

TALLER 4.

VERIFICACIÓ FORMAL DE PROGRAMES.

2. Sigui $E = \{ \exists i: 0 \leq i < n: (b[0..n-1] \leq b[i]) \wedge (b[i] > b[0..i-1]) \}$ Calcula:

a) $((E)_c^b)_w^n$ b) $((E)_c^b)_k^i$

a) $E = \{ \exists i: 0 \leq i < n: (b[0..n-1] \leq b[i]) \wedge (b[i] > b[0..i-1]) \} =$
 $= \{ \exists i: 0 \leq i < n: (c[0..n-1] \leq c[i]) \wedge (c[i] > c[0..i-1]) \} =$
 $= \{ \exists i: 0 \leq i < w: (b[0..w-1] \leq b[i]) \wedge (b[i] > b[0..i-1]) \}$

b) $E = \{ \exists i: 0 \leq i < n: (b[0..n-1] \leq b[i]) \wedge (b[i] > b[0..i-1]) \} =$
 $= \{ \exists i: 0 \leq i < n: (c[0..n-1] \leq c[i]) \wedge (c[i] > c[0..i-1]) \} =$
 no se puede sustituir 'i' porque es una variable ligada

3. Sigui $E = \{(x < y) \wedge a \wedge (y \leq z) \vee (x \leq y)\}$. Calcula:

a) $((E)_y^x)_4^z$ b) $E_{y,4,6}^{x,y,z}$

a) $E = \{(y < x) \wedge a \wedge (y \leq z) \vee (y \leq y)\} = \{(4 < 4) \wedge a \wedge (4 \leq z) \vee (4 \leq 4)\} = \{(4 < 4) \wedge a \wedge (4 \leq 6) \vee (4 \leq 4)\}$
 $= \{F \vee (4 \leq 4)\} = \{4 \leq 4\} = T$

b) $E = \{(y < 4) \wedge a \wedge (4 \leq 6) \vee (y \leq 4)\} = \{(y < 4) \wedge a \wedge T \vee (y \leq 4)\} = \{(y < 4) \wedge a \vee (y \leq 4)\}$

5. Demuestra la llei de distributivitat de disjunció per un programa no determinista: $wp(S, Q) \vee wp(S, R) \Rightarrow wp(S, Q \vee R)$

- Demostración:

Si tenemos un dados de seis caras, no se puede asegurar que número saldrá en el dado, es decir:

$wp(\text{lanzar}, \text{seis}) = F$ y $wp(\text{lanzar}, \text{cinco}) = F \dots wp(\text{lanzar}, \text{uno}) = F$

Pero sí que está garantizado que saldrá alguna cara del dado

$wp(\text{lanzar}, \text{seis} \vee \dots \vee \text{uno}) = T$

Por lo que

$wp(\text{anzar}, \text{seis}) \vee \dots \vee wp(\text{anzar}, \text{uno}) \Rightarrow wp(\text{lanzar}, \text{seis} \vee \dots \vee \text{uno})$

$P \in \text{LHS} \Rightarrow p \in \text{RHS} \quad P \in \text{LHS} \Rightarrow Q \Rightarrow R \Rightarrow Q \wedge R \quad Q \vee R \vee (Q \wedge R) = Q \vee R$

6. Demuestra que si un programa S és determinista, llavors:

$$wp(S, Q) \vee wp(S, R) = wp(S, Q \vee R)$$

- En $wp(S, Q)$, sabem que el programa empieza por S y da como resultado seguro Q, en $wp(S, R)$ tenemos que empieza por S y da como resultado R, por lo que podemos extraer que el programa siempre va a empezar por S, y por el enunciado sabemos que debe acabar en Q o en R sí o sí, es decir, el resultado del programa puede ser tanto Q como R, $wp(S, Q \vee R)$. Demostración con ejemplo:

$$wp(S, Q) \vee wp(S, R) = wp(S, Q \vee R)$$

$$Wp(\text{sortir el sol, si}) = T, wp(\text{sortir el sol, No}) = F$$

Hi ha garantia de que demà sortirà el sol

S=sortir el sol

Q=si

R=no

$$Wp(\text{sortir el sol, si}) \vee wp(\text{sortir el sol, no}) = wp(\text{sortir el sol, si} \vee \text{no})$$

7. Demuestra la llei de monotonicitat emprant sols la llei de distributivitat de conjunció:

$$\text{Si } Q \Rightarrow R, \text{ llavors } wp(S, Q) \Rightarrow wp(S, R)$$

- Sabemos que si Q implica R, significa que el resultado Q implicará a R entonces $wp(S, Q) \Rightarrow wp(S, R)$. Demostración:

$$\text{Si } Q \rightarrow R, \text{ llavors } wp(S, Q) \rightarrow wp(S, R)$$

Consideram la part esquerra com a s (LHS):

$$Q \rightarrow R, \text{ llavors } Q \rightarrow R \text{ per tant es True}$$

Consideram la part de la dreta com a s

Consideram la part de la dreta com a s (RHS):

$$Q \rightarrow R, \text{ llavors } Q \rightarrow R \text{ per tant es true}$$

8. Demuestra a partir de la llei de distributivitat de conjunció que:

$$wp(S, R) \wedge wp(S, \neg R) = F$$

- Sabemos que por una parte S tiene que dar R como resultado, y por la otra parte dice lo contrario, que no debe dar R. Por lo que: $wp(S, R) \wedge wp(S, \neg R) = wp(S, R \wedge \neg R) = wp(S, F) = F$, ya que, por la ley de contradicción $R \wedge \neg R = F$. Demostración:

$$wp(S, R) \wedge wp(S, \neg R) = F$$

$$wp(S, R \wedge \neg R) = wp(S, R) \wedge wp(S, \neg R)$$

$$wp(S, R \wedge \neg R) = F$$

Consideram la part de la esquerra com a s:

$R \wedge \neg R = F$ se cumple la condición de contradicción.

10. És correcta la següent especificació?. $\{Q: x < 3 \wedge y = 0\} \ x := 6; y := x + 1 \ \{R: x > 3 \wedge y > x\}$
Demuestra formalment la resposta.

- Primero se calcula $wp(S, R)$:

$$wp(S, R) = wp(S1; S2, R) = wp(x := 6; y := x + 1, x > 3 \wedge y > x) =$$

$$= wp(x = 6, wp(y := x + 1, x > 3 \wedge y > x)) =$$

$$= wp(x = 6, (x > 3 \wedge y > x)_{x+1}) =$$

$$= wp(x = 6, (x > 3 \wedge x + 1 > x)) =$$

$$= (x > 3 \wedge x + 1 > x)^x_6 = 6 > 3 \wedge 6 + 1 > 6 = T \wedge T = T$$

- Segundo $[Q \Rightarrow wp(S, R)] = T$
 $Q \Rightarrow wp(S1; S2, R)$
 $[Q \Rightarrow T] = T$
 $[Q \Rightarrow T] = \neg Q \vee T = T$

11. Calcula $wp(x:=K; y:=M, \{R: x > y\})$, on K i M són dues constants.

$$\begin{aligned} wp(S1; S2, R) &= wp(x:=K, wp(y:=M, x > y)) = \\ &= wp(x:=K, (x > y)^y_M) = \\ &= wp(x:=K, (x > M)) = \\ &= wp(x:=K, (x > M)^x_K) = K > M \end{aligned}$$

12. Demuestra formalment que les dues especificacions següents són correctes:

a. $\{T\} x:=X; y:=Y; x:=y \{R: x=y\}$

- Es T, ya que el resultado del problema $x=y$ es verdadera, y en el programa se asigna $x:=y$, por lo que x e y valen lo mismo.

b. $\{x > y\} x:=X; y:=Y; x:=y \{R: x=y\}$ on X, Y són dues constants.

- Serà T, siempre y cuando x sea > que y, ya que luego se cumplirá la asignación de $x:=y$.

13. És vàlida la següent especificació: $\{F\} S \{R\}$? Com interpretes el resultat?

14. Sigui S: $x:=y$. Demuestra que $\{y=Y\} S \{x=Y\}$ és una especificació vàlida, és a dir, és una tautologia.

15. Calcula $wp(t:=x; x:=y; y:=t, x=X \wedge y=Y)$.

$$\begin{aligned} wp(S1; S2; S3, R) &= wp(t:=x; x:=y; y:=t, x=X \wedge y=Y) = \\ &= wp(t:=x; x:=y, wp(y:=t, x=X \wedge y=Y)) = \\ &= wp(t:=x; x:=y, (x=X \wedge y=Y)^y_t) = \\ &= wp(t:=x; x:=y, (x=X \wedge t=Y)) = \\ &= wp(t:=x, wp(x:=y, (x=X \wedge t=Y))) = \\ &= wp(t:=x, (x=X \wedge t=Y)^x_y) = \\ &= wp(t:=x, y=X \wedge t=Y) = (y=X \wedge t=Y)^t_x = (y=X \wedge x=Y) \end{aligned}$$

16. A en Joan li donen una quantitat de doblers X, deriva formalment emprant el wp quina quantitat de doblers hem de donar a na Lluïsa per què tenguim les tres quartes parts de doblers que li han donat a en Joan.

S1: Joan = X
 S2: Lluïsa = Y

{R: Lluisa = $\frac{3}{4}$ Joan}

$$\begin{aligned}
 wp(S1;S2,R) &= wp(S1;wp(S2,R)) = \\
 &= wp(\text{Joan} = x; wp(\text{Lluisa} = y, \text{Lluisa} = \tfrac{3}{4} \text{Joan})) = \\
 &= wp(\text{Joan} = x; (\text{Lluisa} = \tfrac{3}{4} \text{Joan})^{Lluisa_y}) = \\
 &= wp(\text{Joan} = x, y = \tfrac{3}{4} \text{Joan}) = \\
 &= (y = \tfrac{3}{4} \text{Joan})^{Joan_x} \\
 y &= \tfrac{3}{4} * X
 \end{aligned}$$

17. En Joan té inicialment una certa quantitat de doblers i na Lluisa en té la meitat que en Joan. Si posteriorment a en Joan li donen una determinada quantitat de doblers, deriva formalment la quantitat de doblers que hauran de donar a na Lluisa per què finalment tengui les tres quartes parts dels doblers que té en Joan.

$$\begin{aligned}
 S1: \text{Joan} &= x + \alpha \\
 S2: \text{Lluisa} &= x/2 + \beta \\
 \{R: \text{Lluisa} &= \tfrac{3}{4} \text{Joan}\} \\
 wp(S1;S2,R) &= wp(S1; wp(S2,R)) = \\
 &= wp(\text{Joan} = x + \alpha; wp(\text{Lluisa} = x/2 + \beta, \text{Lluisa} = \tfrac{3}{4} \text{Joan})) = \\
 &= wp(\text{Joan} = x + \alpha; (\text{Lluisa} = \tfrac{3}{4} \text{Joan})^{Lluisa_{x/2+\beta}}) = \\
 &= wp(\text{Joan} = x + \alpha, x/2 + \beta = \tfrac{3}{4} \text{Joan}) = \\
 &= (x/2 + \beta = \tfrac{3}{4} \text{Joan})^{Joan_{x+\alpha}} = \\
 &= x/2 + \beta = \tfrac{3}{4} * (x + \alpha) \\
 \beta &= \tfrac{1}{4} * x + \tfrac{3}{4} * \alpha
 \end{aligned}$$

21. Demuestra pas a pas que la següent definició referent a la comanda composta: $wp(S1;S2, R) = wp(S1, wp(S2,R))$ satisfà les lleis del miracle exclòs i la distributivitat de conjunció.

$$wp(S1;S2, R) =$$

22. Construeix un programa, S, que calculi el major de dos nombres naturals x, y, i guardi el resultat en una variable z, i fes un esquema de demostració del mateix.

```

{Q} S {R}

if x ≥ y then
    z := x
else
    z := y

{x=X, y=Y}
{x=X; y=Y ∧ x ≥ y}
{z=X; y=Y ∧ x ≥ y}
{x=X; y=Y ∧ y > x}
{x=X; z=Y ∧ y > x}

{R: z = max(x,y)}

```

23. Verifica formalment el programa de la qüestió anterior.

$$wp(S1;S2,R) = wp(x=X; y=Y, z = \max(x,y)) =$$

24. Calcula i simplifica al màxim: $wp(S, x \leq y)$, on S:

```

if x > y → x, y := y, x
x ≤ y → skip
fi

```

$$\begin{aligned}
wp(if, R) &= BB \wedge [(B1 \rightarrow wp(S1, R)) \wedge (B2 \rightarrow wp(S2, R))] = \\
&= [(x > y) \vee (x \leq y)] \wedge [(x > y) \rightarrow wp(x, y := y, x, y \geq x)] \wedge [(x \leq y) \rightarrow wp(skip, y \geq x)] = \\
&= T \wedge [(x > y) \rightarrow (y \geq x)] \wedge [(x \leq y) \rightarrow (y \geq x)] = T \wedge T \wedge T = T
\end{aligned}$$

26. Demuestra formalment, pas a pas, la validesa de la següent especificació:

{a, b > 0}
if (a ≥ b) ∧ (b = 1) → z := a;
(a < b) ∧ (a = 1) → z := b;
(a ≥ b) ∧ (b ≠ 1) → z := a * b;
(a < b) ∧ (a ≠ 1) → z := a * b;
fi
{R: z = a * b}

$$\begin{aligned}
wp(IF, R) &= BB \wedge [(B1 \rightarrow wp(S1, R)) \wedge (B2 \rightarrow wp(S2, R)) \wedge (B3 \rightarrow wp(S3, R)) \wedge \\
&(B4 \rightarrow wp(S4, R))] = \\
&= [a \geq b \vee a < b] \wedge [\{(a \geq b \wedge b = 1) \rightarrow wp(z := a, z = a * b)\} \wedge \{(a < b \wedge a = 1) \rightarrow wp(z := b, z = a * b)\} \wedge \\
&\quad \wedge \{(a \geq b \wedge b \neq 1) \rightarrow wp(z := a * b, z = a * b)\} \wedge \{(a < b \wedge a \neq 1) \rightarrow wp(z := a * b, z = a * b)\}] = \\
&= T \wedge [\{(a \geq b \wedge b = 1) \rightarrow (z = a * b)^z_a\} \wedge \{(a < b \wedge a = 1) \rightarrow (z = a * b)^z_b\} \wedge \\
&\{(a \geq b \wedge b \neq 1) \rightarrow (z = a * b)^z_{a*b}\} \wedge \\
&\quad \wedge \{(a < b \wedge a \neq 1) \rightarrow (z = a * b)^z_{a*b}\}] = \\
&= T \wedge [\{(a \geq b \wedge b = 1) \rightarrow (a = a * b)\} \wedge \{(a < b \wedge a = 1) \rightarrow (b = a * b)\} \wedge \{(a \geq b \wedge b \neq 1) \rightarrow (a * b = a * b)\} \\
&\quad \wedge \\
&\quad \wedge \{(a < b \wedge a \neq 1) \rightarrow (a * b = a * b)\}] = \\
&= T \wedge [(a = a * 1) \wedge (b = 1 * b) \wedge (a * b = a * b) \wedge (a * b = a * b)] = T \wedge T \wedge T \wedge T \wedge T = T
\end{aligned}$$

27. Calcula wp(IF, R) del següent programa:

if x ≥ 0 → z := x
x < 0 → z := -x
fi
{R: z = abs(x)}

$$\begin{aligned}
wp(IF, R) &= BB \wedge [(B1 \rightarrow wp(S1, R)) \wedge (B2 \rightarrow wp(S2, R))] = \\
&= [(x \geq 0) \vee (x < 0)] \wedge [(x \geq 0 \rightarrow wp(z := x, z = abs(x))) \wedge (x < 0 \rightarrow wp(z := -x, z = abs(x)))] = \\
&= T \wedge [(x \geq 0 \rightarrow (z = abs(x))^z_x) \wedge (x < 0 \rightarrow (z = abs(x))^z_{-x})] = \\
&= [(x \geq 0 \rightarrow x = abs(x)) \wedge (x < 0 \rightarrow -x = abs(x))] = T \wedge T = T
\end{aligned}$$