CRA. Tema 5: Semántica y verificación de programas (segunda parte)

José Enrique Morais San Miguel

Universidad de Alcalá

Un sistema de verificación formal: Hoare Logic

1) Se toman como axiomas todas las fórmulas verdaderas sobre los dominios a los que pertenecen las variables del programa
2) Axioma de asignación: $\vdash \{p(x)\{x \leftarrow t\}\}\,\mathtt{x:=t}\,\{p(x)\}$
3) Regla de composición:

$$\frac{\vdash \{p\}\ S_1\ \{q\} \quad \vdash \{q\}\ S_2\ \{r\}}{\vdash \{p\}\ S_1; S_2\ \{r\}}$$

4) Regla de la alternativa:

$$\frac{\vdash \{p \wedge B\}\ S_1\ \{q\} \quad \vdash \{p \wedge \neg B\} S_2 \{q\}}{\vdash \{p\}\ \texttt{si}\ \ B\ \texttt{entonces}\ \ S_1\ \texttt{si no}\ \ S_2\ \{q\}}$$

5) Regla del bucle:

$$\frac{\vdash \{p \wedge B\} S \{p\}}{\vdash \{p\}\ \texttt{mientras}\ \ B\ \texttt{hacer}\ \ S\{p \wedge \neg B\}}$$

6) Regla de la consecuencia

$$\frac{\vdash p_1 \rightarrow p \quad \vdash \{p\}\ S\ \{q\} \quad \vdash q \rightarrow q_1}{\vdash \{p_1\}\ S\ \{q_1\}}$$

Un sistema de verificación formal: Hoare Logic

La fórmula p en la regla del bucle se llama invariante y describe el comportamiento de una sola ejecución de S en el cuerpo de la expresión `mientras`. Para probar

$$\vdash \{p_0\}\, \texttt{mientras}\ B\, \texttt{hacer}\ S\, \{q\},$$

basta encontrar un invariante p. Si $p_0 \rightarrow p$ la precondición de la regla del bucle se verifica. Si además se puede demostrar que $(p \wedge \neg B) \rightarrow q$ es verdadera, no importa cuantas veces se ejecute el bucle, la postcondición de la regla implica la postcondición q. La dificultad en la prueba de la corrección parcial de programas radica en la elección de un invariante adecuado.

Un sistema de verificación formal: Hoare Logic

- ¿Cómo probar que un invariante es tal?
- Respuesta: Precondición más débil (Dijkstra)

Para probar que I es un invariante para un bucle

```
mientras B hacer S
```

basta probar que $wp(S, I) = I$ y esto se puede hacer aplicando las reglas de cálculo de precondiciones más débiles anteriores.

Un sistema de verificación formal: Hoare Logic

Probemos la corrección (parcial) de este programa que admite como entrada dos números naturales cualesquiera y devuelve el producto de los mismos:

$\{T\}$
```
x:=0;y:=b
mientras y ≠ 0 hacer
   x:= x+a; y:= y-1;
fmientras
```
$\{x = a \cdot b\}$

Un sistema de verificación formal: Hoare Logic

$\{T\}$
```
x:=0;y:=b
mientras y ≠0 hacer
   x:= x+a; y:= y-1;
fmientras
```
$\{x = a \cdot b\}$

Si encontramos un invariante I para el bucle:

$$\vdash \{I \wedge y \neq 0\}\ \mathtt{x := x + a; y := y - 1}\ \{I\}$$

y si, además, $(I \wedge y = 0) \rightarrow (x = a \cdot b)$, por la regla del bucle y la regla de la consecuencia, la corrección del programa estaría probada. Vimos la semana pasada que $x = (b - y) \cdot a$ es un invariante para el cuerpo del bucle y cumple las exigencias anteriores. De hecho, $I \wedge (y = 0)$ es igual a la postcondición.

Ejemplo 5.22.1. Cálculo de potencias

Probar la corrección (parcial) de:

$\{a > 0 \wedge b \geq 0\}$

```
    x:=a; y:= b; z:=1;
      mientras y ≠ 0 hacer
        si impar(y) entonces y:=y-1; z:= x*z
                    si no, x:=x*x; y=coc(y,2)
        fsi
      fmientras
```

$\{z = a^b\}$

Ejemplo 5.22.1. Cálculo de potencias

$\{a > 0 \wedge b \geq 0\}$

```
    x:=a; y:= b; z:=1;
      mientras y ≠ 0 hacer
         si impar(y) entonces y:=y-1; z:= x*z
                      si no, x:=x*x; y=coc(y,2)
         fsi
      fmientras
```

$\{z = a^b\}$

Ejemplo 5.22.1. Cálculo de potencias

$\{a > 0 \wedge b \geq 0\}$

```
x:=a; y:= b; z:=1;
  mientras y≠0 hacer
     si impar(y) entonces y:=y-1; z:= x*z
                    si no, x:=x*x; y=coc(y,2)
     fsi
  fmientras
```

$\{z = a^b\}$

Invariante: $I \equiv (z \cdot x^y = a^b)$

$$wp(W, I) = (impar(y) \wedge wp(\mathtt{y := y - 1; z := x * z}, I)) \vee$$

$$\vee (par(y) \wedge wp(\mathtt{x := x * x; y = y/2}, I)) =$$

$$= (impar(y) \wedge zxx^{y-1} = a^b) \vee (par(y) \wedge z(x^2)^{y/2} = a^b)$$

$$\equiv (impar(y) \wedge zx^y = a^b) \vee (par(y) \wedge zx^y = a^b) \equiv I$$

Ejemplo 5.22.2. Máximo común divisor

Probar la corrección (parcial) de:

$\{a > 0 \wedge b > 0\}$

```
    x:=a; y:= b
      mientras x≠y hacer
        si (x >y) entonces x:=x-y
                si no, y:=y-x
        fsi
      fmientras
```

$\{x = mcd(a, b)\}$

Ejemplo 5.22.2. Máximo común divisor

$\{a > 0 \wedge b > 0\}$

```
   x:=a; y:= b
     mientras x≠y hacer
       si (x >y) entonces x:=x-y
                 si no, y:=y-x
       fsi
     fmientras
```

$\{x = mcd(a, b)\}$

Ejemplo 5.22.2. Máximo común divisor

$\{a > 0 \wedge b > 0\}$

```
x:=a; y:= b
  mientras x≠y hacer
    si (x >y) entonces x:=x-y
                si no, y:=y-x
    fsi
  fmientras
```

$\{x = mcd(a,b)\}$

Invariante: $I \equiv (mcd(x,y) = mcd(a,b))$

$$wp(W,I) = ((x > y) \wedge wp(\mathrm{x} := \mathrm{x} - \mathrm{y}, I)) \vee$$

$$\vee((x \leq y) \wedge wp(\mathrm{y} := \mathrm{y} - \mathrm{x}, I)) =$$

$$= ((x > y) \wedge mcd(x-y,y) = mcd(a,b)) \vee$$

$$\vee((x \leq y) \wedge mcd(x,y-x) = mcd(a,b)) =$$

$$= ((x > y) \wedge mcd(x,y) = mcd(a,b)) \vee$$

$$\vee((x \leq y) \wedge mcd(x,y) = mcd(a,b)) \equiv I$$

Ejemplo 5.22.4. Inversión de listas

Probar la corrección (parcial) de:

$\{true\}$

```
    L:=a; L1:= nil
      mientras L≠nil hacer
         L1:= cons(hd(L),L1); L:=tl(L)
      fmientras
```

$\{L1 = reverse(a)\}$

Ejemplo 5.22.4. Inversión de listas

$\{true\}$

```
   L:=a; L1:= nil
     mientras L≠nil hacer
       L1:= cons(hd(L),L1); L:=tl(L)
     fmientras
```

$\{L1 = reverse(a)\}$

Ejemplo 5.22.4. Inversión de listas

$\{true\}$
```
    L:=a; L1:= nil
      mientras L≠nil hacer
        L1:= cons(hd(L),L1); L:=tl(L)
      fmientras
```
$\{L1 = reverse(a)\}$

Invariante: $I \equiv \texttt{append}(reverse(L), L1) = reverse(a)$

$$wp(W, I) = wp(\texttt{L1} := \texttt{cons}(\texttt{hd}(\texttt{L}), \texttt{L1}); \texttt{L} := \texttt{tl}(\texttt{L}), I) =$$

$$= (\texttt{append}(reverse(\texttt{tl}(L)), \texttt{cons}(\texttt{hd}(\texttt{L}), \texttt{L1})) = reverse(a)) \equiv$$

$$\equiv (\texttt{append}(reverse(L), L1) = reverse(a)) = I$$

Hemos aplicado que

$$append(reverse(tl(L)), cons(hd(L), L1)) = append(reverse(L), L1).$$

Ejemplo 5.22.5. Algoritmo de Euclides extendido

Probar la corrección (parcial) de:

$\{a > 0;\ b \geq 0\}$

$$\mathtt{S:=}\begin{pmatrix} a & b \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$\mathtt{mientras}\ \mathtt{s}_{12} \neq 0\ \mathtt{hacer}$

$$\mathtt{q:=coc(s_{11},s_{12});}\quad \mathtt{Q:=}\begin{pmatrix} 0 & 1 \\ 1 & -\mathtt{q} \end{pmatrix}\mathtt{;}\quad \mathtt{S:=S\star Q}$$

$\mathtt{fmientras}$

$\{s_{21}a + s_{31}b = s_{11} \wedge s_{11} = mcd(a,b)\}$

Ejemplo 5.22.5. Algoritmo de Euclides extendido

$\{a > 0;\ b \geq 0\}$

$$\mathtt{S:=}\begin{pmatrix} a & b \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

```
mientras s12 ≠ 0 hacer
    q:=coc(s11,s12); Q:=( 0  1 ; 1  -q ); S:=S*Q
fmientras
```

$$\mathtt{q:=coc(s_{11},s_{12});}\quad \mathtt{Q:=}\begin{pmatrix} 0 & 1 \\ 1 & -\mathtt{q} \end{pmatrix};\quad \mathtt{S:=S\star Q}$$

$\{s_{21}a + s_{31}b = s_{11} \wedge s_{11} = mcd(a, b)\}$

Observación

La asignación $\mathtt{S:=S\star Q}$ en el algoritmo puede escribirse como

$$\mathtt{S} := \begin{pmatrix} s_{12} & s_{11} - q \cdot s_{12} \\ s_{22} & s_{21} - q \cdot s_{22} \\ s_{32} & s_{31} - q \cdot s_{32} \end{pmatrix}$$

Ejemplo 5.22.5. Algoritmo de Euclides extendido

$\{a > 0;\ b \geq 0\}$

$$\mathtt{S:=}\begin{pmatrix} a & b \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

`mientras` $\mathtt{s}_{12} \neq 0$ `hacer`

$$\mathtt{q:=coc(s_{11},s_{12});}\quad \mathtt{Q:=}\begin{pmatrix} 0 & 1 \\ 1 & -\mathtt{q} \end{pmatrix}\mathtt{;}\quad \mathtt{S:=S\star Q}$$

`fmientras`

$\{s_{21}a + s_{31}b = s_{11} \ \wedge\ s_{11} = mcd(a,b)\}$

Ejemplo 5.22.5. Algoritmo de Euclides extendido

$\{a > 0;\ b \geq 0\}$

$$\texttt{S:=} \begin{pmatrix} a & b \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

```
mientras s12 ≠ 0 hacer
    q:=coc(s11,s12); Q:=( 0  1 ; 1  -q ); S:=S*Q
fmientras
```

$\{s_{21}a + s_{31}b = s_{11} \wedge s_{11} = mcd(a,b)\}$

Invariante:

$$\{s_{21}a + s_{31}b = s_{11} \ \wedge\ s_{22}a + s_{32}b = s_{12} \wedge mcd(s_{11}, s_{12}) = mcd(a,b)\}$$

$$wp(W, I) = \{(s_{22}a + s_{32}b = s_{12}) \wedge$$

$$\wedge((s_{21} - qs_{22})a + (s_{31} - qs_{32})b = s_{11} - qs_{12})$$

$$\wedge\ (mcd(s_{21}, s_{11} - qs_{12}) = mcd(a,b))\} = I$$