la siguiente manera:

$$tipo\ vect = \textbf{vector}\ [0..1000]\ \textbf{de}\ entero$$

Además se supondrá que un número binario $b_n \dots b_0$ se representa como un vector v de tipo *vect* donde $v[i] = b_i$ para $i = 0, \dots, n$.

(a) Especificar una función *decimal* que dado un número binario en forma de vector v y un entero n (indicando la posición de la cifra más significativa) calcule el número decimal d correspondiente.

(b) Se ha escrito el siguiente fragmento de código como cuerpo de la función *decimal*:

$j := 0$; $d := v[0]$; $p := 2$;
mientras $j < n$ **hacer**
  $d := d + v[j + 1] * p$;
  $p := p * 2$;
  $j := j + 1$
fmientras

Verificar si es correcto respecto a la especificación proporcionada. Si encuentras algún error con respecto a ella, corrígelo y verifica la versión corregida.