

Algoritmos y Estructuras de Datos I - 2007

Práctico 6: Programación Imperativa - Derivación de ciclos

Docentes: Silvina Smith, Renato Cherini, Valeria Rulloni,
Alejandro Peralta Frias, Mariana Badano

6 de noviembre de 2007

1. Determinar el predicado más débil P para el cual el programa es correcto:

$$\begin{array}{ll}
 \begin{array}{l}
 \llbracket \text{Var } x : \text{Int} \\
 \{P\} \\
 \mathbf{do} \ x \neq 0 \rightarrow x := x - 1 \\
 \mathbf{od} \\
 \{x = 0\} \\
 \rrbracket
 \end{array} &
 \begin{array}{l}
 \llbracket \text{Var } x : \text{Int} \\
 \{P\} \\
 \mathbf{do} \ x \neq 0 \rightarrow x := x - 2 \\
 \mathbf{od} \\
 \{x = 0\} \\
 \rrbracket
 \end{array} \\
 a) & b)
 \end{array}$$

2. En cada caso derivar un programa imperativo S que satisfaga la especificación dada:

$$\begin{array}{l}
 \llbracket \text{Var } r : \text{Bool} \\
 \text{Con } AS, BS : [\text{Int}] \\
 a) \ \{N \geq 0 \wedge \#AS = N + 1 \wedge \#BS = N + 1\} \\
 \quad S \\
 \quad \{r = (\forall i : 0 \leq i \leq N : AS.i = BS.(N - i))\} \\
 \rrbracket \\
 \\
 \llbracket \text{Var } f : \text{array}[0..M + 1] \text{ of } \text{Int} \\
 \text{Con } N, M : \text{Int}, XS : [\text{Int}] \\
 b) \ \{(\forall i : 0 \leq i < N : 0 \leq XS.i \leq M) \wedge N \geq 0 \wedge \#XS = N\} \\
 \quad S \\
 \quad \{(\forall i : 0 \leq i \leq M : f.i = (N j : 0 \leq j < N : XS.j = i))\} \\
 \rrbracket
 \end{array}$$

Nota: En este ejercicio hay dos constantes que pueden ser reemplazadas por variables: N y M . Analice la diferencia entre cambiar primero M y luego N o viceversa.

3. Sea $N : \text{Int}$ y $N \geq 0$.

- a) Derivar un programa que calcule el menor entero x que satisface $x^3 + x \geq N$.
b) Derivar un programa que calcule el mayor entero x que satisface $x^3 + x \leq N$.

4. Sea $A : \text{array}[0..N] \text{ of } \text{Int}$.

- a) Derivar un programa que determine si todos los elementos de A son positivos.
b) Derivar un programa que determine si algún elemento de A es positivo.

5. Derivar un programa que almacene en una variable el valor máximo de un arreglo de enteros.

6. Derivar un programa que calcule la cantidad de elementos pares de un arreglo de enteros.

7. (Difícil). Calcular un programa que dados dos enteros positivos X e Y , devuelva en una variable el mínimo común múltiplo de ambos.

Ayuda:

a) El mínimo común múltiplo de dos enteros positivos se puede especificar por:

$$mcm.x.y = (Min\ n : 1 \leq n \wedge n\ mod\ x = 0 \wedge n\ mod\ y = 0 : n)$$

b) Una propiedad que puede facilitar el ejercicio es: $a * b = mcd.a.b * mcm.a.b$

8. Escribir programas imperativos para cada uno de los siguientes problemas, cuya derivación puede encontrarse en el apunte del curso. En un primer paso, suponer que el lenguaje dispone de primitivas para el manejo de listas; luego implementar las listas usando arreglos.

a) Evaluación de un polinomio.

b) Cálculo de una aproximación de la constante matemática e .

c) Problema de la lista balanceada.

d) Problema del máximo segmento balanceado.