



Integrantes: Andrés M. Hense, Victoria Espil

Ejercicio 12 Para probar que el programa es correcto respecto a la especificación, vamos a probar estas implicaciones por separado, y por monotonía llegaremos a que el programa es correcto.

- $Pre \rightarrow wp(\text{codigo previo al ciclo}, P_c)$
- $P_c \rightarrow wp(\text{ciclo}, Q_c)$
- $Q_c \rightarrow wp(\text{codigo posterior al ciclo}, Post)$

Especificación del ciclo:

- $Pre : True$
- $Post : r = True \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e)$
- $P_c : i = 0 \wedge j = -1$
- $Q_c : i = |s| \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e)$
- $B : i < |s|$
- $I : 0 \leq i \leq |s| \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i) \wedge_L s[k] = e)$
- $f_v : |s| - i$

Empecemos probando la primer implicación

$Pre \rightarrow wp(\text{codigo previo al ciclo}, P_c)$

$$\begin{aligned} wp(i := 0; j := -1, P_c) &\equiv wp(i := 0, wp(j := -1, P_c)) \\ &\equiv wp(i := 0, (P_c)_{-1}^j) \\ &\equiv (i = 0 \wedge -1 = -1)_0^i \\ &\equiv 0 = 0 \wedge True \\ &\equiv True \end{aligned}$$

Luego $True \rightarrow True$, es tautología.

$P_c \rightarrow wp(\text{ciclo}, Q_c)$

Demostraremos que el ciclo es correcto respecto a su especificación y además termina.

$P_c \Rightarrow I$

Debemos demostrar que vale I sabiendo que vale P_c

- $0 \leq i \leq |s|$
Por P_c sabemos que $i = 0$, entonces $0 \leq i$ vale. Además, $|s| \geq 0$ (porque las listas no pueden tener una cantidad negativa de elementos), luego $|s| \geq i$
- $(j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i) \wedge_L s[k] = e)$
 P_c indica que $i = 0$ y $j = -1$. Luego como no existe un k entre $0 \leq k < 0$, la doble implicación no se cumple, por lo que tiene que pasar que $j = -1$.

$(I \wedge \neg B) \Rightarrow Q_c$

Queremos demostrar que vale Q_c , asumiendo que vale $I \wedge \neg B$.

- Por I sabemos que $i \leq |s|$, y por $\neg B$ sabemos que $i \geq |s|$. Entonces i debe ser igual a $|s|$

- Es trivial ver que vale $(j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e)$ ya que, al reemplazar $i = |s|$ en el invariante llego a eso.

$\{I \wedge B\}$ **ciclo** $\{I\}$

Queremos ver que vale la siguiente tripla de Hoare $\{I \wedge B\}$ **ciclo** $\{I\}$.

Llamemos S1 a la primer instrucción del cuerpo del ciclo, S2 a la segunda:

S1: **if** $(s[i] = e)$ **then** $j := i$ **else** *skip* **endif**

S2: $i := i + 1$

Lo primero que haremos es calcular $wp(\text{ciclo}, I)$.

$$wp(S1; S2, I) \stackrel{Ax3}{\equiv} wp(S1, wp(S2, I)) \quad (1)$$

Antes de seguir, debemos calcular $wp(S2, I)$. Para eso usaremos el axioma 1:

$$\begin{aligned} wp(S2, I) &\stackrel{Ax1}{\equiv} \text{def}(i + 1) \wedge_L I_{i+1}^i \\ &\equiv \text{true} \wedge_L (0 \leq i + 1 \leq |s| \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e)) \\ &\equiv 0 \leq i + 1 \leq |s| \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e) \end{aligned}$$

Volviendo a (1), reemplazamos $wp(S2, I)$ y nos queda:

$$\begin{aligned} wp(S1; S2, I) &\stackrel{Ax3}{\equiv} wp(S1, wp(S2, I)) \\ &\equiv wp(S1, 0 \leq i + 1 < |s| \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e)) \\ &\equiv wp(S1, I_{i+1}^i) \\ &\equiv \text{def}(s[i] = e) \wedge_L \left(\left((s[i] = e) \wedge wp(j := i, I_{i+1}^i) \right) \vee \left(\neg(s[i] = e) \wedge wp(\text{skip}, I_{i+1}^i) \right) \right) \\ &\equiv 0 \leq i < |s| \wedge \left(\left((s[i] = e) \wedge wp(j := i, I_{i+1}^i) \right) \vee \left(s[i]! = e \wedge I_{i+1}^i \right) \right) \\ &\equiv 0 \leq i < |s| \wedge \left(\left((s[i] = e) \wedge wp(j := i, 0 \leq i + 1 < |s| \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e)) \right) \vee \right. \\ &\quad \left. \left(s[i]! = e \wedge 0 \leq i + 1 < |s| \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e) \right) \right) \\ &\equiv 0 \leq i < |s| \wedge \left(\left((s[i] = e) \wedge (i! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e) \right) \vee \right. \\ &\quad \left. \left(s[i]! = e \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e) \right) \right) \\ &\equiv 0 \leq i < |s| \wedge \left(\left((s[i] = e) \wedge (\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e) \right) \vee \right. \\ &\quad \left. \left(s[i]! = e \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e) \right) \right) \end{aligned}$$

(En el pase del ultimo renglon estoy usando que i no puede estar en rango y valer algo negativo, por lo que $(i! = -1)$ es verdadero)

Una vez calculada la precondition más débil, debemos ver si $(I \wedge B)$ implican dicha precondition. Probamos cada parte por separado:

- $0 \leq i < |s|$

Esta incluido en I .

- $\left((s[i] = e) \wedge (\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e \right) \vee \left(s[i]! = e \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e) \right)$

Analicemos por separado:

- $(s[i] = e) \wedge (\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e$
Supongamos que vale $(s[i] = e)$, entonces es cierto que existe un k tal que... en particular $k = i$.

- $s[i]! = e \wedge (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i + 1) \wedge_L s[k] = e)$

Supongo que $(s[i]! = e)$, luego como el ultimo termino del existe se que no vale, lo puedo reescribir como $(j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < i) \wedge_L s[k] = e)$, y esto es lo mismo que tengo en I .

Como $(I \wedge B) \Rightarrow wp(ciclo, I)$, podemos afirmar que el cuerpo del ciclo preserva el invariante.

$$\{(I \wedge B \wedge v_0 = f_v)\} \text{ ciclo } \{(f_v < v_0)\}$$

Dado que queremos demostrar que vale una tripla de Hoare, comenzaremos calculando la precondition más débil $wp(ciclo, f_v < v_0)$.

$$\begin{aligned} wp(S1; S2, f_v < v_0) &\stackrel{Ax3}{=} wp(S1, wp(S2, |s| - i < v_0)) \\ &\stackrel{Ax1}{=} wp(S1, true \wedge_L |s| - (i + 1) < v_0) \\ &\stackrel{Ax3}{=} true \wedge_L (true \wedge_L |s| - (i + 1) < v_0) \\ &\equiv |s| - i - 1 < v_0 \end{aligned}$$

Es decir, $wp(S1; S2, f_v < v_0) = |s| - i - 1 < v_0$. Ahora debemos ver que $(I \wedge B \wedge v_0 = f_v)$ implican dicha WP. Parte de la hipótesis es que $v_0 = f_v$, es decir $v_0 = |s| - i$. Restando 1 a ambos lados se prueba, $|s| - i - 1 = v_0 - 1 < v_0$.

$$(I \wedge f_v \leq 0) \Rightarrow \neg B$$

Debemos mostrar que vale $\neg B$, es decir $i \geq |s|$.

Sabemos que $f_v \leq 0$, es decir $|s| - i \leq 0$, luego $|s| \leq i$, como queriamos demostrar.

$$Q_c \rightarrow wp(\text{codigo posterior al ciclo}, Post)$$

S: if $(j! = -1)$ then $r = True$ else $r = False$ endif

$$\begin{aligned} wp(\mathbf{S}, Post) &\equiv \text{def}(j! = -1) \wedge_L \left(\left((j! = -1) \wedge wp(r = True, Post) \right) \vee \left(\neg(j! = -1) \wedge wp(j = False, Post) \right) \right) \\ &\equiv True \wedge \left(\left((j! = -1) \wedge Post_{True}^r \right) \vee \left(j = -1 \wedge Post_{False}^r \right) \right) \\ &\equiv \left(j! = -1 \wedge True = True \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e) \right) \vee \\ &\quad \left(j = -1 \wedge False = True \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e) \right) \\ &\equiv \left(j! = -1 \wedge True \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e) \right) \vee \\ &\quad \left(j = -1 \wedge False \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e) \right) \\ &\equiv \left(j! = -1 \wedge ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e) \right) \vee \\ &\quad \left(j = -1 \wedge \neg((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e) \right) \\ &\equiv (j! = -1) \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e) \end{aligned}$$

Como llegue a lo mismo que Q_c , entonces vale.