

## Ejercicios sobre Verificación de Programas (Clase 14)

**Ejercicio 1.** Indicar y justificar (semánticamente, es decir sin utilizar axiomas y reglas) si valen o no las siguientes ternas de Hoare:

- 1)  $\{x > 0\}$  while  $x \neq 0$  do  $x := x - 1$  od  $\{x = 0\}$
- 2)  $\{\text{true}\}$  while  $x \neq 0$  do  $x := x - 1$  od  $\{\text{true}\}$

Comentario: el predicado `true` representa cualquier estado. Por lo tanto, en (2) hay que indicar si el programa siempre termina, independientemente de con qué estado empiece.

**Ejercicio 2.** Asumiendo que se cumple  $\{p\} S \{q\}$ , es decir que a partir de todo estado inicial que satisface la precondition  $p$ , luego de la ejecución del programa  $S$  se alcanza un estado final que satisface la postcondición  $q$ , indicar y justificar si valen o no las afirmaciones siguientes:

- 1) Si el programa  $S$  terminó en un estado final que no satisface  $q$ , entonces significa que empezó en un estado inicial que no satisface  $p$ .
- 2) Si el programa  $S$  terminó en un estado final que satisface  $q$ , entonces significa que empezó en un estado inicial que satisface  $p$ .

**Ejercicio 3.** Aplicar el axioma de asignación (ASI) para obtener las preconditiones correspondientes a las siguientes ternas de Hoare:

- 1)  $\{?\} x := x + 1 \{x + 1 \neq 0\}$
- 2)  $\{?\} x := y \{x = y\}$

**Ejercicio 4.** Especificar un programa que a partir de un estado inicial en el que  $x > 0$ , termine en un estado final en el que el valor de la variable  $y$  sea el doble del valor inicial de  $x$ .

Ayuda: la especificación  $(x > 0, y = 2.x)$  NO es correcta.

**Ejercicio 5.** Se cuenta con un programa que calcula en una variable *prod* el producto de dos números enteros  $x$  e  $y$  mayores o iguales que cero, sumando  $y$  veces el valor de  $x$ . Una terna de Hoare que expresa esto es la siguiente:

```
{x ≥ 0 ∧ y ≥ 0 ∧ prod = 0 ∧ tope = 0}
while tope ≠ y do
    prod := prod + x ;
    tope := tope + 1
od
{prod = x.y}
```

Para la verificación del programa, que es un *while*, puede servir utilizar como invariante el predicado  $p = (\text{prod} = x.\text{tope})$ , y como variante la función  $t = (y - \text{tope})$ . Se pide comprobar (informalmente, sin recurrir a los axiomas y reglas del método presentado en clase) que efectivamente  $p$  y  $t$  cumplen con las definiciones de invariante y variante, respectivamente. Es decir, se pide comprobar:

- 1)  $p$  se cumple antes del *while*.
- 2)  $p$  se cumple después de toda iteración.
- 3)  $t$  se decrementa después de toda iteración.
- 4)  $t$  siempre es mayor o igual que cero.