Corrección de Ciclos

Algoritmos y Estructuras de Datos - Práctica (Labo)

viernes 12 de abril

Teorema de corrección de un ciclo

- **Teorema.** Sean un predicado I y una función $fv : \mathbb{V} \to \mathbb{Z}$ y supongamos que $I \Rightarrow \mathrm{def}(B)$. Si

  1. $P_C \Rightarrow I$,
  2. $\{I \wedge B\}\, S\, \{I\}$,
  3. $I \wedge \neg B \Rightarrow Q_C$,
  4. $\{I \wedge B \wedge v_0 = fv\}$ **S** $\{fv < v_0\}$,
  5. $I \wedge fv \leq 0 \Rightarrow \neg B$,

  ... entonces la siguiente tripla de Hoare es válida:

  $$\{P_C\}\ \textbf{while B do S endwhile}\ \{Q_C\}$$

¡Observaciones importantes!

Observación #1

Se requiere inventiva para enunciar un **invariante** y una **función variante**.

Observación #2

¡Nunca calculamos la wp de un ciclo!

... Sí sabemos que vale la implicación:
$P_C \Rightarrow wp(\textbf{while B do S endwhile}, Q_C)$

Machete de Axiomas

- **Axioma A1.** $wp(\texttt{x := E}, Q) \equiv \text{def}(E) \wedge_L Q^x_E$.
- **Axioma A2.** $wp(\texttt{skip}, Q) \equiv Q$.
- **Axioma A3.** $wp(\texttt{S1; S2}, Q) \equiv wp(\texttt{S1}, wp(\texttt{S2}, Q))$.
- **Axioma A4.** $wp(\texttt{if B then S1 else S2 endif}, Q) \equiv$

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

Machete Lógica

- **L1** Si P entonces $(P \vee Q)$.
- **L2** Si $P \wedge Q$ entonces P.
- **L3** Si $(P \Rightarrow Q) \wedge (\neg P \Rightarrow Q)$ entonces Q.
- **L4** Si $(R \Rightarrow Q)$ entonces $(\exists x)(R \wedge P) \Rightarrow (\exists x)(Q \wedge P)$.
- **L5** Si $(R \Rightarrow Q)$ entonces $(\forall x)(Q \Rightarrow P) \Rightarrow (\forall x)(R \Rightarrow P)$.
- **L6** Si $(\forall x)(R \Rightarrow P) \wedge (\forall x)(Q \Rightarrow P)$
  entonces $(\forall x)((R \vee Q) \Rightarrow P)$

Ejercicio #1: Búsqueda de elemento

Demostrar la correctitud de la siguiente tripla de Hoare

$\{P_C : i = |s| - 1 \wedge res = false\}$

```
while i >= 0 do
    if s[i] == 7
        res = true
    else
      skip
    endif
    i = i -1
endwhile
```

$\{Q_C : res = true \iff (\exists i : \mathbb{Z})(0 \leq i < |s| \wedge_L s[i] = 7)\}$

Guía para demostrar programas con ciclos

Para probar $\{Pre\}$`S1;while...;S3`$\{Post\}$ voy a probar primero

1. $Pre \Rightarrow_L wp(S1, P_C)$
2. $P_C \Rightarrow_L wp(while..., Q_C)$
3. $Q_C \Rightarrow_L wp(S3, Post)$

donde $S1$ es ..., P_C es ... y Q_c es ..., $S3$ es ... .

Dado que valen (1), (2) y (3), por monotonía, el predicado $Pre \Rightarrow_L wp($`S1;while...;S3`$, Post)$ es verdadero.

Ejercicio #2: ¿Está ordenado?

Dada la especificación,

```
proc estaOrdenado(in s:  seq<ℤ>) :  bool
     requiere {|s| ≥ 1}
     asegura {
```
$res = true \iff (\forall i : \mathbb{Z})(0 \leq i < |s| - 1 \Rightarrow s[i] \leq s[i+1])$

```
}
```

Demostrar que el siguiente programa cumple la misma:

```
i := 0
res := false

while i < |s|-1 && s[i] <= s[i+1]
    i = i + 1
endwhile


if i == |s| - 1 then
    res = true
else
    skip
fi
```

Fin.