Precondición más débil de ciclos

Algoritmos y Estructuras de Datos I

Repaso: Triplas de Hoare

- Consideremos la siguiente tripla de Hoare:

$$\{P\}\ \mathbf{S}\ \{Q\}.$$

- Esta tripla es válida si se cumple que:
  1. Si el programa S comienza en un estado que cumple P ...
  2. ... entonces termina luego de un número finito de pasos ...
  3. ... Y además en un estado que cumple Q.

Repaso: Lenguaje SmallLang

- Definimos un lenguaje imperativo basado en variables y las siguientes instrucciones:
  1. Nada: Instrucción **skip** que no hace nada.
  2. Asignación: Instrucción **x := E**.
- Además, tenemos las siguientes estructuras de control:
  1. Secuencia: **S1; S2** es un programa, si **S1** y **S2** son dos programas.
  2. Condicional: **if B then S1 else S2 endif** es un programa, si **B** es una expresión lógica y **S1** y **S2** son dos programas.
  3. Ciclo: **while B do S endwhile** es un programa, si **B** es una expresión lógica y **S** es un programa.

Repaso: Precondición más débil

- **Definición.** La precondición más débil de un programa **S** respecto de una postcondición Q es el predicado P más débil posible tal que $\{P\}\mathbf{S}\{Q\}$.
- **Notación.** $wp(\mathtt{S}, Q)$.
- **Teorema:** Decimos que $\{P\}$ **S** $\{Q\}$ es válida sii $P \Rightarrow_L wp(\mathtt{S}, Q)$

Repaso: Axiomas wp

- **Axioma 1.** $wp(\texttt{x := E}, Q) \equiv \text{def}(E) \wedge_L Q^x_E$
- **Axioma 2.** $wp(\texttt{skip}, Q) \equiv Q$
- **Axioma 3.** $wp(\texttt{S1; S2}, Q) \equiv wp(\texttt{S1}, wp(\texttt{S2}, Q))$
- **Axioma 4.** $wp(\texttt{if B then S1 else S2 endif}, Q) \equiv$

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

- **Observación:** $wp(\texttt{b[i]:=E}, Q) \equiv wp(\texttt{b:=setAt(b,i,E)}, Q)$

¿Cuál es la precondición más débil?

$\{???\}$

```
while (x>0) do
  x := x -1
endwhile
```

$\{x = 0\}$

$$wp(\texttt{while (x>0) do x := x -1 endwhile}, x = 0) \equiv x \geq 0$$

¿Cuál es la precondición más débil?

$\{???\}$

```
while (x>0) do
  x := x -1
endwhile
```

$\{Q : x = 0 \wedge j = 15\}$

$$wp(\texttt{while ...}, Q) \equiv x \geq 0 \wedge j = 15$$

¿Cuál es la precondición más débil?

$\{P : ???\}$

```
j := 0;
while (x<5) do
  x := x + 1;
  j := j + 1
endwhile
```

$\{x = 5 \wedge j = 5\}$

$$\begin{aligned} P &\equiv wp(\texttt{j:=0; while ...}, x = 5 \wedge j = 5) \\ &\equiv wp(\texttt{j:=0}, wp(\texttt{while ...}, x = 5 \wedge j = 5)) \\ &\equiv wp(\texttt{j:=0}, x = 0 \wedge j = 0)) \\ &\equiv x = 0 \wedge 0 = 0 \\ &\equiv x = 0 \end{aligned}$$

¿Cuál es la precondición más débil?

$\{???\}$

```
while (x==5) do
  x := 5
endwhile
```

$\{x \neq 5\}$

$$wp(\texttt{while } \ldots, x \neq 5) \equiv x \neq 5$$

¿Es válida la siguiente tripla de Hoare?

$\{n \geq 0 \wedge j = 1 \wedge z = 0\}$

```
while (j ≤ n) do
  z := z + j;
  j := j + 1
endwhile
```

$\{z = \sum_{k=1}^{n} k\}$

Precondición más débil de un ciclo

- Supongamos que tenemos el ciclo **while B do S endwhile**.
- **Definición.** Definimos $H_k(Q)$ como el predicado que define el conjunto de estados a partir de los cuales la ejecución del ciclo termina en exactamente k iteraciones:

$$\begin{array}{rcl} H_0(Q) & \equiv & \mathrm{def}(B) \wedge \neg B \wedge Q, \\ H_{k+1}(Q) & \equiv & \mathrm{def}(B) \wedge B \wedge wp(\mathtt{S}, H_k(Q)) \qquad \text{para } k \geq 0. \end{array}$$

- **Propiedad:** Si el ciclo realiza a lo sumo k iteraciones, entonces

$$wp(\texttt{while B do S endwhile}, Q) \equiv \vee_{i=0}^{k} H_i(Q)$$

Ejemplo

$\{???\}$

```
while (0<j && j<3) do
  j := j +1
endwhile
```

$\{j = 3\}$

- A lo sumo, se va a ejecutar 2 veces el cuerpo del ciclo
- ¿Cuál es la precondición más débil?

$$\begin{array}{rl} & wp(\texttt{while } 0 < j < 3 \texttt{ do j:=j+1 endwhile}, j = 3) \\ \equiv & \vee_{j=0}^{2} H_j(j = 3) \\ \equiv & H_0(j = 3) \vee H_1(j = 3) \vee H_2(j = 3) \\ \equiv & j = 1 \vee j = 2 \vee j = 3 \end{array}$$

Otro ejemplo

$\{???\}$

```
while (0<j && j<n) do
  j := j +1
endwhile
```

$\{i \geq 0\}$

- ¿Cuántas veces se va a ejecutar el cuerpo del ciclo?
- ¿Podemos usar la propiedad anterior para conocer la precondición más débil?
- ¡No! Porque no podemos fijar a priori una cota superior a la cantidad de iteraciones que va a realizar el ciclo.

Precondición más débil de un ciclo

- **Intituivamente:** $wp(\texttt{while B do S endwhile}, Q)$ tiene que ser una fórmula lógica capaz de capturar todos los estados tales que, luego de ejecutar el ciclo una cantidad arbitraria de veces, vale Q.
- **Axioma 5:**

$$wp(\texttt{while B do S endwhile}, Q) \equiv (\exists_{i \geq 0})(H_i(Q))$$

Precondición más débil de un ciclo

- Ahora tratemos de usar el **Axioma 5**:

$$\begin{aligned} & wp(\texttt{while B do S endwhile}, Q) \\ & \quad \equiv (\exists_{i \geq 0}) H_i(Q) \\ & \quad \equiv H_0(Q) \vee H_1(Q) \vee H_2(Q) \vee \dots \\ & \quad \equiv \vee_{i=0}^{\infty}(H_i(Q)) \end{aligned}$$

¡Es una fórmula infinitaria!

- Por lo tanto, no podemos usar mecánicamente el **Axioma 5** para demostrar la corrección de un ciclo con una cantidad no acotada a priori de iteraciones :(

Recap: Teorema del Invariante

- **Teorema.** Si def($\mathbb{B}$) y existe un predicado $\mathbb{I}$ tal que
  1. $\mathbb{P}_{\mathbb{C}} \Rightarrow \mathbb{I}$,
  2. $\{\mathbb{I} \wedge \mathbb{B}\}\ S\ \{\mathbb{I}\}$,
  3. $\mathbb{I} \wedge \neg\mathbb{B} \Rightarrow \mathbb{Q}_{\mathbb{C}}$,

  ... y **el ciclo termina**, entonces la siguiente tripla de Hoare es válida:

  $$\{\mathbb{P}_{\mathbb{C}}\}\ \textbf{while B do S endwhile}\ \{\mathbb{Q}_{\mathbb{C}}\}$$

- Esta observación es un teorema que se deduce de la definición anterior.
- Las condiciones 1-3 garantizan la corrección parcial del ciclo (la hipótesis de terminación es necesaria para garantizar corrección).

Ejemplo: suma de índices

- Sea la siguiente tripla de Hoare:

  $\{n \geq 0 \wedge j = 1 \wedge z = 0\}$

```
while (j ≤ n) do
  z := z + j;
  j := j + 1
endwhile
```

  $\{z = \sum_{k=1}^{n} k\}$

- Habíamos identificamos los predicados necesarios para aplicar el Teorema del Invariante:
  - $\mathbb{P_C} \equiv n \geq 0 \wedge j = 1 \wedge z = 0$
  - $\mathbb{Q_C} \equiv z = \sum_{k=1}^{n} k$
  - $\mathbb{B} \equiv j \leq n$
  - $\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
- Para tener a mano el programa del cuerpo del ciclo:
  - $\mathbb{S} \equiv z := z + j; j := j + 1$

Repaso: $\mathbb{P_C} \Rightarrow \mathbb{I}$

$\mathbb{P_C} \equiv n \geq 0 \wedge j = 1 \wedge z = 0$
$\mathbb{Q_C} \equiv z = \sum_{k=1}^{n} k$
$\mathbb{B} \equiv j \leq n$
$\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
$\mathbb{S} \equiv z := z + j; j := j + 1$

$$\begin{aligned}
\mathbb{P_C} &\equiv n \geq 0 \wedge j = 1 \wedge z = 0 \\
&\Rightarrow 1 \leq j \leq n+1 \wedge z = 0 \\
&\Rightarrow 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{0} k \\
&\Rightarrow 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k \\
&\equiv \mathbb{I} \checkmark
\end{aligned}$$

Repaso: $\mathbb{I} \wedge \neg\mathbb{B} \Rightarrow \mathbb{Q_C}$

$\mathbb{P_C} \equiv n \geq 0 \wedge j = 1 \wedge z = 0$
$\mathbb{Q_C} \equiv z = \sum_{k=1}^{n} k$
$\mathbb{B} \equiv j \leq n$
$\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
$\mathbb{S} \equiv z := z + j; j := j + 1$

$$\begin{aligned}
\mathbb{I} \wedge \neg\mathbb{B} &\equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k \wedge \neg(j \leq n) \\
&\equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k \wedge j > n \\
&\Rightarrow 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k \wedge j = n+1 \\
&\Rightarrow 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{n+1-1} k \wedge j = n+1 \\
&\Rightarrow z = \sum_{k=1}^{n} k \equiv Q_C \checkmark
\end{aligned}$$

$\{\mathbb{I} \wedge \mathbb{B}\}\ \mathbb{S}\ \{\mathbb{I}\}$

$\mathbb{P_C} \equiv n \geq 0 \wedge j = 1 \wedge z = 0$
$\mathbb{Q_C} \equiv z = \sum_{k=1}^{n} k$
$\mathbb{B} \equiv j \leq n$
$\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
$\mathbb{S} \equiv z := z + j; j := j + 1$

Alcanza con probar que:

$$\mathbb{I} \wedge \mathbb{B} \Rightarrow wp(\mathbb{S}, \mathbb{I})$$

$$\begin{aligned}
& wp(\texttt{z:=z+j;j:=j+1}, 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k) \\
\equiv\ & wp(\texttt{z:=z+j}, wp(\texttt{j:=j+1}, 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k)) \\
\equiv\ & wp(\texttt{z:=z+j}, def(j+1) \wedge_L (1 \leq j+1 \leq n+1 \wedge z = \sum_{k=1}^{j+1-1} k)) \\
\equiv\ & wp(\texttt{z:=z+j}, 1 \leq j+1 \leq n+1 \wedge z = \sum_{k=1}^{j+1-1} k) \\
\equiv\ & def(z+j) \wedge_L (1 \leq j+1 \leq n+1 \wedge z+j = \sum_{k=1}^{j+1-1} k)
\end{aligned}$$

$\{\mathbb{I} \wedge \mathbb{B}\}\ \mathbb{S}\ \{\mathbb{I}\}$ (cont.)

$\mathbb{P}_C \equiv n \geq 0 \wedge j = 1 \wedge z = 0$
$\mathbb{Q}_C \equiv z = \sum_{k=1}^{n} k$
$\mathbb{B} \equiv j \leq n$
$\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
$\mathbb{S} \equiv z := z + j; j := j + 1$

$$\equiv \quad 0 \leq j \leq n \wedge z + j = \sum_{k=1}^{j} k$$

$$\equiv \quad 0 \leq j \leq n \wedge z = (\sum_{k=1}^{j} k) - j$$

$$\equiv \quad 0 \leq j \leq n \wedge z = \sum_{k=1}^{j-1} k$$

- Luego de simplificar, nos falta probar que:

$$(\underbrace{1 \leq i \leq n+1 \wedge s = \sum_{k=1}^{i-1} k}_{\mathbb{I}} \wedge \underbrace{i \leq n}_{\mathbb{B}}) \Rightarrow \underbrace{(0 \leq i \leq n \wedge s = \sum_{k=1}^{i-1} k)}_{wp(\mathbb{S},\mathbb{I})}$$

- Lo cual es trivialmente cierto. Por lo tanto podemos concluir que $\{\mathbb{I} \wedge \mathbb{B}\}\ \mathbb{S}\ \{\mathbb{I}\}$ es una tripla de Hoare válida.

Ejemplo: suma de índices

- Habiendo probado las hipótesis del Teorema del Invariante podemos decir que si el ciclo siempre terminara, entonces la siguiente tripla de Hoare es válida:

  $\{n \geq 0 \wedge j = 1 \wedge z = 0\}$

```
while (j ≤ n) do
  z := z + j;
  j := j + 1
endwhile
```

  $\{z = \sum_{k=1}^{n} k\}$

- **Pero** ..., ¡todavía no probamos que el ciclo siempre termina!
- ¿Cómo podemos probar si dada una precondición, un ciclo siempre termina?
  - Para eso tenemos el Teorema de terminación.

Teorema de terminación de un ciclo

- **Teorema.** Sea $\mathbb{V}$ el producto cartesiano de los dominios de las variables del programa y sea $\mathbb{I}$ un invariante del ciclo **while B do S endwhile**. Si existe una función $fv : \mathbb{V} \to \mathbb{Z}$ tal que

  1. $\{\mathbb{I} \wedge \mathbb{B} \wedge V_0 = fv\}$ **S** $\{fv < V_0\}$,
  2. $\mathbb{I} \wedge fv \leq 0 \Rightarrow \neg\mathbb{B}$,

  ... entonces la ejecución del ciclo **while B do S endwhile** siempre termina.

- La función fv se llama función variante del ciclo.

- El Teorema de terminación nos permite demostrar si un ciclo termina (i.e. no se cuelga).

Ejemplo: Suma de índices

$\mathbb{P}_C \equiv n \geq 0 \wedge j = 1 \wedge z = 0$
$\mathbb{Q}_C \equiv z = \sum_{k=1}^{n} k$
$\mathbb{B} \equiv j \leq n$
$\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
$\mathbb{S} \equiv z := z + j; j := j + 1$

- Sea la siguiente tripla de Hoare:

  $\{n \geq 0 \wedge j = 1 \wedge z = 0\}$

```
while (j ≤ n) do
  z := z + j;
  j := j + 1
endwhile
```

  $\{z = \sum_{k=1}^{n} k\}$

- Ya probamos que el siguiente predicado es un invariante de este ciclo.

$$\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$$

- ¿Cúal sería una buena función variante para este ciclo?

```
while (j ≤ n) do
  z := z + j;
  j := j + 1
endwhile
```

- Ejecutemos el ciclo con $n = 6$.

Iteración	j	z	n	n + 1 - j
0	1	0	6	6
1	2	1	6	5
2	3	3	6	4
3	4	6	6	3
4	5	10	6	2
5	6	15	6	1
6	7	21	6	0

- Una función variante representa una cantidad que se va reduciendo a lo largo de las iteraciones y el ciclo termina cuando es menor o igual a cero.
- Proponemos entonces $fv = n + 1 - j$, que representa la cantidad de índices que falta sumar.

Ejemplo: Suma de índices

Teorema de terminación

$\mathbb{P_C} \equiv n \geq 0 \wedge j = 1 \wedge z = 0$
$\mathbb{Q_C} \equiv z = \sum_{k=1}^{n} k$
$\mathbb{B} \equiv j \leq n$
$\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
$\mathbb{S} \equiv z := z + j; j := j + 1$

1. Para verificar que $\{\mathbb{I} \wedge \mathbb{B} \wedge fv = V_0\}\ \mathbb{S}\ \{fv < V_0\}$ para todo V_0, calculamos $wp(\mathbb{S}, fv < V_0)$ con $fv = n + 1 - j$.

$$
\begin{aligned}
& wp(\texttt{z:=z+1;j:=j+1}, fv < V_0) \\
& \equiv wp(\texttt{z:=z+1;j:=j+1}, (n+1-j) < V_0) \\
& \equiv wp(\texttt{z:=z+1}, wp(\texttt{j:=j+1}, (n+1-j) < V_0)) \\
& \equiv wp(\texttt{z:=z+1}, def(j+1) \wedge_L (n+1-(j+1)) < V_0)) \\
& \equiv wp(\texttt{z:=z+1}, (n+1-(j+1)) < V_0)) \\
& \equiv def(s+1) \wedge_L n - j < V_0 \\
& \equiv n - j < V_0 \\
& \equiv n - j < n + 1 - j \\
& \equiv n - j < n - j + 1 \\
& \equiv 0 < 1 \checkmark
\end{aligned}
$$

Ejemplo: Suma de índices

Teorema de terminación

$\mathbb{P_C} \equiv n \geq 0 \wedge j = 1 \wedge z = 0$
$\mathbb{Q_C} \equiv z = \sum_{k=1}^{n} k$
$\mathbb{B} \equiv j \leq n$
$\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
$\mathbb{S} \equiv z := z + j; j := j + 1$

2. Verifiquemos que $\mathbb{I} \wedge fv \leq 0 \Rightarrow \neg\mathbb{B}$

$$
\begin{aligned}
\mathbb{I} \wedge fv \leq 0 \quad & \equiv \quad \overbrace{1 \leq j \leq n+1 \wedge s = \sum_{k=1}^{j-1} k}^{\mathbb{I}} \wedge \overbrace{n+1-j \leq 0}^{fv \leq 0} \\
& \Rightarrow \quad j \leq n+1 \wedge n+1-j \leq 0 \\
& \Rightarrow \quad j \leq n+1 \wedge n+1 \leq j \\
& \Rightarrow \quad j = n+1 \\
& \Rightarrow \quad \neg(j \leq n) \\
& \Rightarrow \quad \neg\mathbb{B} \checkmark
\end{aligned}
$$

Ejemplo: Suma de índices

Recapitulando, sean

- $\mathbb{I} \equiv 1 \leq j \leq n+1 \wedge z = \sum_{k=1}^{j-1} k$
- $fv = n + 1 - j$

Antes habíamos probado que el ciclo es **parcialmente** correcto dado que:

1. $\mathbb{P_C} \Rightarrow \mathbb{I}$
2. $\{\mathbb{I} \wedge \mathbb{B}\}\ \mathbb{S}\ \{\mathbb{I}\}$
3. $\mathbb{I} \wedge \neg\mathbb{B} \Rightarrow \mathbb{Q_C}$

Ahora acabamos de probar que el ciclo siempre termina ya que:

4. $\{\mathbb{I} \wedge \mathbb{B} \wedge V_0 = fv\}\ \mathbb{S}\ \{fv < V_0\}$,
5. $\mathbb{I} \wedge fv \leq 0 \Rightarrow \neg\mathbb{B}$,

Por lo tanto, por (1)-(5) tenemos (finalmente) que ...

Ejemplo: Suma de índices

- Que la siguiente tripla de Hoare:

  $\{\mathbb{P}_{\mathbb{C}} : n \geq 0 \land j = 1 \land z = 0\}$

  ```
  while (j ≤ n) do
    z := z + j;
    j := j + 1
  endwhile
  ```

  $\{\mathbb{Q}_{\mathbb{C}} : z = \sum_{k=1}^{n} k\}$

  ¡es una tripla de Hoare válida!

- Esto significa que:
  1. Si el ciclo comienza en un estado que cumple $\mathbb{P}_{\mathbb{C}}$
  2. ... entonces después de un número finito de pasos termina
  3. y además en un estado que cumple $\mathbb{Q}_{\mathbb{C}}$

Intervalo

¡A despejarse un poco!

Otro ejemplo: Chequeo de paridad

Sea una secuencia de booleans (*sec*), contar la cantidad de posiciones de la secuencia iguales a `true`.

Algunas propiedades de $\#apariciones$:

- $\#apariciones(\langle\rangle, true) = 0$
- $\#apariciones(concat(s, \langle e\rangle), true) =$
  $\#apariciones(s, true) + (\text{if } e = true \text{ then } 1 \text{ else } 0 \text{ fi})$

Otro ejemplo: Chequeo de paridad

- ¿Es válida la siguiente tripla de Hoare?
- $\{j = 0 \land c = 0\}$

  ```
  while ( j < |seq| ) do
     if sec[j]=true then
       c := c + 1
     else
       skip
     endif;
     j := j +1
  endwhile
  ```

  $\{c = \#apariciones(sec, true)\}$

Otro ejemplo: Chequeo de Paridad

- Para probar que se cumplen las condiciones del Teorema del Invariante tenemos que demostrar formalmente que se cumple:
  1. $\mathbb{P}_\mathbb{C} \Rightarrow \mathbb{I}$
  2. $\{\mathbb{I} \wedge \mathbb{B}\}\ \mathbb{S}\ \{\mathbb{I}\}$
  3. $\mathbb{I} \wedge \neg\mathbb{B} \Rightarrow \mathbb{Q}_\mathbb{C}$
- ¿Cuál es el predicado $\mathbb{P}_\mathbb{C}, \mathbb{Q}_\mathbb{C}, \mathbb{I}$ y $\mathbb{B}$?
  - $\mathbb{P}_\mathbb{C} \equiv j = 0 \wedge c = 0$
  - $\mathbb{Q}_\mathbb{C} \equiv c = \#apariciones(sec, true)$
  - $\mathbb{B} \equiv j < |sec|$
  - $\mathbb{I} \equiv 0 \leq j \leq |seq| \wedge_L c = \#apariciones(subseq(sec, 0, j), true)$

Al pizarrón

```
while ( j < |seq| ) do
   if sec[j]=true then
     c := c + 1
   else
     skip
   endif;
   j := j +1
endwhile
```

Teorema del Invariante

1. $\mathbb{P}_\mathbb{C} \Rightarrow \mathbb{I}$
2. $\{\mathbb{I} \wedge \mathbb{B}\}\ \mathbb{S}\ \{\mathbb{I}\}$
3. $\mathbb{I} \wedge \neg\mathbb{B} \Rightarrow \mathbb{Q}_\mathbb{C}$

- $\mathbb{P}_\mathbb{C} \equiv j = 0 \wedge c = 0$
- $\mathbb{Q}_\mathbb{C} \equiv c = \#apariciones(sec, true)$
- $\mathbb{B} \equiv j < |sec|$
- $\mathbb{I} \equiv 0 \leq j \leq |seq| \wedge_L c = \#apariciones(subseq(sec, 0, j), true)$

- $\#apariciones(\langle\rangle, true) = 0$
- $\#apariciones(concat(s, \langle e\rangle), true) = \#apariciones(s, true) + (\text{if } e = true \text{ then } 1 \text{ else } 0 \text{ fi})$

Hasta ahora

- Ya que probamos
  - $\mathbb{P}_\mathbb{C} \Rightarrow \mathbb{I}$
  - $\{\mathbb{I} \wedge \mathbb{B}\}\mathbb{S}\{\mathbb{I}\}$
  - $\mathbb{I} \wedge \neg\mathbb{B} \Rightarrow \mathbb{Q}_\mathbb{C}$
- usando el teorema del invariante pudimos probar que (si el ciclo termina), se cumple $\mathbb{Q}_\mathbb{C}$.
- Ya probamos que $\mathbb{I} \equiv 0 \leq j \leq |seq| \wedge_L c = \#apariciones(seq, true)$ es un invariante del ciclo.
- ¡Pero no probamos todavía que la ejecución del ciclo termina!

Función variante

- La función variante representa una cantidad que se va reduciendo.
- Pero... ¿Cuál la condición para que se detenga el ciclo?
  - $\neg\mathbb{B} \equiv \neg(j < |seq|)$
  - Necesitamos una función tal que cuando vale que $fv \leq 0$ implique $\neg(j < |seq|)$. Es decir:

$$fv \leq 0 \Rightarrow \neg(j < |seq|)$$

En busca de la función variante

```
while ( j < |seq| ) do
   if sec[j]=true then
     c := c + 1
   else
     skip
   endif;
   j := j +1
endwhile
```

- Veamos como evolucionan las variables que existen con los valores para $|seq| = 4$

Iterac.	$\|seq\|$	j	c	$fv = \|seq\| - j$
0	4	0	?	4-0=4
1	4	1	?	4-1=3
2	4	2	?	4-2=2
3	4	3	?	4-3=1
4	4	4	?	4-4=0

- Sea la siguiente función candidata a función variante:

$$fv = |seq| - j$$

Terminación del ciclo

- Con esta definición de fv veamos el ciclo termina, para eso hay que probar:
  1. $\{\mathbb{I} \wedge \mathbb{B} \wedge fv = V_0\}$ S $\{fv < V_0\}$
  2. $\mathbb{I} \wedge fv \leq 0 \Rightarrow \neg\mathbb{B}$

Al pizarrón (segunda parte)

```
while ( j < |seq| ) do
   if sec[j]=true then
     c := c + 1
   else
     skip
   endif;
   j := j +1
endwhile
```

Teorema del Invariante

1. $\mathbb{P}_\mathbb{C} \Rightarrow \mathbb{I}$
2. $\{\mathbb{I} \wedge \mathbb{B}\}\ \mathbb{S}\ \{\mathbb{I}\}$
3. $\mathbb{I} \wedge \neg\mathbb{B} \Rightarrow \mathbb{Q}_\mathbb{C}$

Teorema de Terminación

1. $\{\mathbb{I} \wedge \mathbb{B} \wedge fv = V_0\}$ S $\{fv < V_0\}$
2. $\mathbb{I} \wedge fv \leq 0 \Rightarrow \neg\mathbb{B}$

- $\mathbb{P}_\mathbb{C} \equiv j = 0 \wedge c = 0$
- $\mathbb{Q}_\mathbb{C} \equiv c = \#apariciones(sec, true)$
- $\mathbb{B} \equiv j < |sec|$
- $\mathbb{I} \equiv 0 \leq j \leq |seq| \wedge_L c = \#apariciones(subseq(sec, 0, j), true)$
- $fv = |seq| - j$

Otro ejemplo: Chequeo de Paridad

- Probamos que:
  1. $\mathbb{P}_\mathbb{C} \Rightarrow \mathbb{I}$
  2. $\{\mathbb{I} \wedge \mathbb{B}\}\mathbb{S}\{\mathbb{I}\}$
  3. $\mathbb{I} \wedge \neg\mathbb{B} \Rightarrow \mathbb{Q}_\mathbb{C}$
  4. $\{\mathbb{I} \wedge V_0 = fv\}\mathbb{S}\{fv < V_0\}$
  5. $\mathbb{I} \wedge fv \leq 0 \Rightarrow \neg\mathbb{B}$
- Entonces, por (1)-(5) , se cumplen las hipótesis de ambos teoremas (teorema del invariante + teorema de terminación)
- Por lo tanto, la tripla de Hoare es válida (i.e., dada $\mathbb{P}_\mathbb{C}$, el ciclo siempre termina y vale $\mathbb{Q}_\mathbb{C}$)

Recap #1: Teorema del invariante

- **Teorema.** Si def($\mathbb{B}$) y existe un predicado $\mathbb{I}$ tal que
  1. $\mathbb{P}_\mathbb{C} \Rightarrow \mathbb{I}$,
  2. $\{\mathbb{I} \land \mathbb{B}\}\, \mathbb{S}\, \{\mathbb{I}\}$,
  3. $\mathbb{I} \land \neg\mathbb{B} \Rightarrow \mathbb{Q}_\mathbb{C}$,

  ... y **el ciclo termina**, entonces la siguiente tripla de Hoare es válida:

  $$\{\mathbb{P}_\mathbb{C}\}\ \textbf{while B do S endwhile}\ \{\mathbb{Q}_\mathbb{C}\}$$

Recap #2: Teorema de terminación de un ciclo

- **Teorema.** Sea $\mathbb{V}$ el producto cartesiano de los dominios de las variables del programa y sea I un invariante del ciclo **while B do S endwhile**. Si existe una función $fv : \mathbb{V} \to \mathbb{Z}$ tal que
  1. $\{\mathbb{I} \land \mathbb{B} \land V_0 = fv\}\ \mathbb{S}\ \{fv < V_0\}$,
  2. $\mathbb{I} \land fv \leq 0 \Rightarrow \neg\mathbb{B}$,

  ... entonces la ejecución del ciclo **while B do S endwhile** SIEMPRE termina.
- La función fv se llama función variante del ciclo.

Teorema de corrección de un ciclo

- **Teorema.** Sean un predicado $\mathbb{I}$ y una función $fv : \mathbb{V} \to \mathbb{Z}$ (donde $\mathbb{V}$ es el producto cartesiano de los dominios de las variables del programa), y supongamos que $\mathbb{I} \Rightarrow$ def($\mathbb{B}$). Si
  1. $\mathbb{P}_\mathbb{C} \Rightarrow \mathbb{I}$,
  2. $\{\mathbb{I} \land \mathbb{B}\}\, \mathbb{S}\, \{\mathbb{I}\}$,
  3. $\mathbb{I} \land \neg\mathbb{B} \Rightarrow \mathbb{Q}_\mathbb{C}$,
  4. $\{\mathbb{I} \land \mathbb{B} \land V_0 = fv\}\ \mathbb{S}\ \{fv < V_0\}$,
  5. $\mathbb{I} \land fv \leq 0 \Rightarrow \neg\mathbb{B}$,

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

  $$\{\mathbb{P}_\mathbb{C}\}\ \textbf{while B do S endwhile}\ \{\mathbb{Q}_\mathbb{C}\}$$

Teorema de corrección de un ciclo

- El **teorema de corrección de un ciclo** nos permite demostrar la validez de una tripla de Hoare cuando el programa es un ciclo.
- Por definición, si probamos que:

  $$\{\mathbb{P}_\mathbb{C}\}\ \text{while B do S endwhile}\ \{\mathbb{Q}_\mathbb{C}\}$$

  ... entonces probamos que:

  $$\mathbb{P}_\mathbb{C} \Rightarrow wp(\texttt{while B do S endwhile}, \mathbb{Q}_\mathbb{C})$$

- **¡Cuidado!** Probar lo anterior no significa haber obtenido un predicado que caracteriza a la precondición más débil del ciclo:

  $$wp(\text{while B do S endwhile}, \mathbb{Q}_\mathbb{C})$$

Programas con ciclos

- En general, no se puede definir un mecanismo efectivo para obtener una fórmula cerrada que represente la precondición más débil de un ciclo.
- Entonces, ¿cómo hacemos para probar la corrección y terminación de un programa que incluye ciclos intercalados con otras instrucciones?

Guía para demostrar programas con ciclos

¿Qué tenemos que hacer para probar que

$$\{Pre\}\ \text{S1;while B do S endwhile;S3}\ \{Post\}$$

es válida?

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

Por monotonía, esto nos permite demostrar que

$$Pre \Rightarrow_L wp(\texttt{S1;while...;S3}, Post)$$

es verdadera.

Recap: SmallLang

- Para las demostraciones de corrección, introdujimos un lenguaje sencillo y con menos opciones (mucho más simple que C++). Llamemos **SmallLang** a este lenguaje.
- SmallLang tiene únicamente:
  - Nada: `skip`
  - Asignación: `x := E`
  - Secuencia: `S1;S2`
  - Condicional: `if B then S1 else S2 endif`
  - Ciclo: `while B do S endwhile`
- No posee memoria dinámica (punteros), aliasing, llamados a función, estructura `for`, etc.

C++ → SmallLang

Pero dado un programa en C++ podemos traducirlo a SmallLang preservando su semántica (comportamiento).
Por ejemplo:

Versión C++

```
for (int i =0; i < s.size(); i++) {
  if (s[i]==0) {
    s[i]++;
  }
}
```

Versión SmallLang

```
i:=0;
while (i < s.size()) do
  if (s[i]==0)
    s[i]:=s[i]+1
  else
    skip
  endif;
  i:=i+1
endwhile
```

Ambos programas tienen el mismo comportamiento.

Corrección de programas en C++

Para demostrar la corrección de un programa en C++ con respecto a una especificación, podemos:

1. Traducir el programa C++ a SmallLang preservando su comportamiento.
2. Demostrar la corrección del programa en SmallLang con respecto a la especificación.
3. Entonces, probamos la corrección del comportamiento del programa original.

Bibliografía

- David Gries - The Science of Programming
  - Part II - The Semantics of a Small Language
    - Chapter 11 - The Iterative Command