Verificación formal

Facultad de Informática

October 7, 2024

Indice

Tabla de contenidos

Introducción
Conceptos

- Un estado es el cjto. de valores que toman el conjunto de variables que conforman la estructura de datos del sistema.
- Las aserciones o asertos son sentencias lógicas que hacen al estado del sistema.
- Las sentencias encerradas en llaves $\{\}$ son asertos.

Un programa es un conjunto de sentencias que transforman un estado inicial en uno final.
Estado:

- inicial: previo a la ejecución de un código
- final: posterior a la ejecución de un código
- del sistema: valores de las variables en cada momento.

Representación formal
Ternas de Hoare

Terna de Hoare:

$$\{P\}\ C\ \{Q\}$$

- C es una parte del código, $\{P\}$ se denomina precondiciones de C y $\{Q\}$ se denomina postcondiciones
- Significa que P son predicados lógicos que deben cumplirse para que el código C funcione.
- El código comienza con un estado válido en P y el programa termina con un estado válido para Q.
  - Ejemplo 1:$\{y \neq 0\} \quad x = 1/y \quad \{x = 1/y\}$.
  - Ejemplo 2: $\{\ \} \quad a = b \quad \{a = b\}$.

Asertos
Ejemplos de asertos

Si se ha demostrado $\{P\}\, C\, \{Q\}$, entonces:

- Si $P_1 \Rightarrow P$;
  - entonces $\{P_1\}\, C\, \{Q\}$ es verdadero;
  - y además se **refuerzan** las precondiciones.
- Si $Q \Rightarrow Q_1$
  - entonces $\{P\}\, C\, \{Q_1\}$ es verdadero;
  - y además se **debilitan** las poscondiciones.

ejemplo

$i > 1$ es más fuerte que $i > 0$

todos los estados que satisfacen $i > 1$ también satisfacen $i > 0$

$$i > 1 \Rightarrow i > 0.$$

- Más fuerte (más selectivo, más específico).
- Más débil (menos selectivo, más general).
- Aserción más débil es $\{\}$
- Aserción más fuerte es false, (ningún estado satisface la cond).

Terna de Hoare
Axiomas

$\{A\}\, P\, \{B\}$

- Axioma SKIP o salto
  - $SKIP\, \{P\}$
- Axioma de la asignación
  - $\vdash \{P[E/V]\}\, V := E\, \{P\}$

ejemplo asignación

$\{\}\, a := b\, \{a = b\}$

Terna de Hoare
Reglas

- Fortalecimiento de la precondición

$$\frac{P_1 \Rightarrow P \quad \{P\}\, C\, \{Q\}}{\{P_1\}\, C\, \{Q\}}$$

- Ejemplo: supongamos que la terna de Hoare es correcta:
  $\{y \neq 0\}$ x=1/y $\{x = 1/y\}$
  demostrar que también lo es $\{y = 4\}$ x=1/y $\{x = 1/y\}$

Ejemplo1

$$\frac{\{y = 4\} \Rightarrow \{y \neq 0\} \quad \{y \neq 0\}\, x = 1/y\, \{x = 1/y\}}{\{y = 4\}\, x = 1/y\, \{x = 1/y\}}$$

Ejemplo2

$$\frac{\{P\} \Rightarrow \{\} \quad \{\}\, a = b\, \{a = b\}}{\{P\}\, a = b\, \{a = b\}}$$

Terna de Hoare
Reglas

Fortalecimiento de la precondición

$$\frac{P \Rightarrow P_1 \quad \{P_1\}\, C\, \{Q\}}{\{P\}\, C\, \{Q\}}$$

- $$\frac{\vdash P \Rightarrow P,\ \vdash \{P_1\} C \{Q\}}{\vdash \{P\} C \{Q\}}$$

Si $P \Rightarrow P_1$
y se ha demostrado $\{P_1\}\, C\, \{Q\}$ entonces
$\{P\}\, C\, \{Q\}$ es verdadero y
además se refuerza el aserto

Reglas
Debilitamiento1

- Debilitamiento de la poscondición
  - $$\frac{\vdash \{P\} C \{Q\},\ \vdash Q \Rightarrow Q_1}{\vdash \{P\} C \{Q_1\}}$$

    - Ejemplo: la terna de Hoare es correcta:
      $\{\}$ max = b $\{max = b\}$ también lo es
      $\{\}$ max = b $\{max \geq b\}$

$$\frac{\{max = b\} \Rightarrow \{max \geq b\} \quad \{\}\ \text{max} = \text{b}\ \{max = b\}}{\{\}\ \text{max} = \text{b}\ \{max \geq b\}}$$

Reglas
Debilitamiento2

- Debilitamiento de la poscondición
  - $\dfrac{\vdash \{P\} C \{Q\},\ \vdash Q \Rightarrow Q_1}{\vdash \{P\} C \{Q_1\}}$

$$\frac{Q \Rightarrow Q_1 \quad \{P\}\ C\ \{Q\}}{\{P\}\ C\ \{Q_1\}}$$

Si *(Q implica a Q_1)* $Q \Rightarrow Q_1$
y se ha demostrado $\{P\}\ C\ \{Q\}$ entonces
$\{P\}\ C\ \{Q_1\}$ es verdadero y
además se debilita el aserto

Reglas
Conjunción y disyunción

- Regla de la conjunción
  - $$\frac{\vdash\{P_1\}C\{Q_1\},\ \vdash\{P_2\}C\{Q_2\}}{\vdash\{P_1 \wedge P_2\}C\{Q_1 \wedge Q_2\}}$$
- Regla de la disyunción
  - $$\frac{\vdash\{P_1\}C\{Q_1\},\ \vdash\{P_2\}C\{Q_2\}}{\vdash\{P_1 \vee P_2\}C\{Q_1 \vee Q_2\}}$$

Reglas
De la conjunción 2

- Conjunción
  - $$\frac{\vdash \{P_1\} C \{Q_1\},\ \vdash \{P_2\} C \{Q_2\}}{\vdash \{P_1 \wedge P_2\} C \{Q_1 \wedge Q_2\}}$$

caso general

$$\{P_1\}\ C\ \{Q_1\}$$
$$\{P_2\}\ C\ \{Q_2\}$$

$$\{P_1 \wedge P_2\}\ C\ \{Q_1 \wedge Q_2\}$$

caso particular

$$\{\}\ C\ \{Q_1\}$$
$$\{P\}\ C\ \{Q_2\}$$

$$\{P\}\ C\ \{Q_1 \wedge Q_2\}$$

Reglas
De la disyunción 2

- Disyunción
  - $$\frac{\vdash \{P_1\} C \{Q_1\},\ \vdash \{P_2\} C \{Q_2\}}{\vdash \{P_1 \vee P_2\} C \{Q_1 \vee Q_2\}}$$

caso general

$$\{P_1\}\ C\ \{Q_1\}$$
$$\{P_2\}\ C\ \{Q_2\}$$

$$\{P_1 \vee P_2\}\ C\ \{Q_1 \vee Q_2\}$$

caso particular

$$\{\}\ C\ \{Q_1\}$$
$$\{P\}\ C\ \{Q_2\}$$

$$\{\}\ C\ \{Q_1 \vee Q_2\}$$

Reglas
Regla de la secuencia

- Regla de la secuencia
  - $\frac{\vdash \{P\} C_1 \{Q_1\}, \quad \vdash \{Q_1\} C_2 \{Q_2\}}{\vdash \{P\} C_1; C_2 \{Q_2\}}$

- Regla de la secuencia general
  - $\frac{\vdash \{P\} C_1 \{Q_1\}, \quad \vdash \{Q_1\} C_2 \{Q_2\}, \ldots, \vdash \{Q_n - 1\} C_n - 1 \{Q_n\}}{\vdash \{P\} C_1; C_2; \ldots; C_n - 1 \{Q_n\}}$

Tabla de contenidos

Estructura de control
Secuencia

- $$\frac{\vdash\{P\}C_1\{Q_1\},\ \vdash\{Q_1\}C_2\{Q_2\},....,\vdash\{Q_n-1\}C_n\{Q_n\}}{\vdash\{P\}C_1;C_2;...;C_n\{Q_n\}}$$

$$\{P\}\ C_1\ \{Q_1\}$$
$$\{Q_1\}\ C_2\ \{Q_2\}$$
$$\{Q_2\}\ C_3\ \{Q_3\}$$
$$.....$$
$$\{Q_n - 1\}\ C_n\ \{Q_n\}$$

$$\{P\}\ C_1; C_2; ....; C_n\ \{Q_n\}$$

Nomenclatura
Variables

- Representación del súbíndice: mediante subíndices se indica si las variables representan valores iniciales o finales.
  - $\{a_\omega = b_\alpha\} \wedge \{b_\omega = a_\alpha\}$
  - ω representa el estado final y α el estado inicial.
- Representación de las variables ocultas: Aparecen o no en el código y se introducen para almacenar los valores iniciales de ciertas posiciones de memoria.
  - $\{a = A, b = B\}\, h = a; a = b; b = h\, \{a = B, b = A\}$

Corrección de un código

Secuencia

Demostrar la corrección de código

- Se parte de la poscondición final (las condiciones que deben satisfacer los resultados).
- A partir de ahí se deduce la precondición.
- El código se verifica **en sentido contrario a como se ejecuta.**

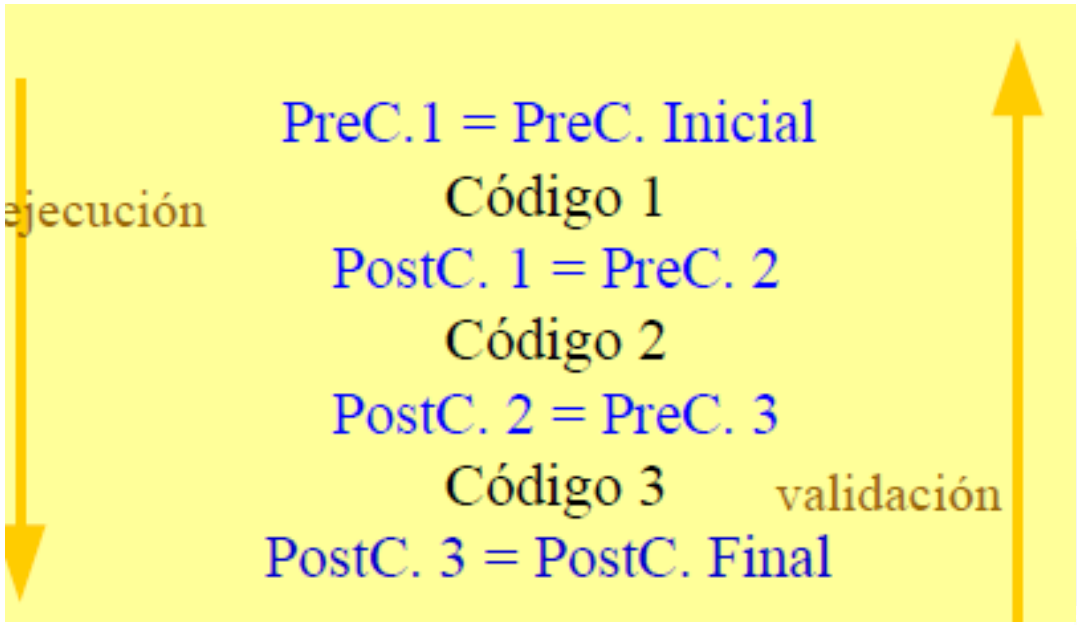
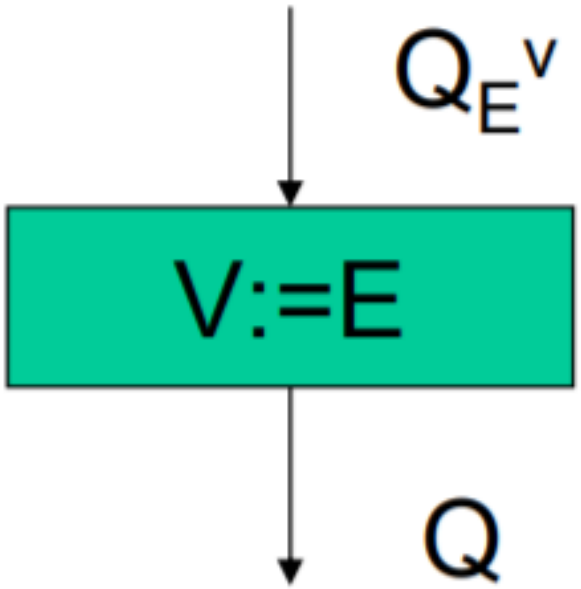


Figure: Orden de validación

Reglas
Asignación

La regla de asignación requiere que las variables implicadas no compartan el mismo espacio de memoria.

- Las sentencias de asignación son sentencias de la forma $V = E$, en donde V es una variable y E es una expresión.
  $\{\}\ V = E\ \{V = E_\alpha\}$



centering

Reglas
Asignación2

Regla de la asignación: Sea C una sentencia de la forma $V = E$, y con la poscondición $\{Q\}$, entonces la precondición de C puede hallarse sustituyendo todos los caso de V en Q por E.

$$\{\}\ V = E\ \{V = E_\alpha\}$$

$$\{P\}\ V = E\ \{Q\}$$
$$\{P\} = \{Q_E{}^V\} \Rightarrow \{V = E_\alpha, Q\}$$
$$\{Q_E{}^V\}\ V = E\ \{Q\}$$

- Ejemplo: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\ i = 2 * i\ \{i < 6\}$
  - $\{Q_E{}^V\}\ \{i_\omega = 2 * i_\alpha, i_\omega < 6\} \Rightarrow \{i_\alpha < 3\}$

Pre y poscondiciones
Ejemplos

- Ejemplo 1: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, j = i + 1\, \{j > 0\}$

Pre y poscondiciones
Ejemplos

- Ejemplo 1: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, j = i + 1\, \{j > 0\}$
  - $\{Q_E{}^V\}\, \{j_\omega = i_\alpha + 1, j_\omega > 0\} \Rightarrow \{i_\alpha + 1 > 0\}$

Pre y poscondiciones
Ejemplos

- Ejemplo 1: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, j = i + 1\, \{j > 0\}$
  - $\{Q_E{}^V\}\, \{j_\omega = i_\alpha + 1, j_\omega > 0\} \Rightarrow \{i_\alpha + 1 > 0\}$
- Ejemplo 2: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, y = x^2\, \{y > 1\}$

Pre y poscondiciones
Ejemplos

- Ejemplo 1: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, j = i + 1\, \{j > 0\}$
  - $\{Q_E{}^V\}\, \{j_\omega = i_\alpha + 1, j_\omega > 0\} \Rightarrow \{i_\alpha + 1 > 0\}$
- Ejemplo 2: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, y = x^2\, \{y > 1\}$
  - $\{Q_E{}^V\}\, \{y_\omega = x_\alpha * x_\alpha, x_\omega > 1\} \Rightarrow \{x_\alpha{}^2 > 1\}$

Pre y poscondiciones
Ejemplos

- Ejemplo 1: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, j = i + 1\, \{j > 0\}$
  - $\{Q_E{}^V\}\, \{j_\omega = i_\alpha + 1, j_\omega > 0\} \Rightarrow \{i_\alpha + 1 > 0\}$
- Ejemplo 2: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, y = x^2\, \{y > 1\}$
  - $\{Q_E{}^V\}\, \{y_\omega = x_\alpha * x_\alpha, x_\omega > 1\} \Rightarrow \{x_\alpha{}^2 > 1\}$
- Ejemplo 3: Determinar la poscondición para que la terna siguiente sea correcta: $\{x > 2\}\, x = x^2\, \{Q\}$

Pre y poscondiciones
Ejemplos

- Ejemplo 1: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, j = i + 1\, \{j > 0\}$
  - $\{Q_E{}^V\}\, \{j_\omega = i_\alpha + 1, j_\omega > 0\} \Rightarrow \{i_\alpha + 1 > 0\}$
- Ejemplo 2: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, y = x^2\, \{y > 1\}$
  - $\{Q_E{}^V\}\, \{y_\omega = x_\alpha * x_\alpha, x_\omega > 1\} \Rightarrow \{{x_\alpha}^2 > 1\}$
- Ejemplo 3: Determinar la poscondición para que la terna siguiente sea correcta: $\{x > 2\}\, x = x^2\, \{Q\}$
  - $\{Q\} \Rightarrow \{x_\alpha > 2, x_\omega = {x_\alpha}^2\} \Rightarrow \{x_\omega > 4\}$

Pre y poscondiciones
Ejemplos

- Ejemplo 1: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, j = i + 1\, \{j > 0\}$
  - $\{Q_E{}^V\}\, \{j_\omega = i_\alpha + 1, j_\omega > 0\} \Rightarrow \{i_\alpha + 1 > 0\}$
- Ejemplo 2: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, y = x^2\, \{y > 1\}$
  - $\{Q_E{}^V\}\, \{y_\omega = x_\alpha * x_\alpha, x_\omega > 1\} \Rightarrow \{{x_\alpha}^2 > 1\}$
- Ejemplo 3: Determinar la poscondición para que la terna siguiente sea correcta: $\{x > 2\}\, x = x^2\, \{Q\}$
  - $\{Q\} \Rightarrow \{x_\alpha > 2, x_\omega = {x_\alpha}^2\} \Rightarrow \{x_\omega > 4\}$
- Ejemplo 4: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, x = 1/x\, \{x \geq 0\}$

Pre y poscondiciones
Ejemplos

- Ejemplo 1: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, j = i + 1\, \{j > 0\}$
  - $\{Q_E{}^V\}\, \{j_\omega = i_\alpha + 1, j_\omega > 0\} \Rightarrow \{i_\alpha + 1 > 0\}$
- Ejemplo 2: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, y = x^2\, \{y > 1\}$
  - $\{Q_E{}^V\}\, \{y_\omega = x_\alpha * x_\alpha, x_\omega > 1\} \Rightarrow \{{x_\alpha}^2 > 1\}$
- Ejemplo 3: Determinar la poscondición para que la terna siguiente sea correcta: $\{x > 2\}\, x = x^2\, \{Q\}$
  - $\{Q\} \Rightarrow \{x_\alpha > 2, x_\omega = {x_\alpha}^2\} \Rightarrow \{x_\omega > 4\}$
- Ejemplo 4: Determinar la precondición para que la terna siguiente sea correcta: $\{P\}\, x = 1/x\, \{x \geq 0\}$
  - $\{Q\} = \{x_\omega = 1/x_\alpha, x_\omega \geq 0\} \Rightarrow \{x_\alpha > 0\}$

Reglas
Alternativa simple

$C1$ es parte de un programa y B una condición, entonces la sentencia se forma *if* B *then* C_1'

- Posibles estados de una sentencia *if* : Si el estado inicial satisface B además de P entonces se ejecutará C_1 , demostrar que $\{P \wedge B\} \Rightarrow \{Q\}$ es correcto.
- Si el estado inicial no satisface B , demostrar que $\{P \wedge \neg B\} \Rightarrow \{Q\}$ es correcto.

$$\frac{\{P \wedge B\}\ C_1\ \{Q\} \qquad \{P \wedge \neg B\} \Rightarrow \{Q\}}{\{P\}\ \text{if B then}\ C_1\ \{Q\}}$$

Ejemplo
Alternativa simple

Ejemplo: Demostrar que: $\{\}$ if max<a then max=a $\{(max \geq a)\}$

Si $\{\neg(max < a)\} \Rightarrow \{max \geq a)\}$
Si $\{(max < a) max = a\{\ max \geq a)\}$
$P = \{max = a, (max \geq a)\} \Rightarrow \{a \geq a\}$
$\{a \geq a\} \Rightarrow \{\}$

$$\frac{\{P \wedge B\}\ C_1\ \{Q\} \qquad \{P \wedge \neg B\} \Rightarrow \{Q\}}{\{P\}\ \text{if B then}\ C_1\ \{Q\}}$$

Reglas
Alternativa con else

Sentencia **if** con precondición $\{P\}$ y poscondición $\{Q\}$:

- Si el estado inicial satisface B además de P entonces se ejecutará C_1, demostrar que $\{P \wedge B\}\ C_1\ \{Q\}$ es correcto.1
- Si el estado inicial no satisface B demostrar $\{P \wedge \neg B\} \Rightarrow \{Q\}$ es correcto.

$$\frac{\{P \wedge B\}\ C_1\ \{Q\} \qquad \{P \wedge \neg B\}\ C_2\ \{Q\}}{\{P\}\ \text{if B then } C_1 \text{ else } C_2\ \{Q\}}$$

Alternativas

Precondición a partir de la poscondición

Cuando se trabaja con la estructura 'if'es bastante fácil obtener la poscondición a partir de la precondición.

Sin embargo normalmente lo que se necesita es lo contrario.

- Regla de inferencia:

$$\frac{\{P \wedge B\}\ C_1\ \{Q\} \qquad \{P \wedge \neg B\}\ C_2\ \{Q\}}{\{P\}\ \text{if B then } C_1 \text{ else } C_2\ \{Q\}}$$

Reglas
Regla de la alternativa

- Regla del IF then
  - $$\frac{\vdash\{P \vee B\} C \{Q\},\ \vdash\{P \vee \neg B\} \Rightarrow Q}{\vdash\{P\} \textit{if } B \textit{ then } C \{Q\}}$$
- Regla del if then else
  - $$\frac{\vdash\{P \vee B\} C_1 \{Q\},\ \vdash\{P \vee \neg B\} C_2 \Rightarrow Q}{\vdash\{P\} \textit{if } B \textit{ then } C_1 \textit{ else } C_2 \{Q\}}$$

Reglas
Regla de Repetitiva

- Regla del While
  - $$\frac{\vdash \{P \wedge B\}\, C\, \{P\}}{\vdash \{P\}\, \textit{while } B \textit{ do } C\, \{P \vee \neg B\}}$$

- Siendo P una invariante, esto significa:
  - P es cierto antes de la ejecución del programa
  - P es cierto durante la ejecución del programa
  - P es cierto después de la ejecución del programa

$$\frac{\{P \wedge B\}\, C\, \{P\}}{\{P\}\ \text{while B do}\ C\ \{(\neg B \wedge P\}}$$