DEPARTAMENTO DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

Trabajo práctico 1

Especificación y WP

21/4/2024 AED

Grupo IATOGYSWWBKAFJVCRWKR

Integrante	LU	Correo electrónico
Calo, Agustín	390/23	caloagustin4@gmail.com
Seri, Rafael Nicolás	362/23	rafaelnicoseri@gmail.com
Pintos Oliveira, Sol María Marcela	428/23	solpintosoliveira@gmail.com
Páez Torrico, Santiago	713/23	santiagopaez122@gmail.com

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Ciudad Universitaria - (Pabellón I/Planta Baja)
Intendente Güiraldes 2610 - C1428EGA
Ciudad Autónoma de Buenos Aires - Rep. Argentina
Tel/Fax: (++54 +11) 4576-3300
http://www.exactas.uba.ar

1. Especificación

1.1. redistribucionDeLosFrutos

`proc` `redistribucionDeLosFrutos` (in recursos: $seq\langle\mathbb{R}\rangle$, in cooperan: $seq\langle\mathsf{Bool}\rangle$) : $seq\langle\mathbb{R}\rangle$

  requiere $\{|recursos| = |cooperan|\}$
  requiere $\{todosPositivos(recursos)\}$
  asegura $\{|res| = |recursos|\}$
  asegura $\{(\forall i : \mathbb{Z})\ (0 \leq i < |res| \longrightarrow_L \text{ if } cooperan[i] \text{ then } res[i] = totalARepartir(recursos, cooperan) \text{ else } res[i] = recursos[i] + totalARepartir(recursos, cooperan) \text{ fi})\}$

`aux` `totalARepartir` (recursos: $seq\langle\mathbb{R}\rangle$, cooperan: $seq\langle\mathsf{Bool}\rangle$) : $\mathbb{R}$ =
$(\sum_{i=0}^{|recursos|-1} \text{if } cooperan[i] \text{ then } recursos[i] \text{ else } 0 \text{ fi})/|recursos|$;

1.2. trayectoriaDeLosFrutosIndividualesALargoPlazo

`proc` `trayectoriaDeLosFrutosIndividualesALargoPlazo` (inout trayectorias: $seq\langle seq\langle\mathbb{R}\rangle\rangle$, in cooperan: $seq\langle\mathsf{Bool}\rangle$, in apuestas: $seq\langle seq\langle\mathbb{R}\rangle\rangle$, in pagos: $seq\langle seq\langle\mathbb{R}\rangle\rangle$, in eventos: $seq\langle seq\langle\mathbb{R}\rangle\rangle$)

  requiere $\{trayectorias = old(trayectorias)\}$
  requiere $\{|cooperan| = |pagos| = |apuestas| = |eventos| = |trayectorias|\}$
  requiere $\{(\forall i : \mathbb{Z})\ (0 \leq i < |pagos| \longrightarrow_L todosPositivos(pagos[i]) \land todosPositivos(apuestas[i]))\}$
  requiere $\{(\forall i : \mathbb{Z})\ (0 \leq i < |trayectoria| \longrightarrow_L trayectoria[i][0] > 0)\}$
  requiere $\{(\forall j : \mathbb{Z})\ (-1 < j < |apuestas| \longrightarrow_L \sum_{i=0}^{|eventos[j]|-1} apuestas[j][i] = 1)\}$
  asegura $\{|trayectorias| = |old(trayectorias)|\}$
  asegura $\{(\forall i : \mathbb{Z})\ (0 \leq i < |old(trayectorias)| \longrightarrow_L |trayectorias[i]| = |old(trayectorias)[i]| + |eventos[i]|)\}$
  asegura $\{(\forall i : \mathbb{Z})\ (0 \leq i < |old(trayectorias)| \longrightarrow_L trayectorias[i][0] = old(trayectorias)[i][0])\}$
  asegura $\{(\forall i : \mathbb{Z})\ (0 \leq i < |old(trayectorias)| \longrightarrow_L (\forall j : \mathbb{Z})\ (0 \leq j < |eventos[i]| \longrightarrow_L trayectorias[i][j+1] = \text{if } cooperan[i] \text{ then } distribucion(aporteIndividual(trayectorias, apuestas, pagos, eventos, cooperan, i, j)) \text{ else } aporteIndividual(trayectorias, apuestas, pagos, eventos, i, j) + distribucion(trayectorias, apuestas, pagos, eventos, cooperan, j) \text{ fi}))\}$

`aux` `distribucion` (trayectorias: $seq\langle seq\langle\mathbb{R}\rangle\rangle$, apuestas: $seq\langle seq\langle\mathbb{R}\rangle\rangle$, pagos: $seq\langle seq\langle\mathbb{R}\rangle\rangle$, eventos: $seq\langle seq\langle\mathbb{N}\rangle\rangle$, cooperan: $seq\langle Bool\rangle$, m: $\mathbb{N}$) : $\mathbb{R}$ =
$(\sum_{k=0}^{|cooperan|-1} \text{if } cooperan[k] \text{ then } aporteIndividual(trayectorias,\ apuestas,\ pagos,\ eventos,\ k,\ m) \text{ else } 0 \text{ fi})/|cooperan|$;

1.3. trayectoriaExtrañaEscalera

`proc` `trayectoriaExtrañaEscalera` (in trayectorias: $seq\langle\mathbb{R}\rangle$) : Bool

  requiere $\{|trayectoria| > 0\}$
  asegura $\{res = True \iff |trayectoria| = 1 \lor (trayectoria[0] > trayectoria[1] \land \neg maximoLocal(trayectoria) \land trayectoria[|trayectoria|-1] < trayectoria[|trayectoria|-2]) \lor (trayectoria[|trayectoria|-1] > trayectoria[|trayectoria|-2] \land \neg maximoLocal(trayectoria) \land trayectoria[0] < trayectoria[1]) \lor (\exists i : \mathbb{Z})\ (0 < i < |trayectoria|-1 \land_L (trayectoria[i] > trayectoria[i+1] \land trayectoria[i] > trayectoria[i-1]) \land (\forall i : \mathbb{Z})\ (0 < j < |trayectoria|-1 \land_L (trayectoria[j] > trayectoria[j+1] \land trayectoria[j] > trayectoria[j-1]) \longrightarrow_L j = i))\}$

`pred` `maximoLocal` (s: $seq\langle\mathbb{R}\rangle$) {
  $(\exists i : \mathbb{Z})\ (0 < i < |s| - 1 \land_L (s[i] > s[i+1] \land s[i] > s[i-1]))$
}

1.4. individuoDecideSiCooperarONo

`proc` `individuoDecideSiCooperarONo` (in individuo: $\mathbb{N}$, in recursos: $seq\langle\mathbb{R}\rangle$, inout cooperan: $seq\langle\mathsf{Bool}\rangle$, in apuestas: $seq\langle seq\langle\mathbb{R}\rangle\rangle$, in pagos: $seq\langle seq\langle\mathbb{R}\rangle\rangle$, in eventos: $seq\langle seq\langle\mathbb{N}\rangle\rangle$)

  requiere $\{cooperan = old(cooperan)\}$
  requiere $\{|cooperan| = |recursos| = |apuestas| = |pagos| = |eventos|\}$
  requiere $\{(\forall i : \mathbb{Z})\ (0 \leq i < |apuestas| \longrightarrow_L todosPositivos(recursos) \land todosPositivos(apuestas[i]) \land todosPositivos(pagos[i]))\}$
  requiere $\{0 \leq individuo < |cooperan|\}$
  asegura $\{|cooperan| = |old(cooperan)|\}$
  asegura $\{(\forall i : \mathbb{Z})\ (0 \leq individuo < |cooperan| \land i \neq individuo \longrightarrow_L cooperan[i] = old(cooperan)[i])\}$
  asegura $\{(\exists s, p : seq\langle seq\langle\mathbb{R}\rangle\rangle)\ (|s| = |p| = |cooperan| \land (\forall n : \mathbb{Z})\ (0 \leq n < |s| \land_L |s[n]| = |p[n]| = (|eventos| + 1) \land s[n][0] = p[n][0] = recursos[n] \land (\forall k : \mathbb{Z})\ (0 < k < |s[n]| \longrightarrow_L s[n][k] = (\text{if } old(cooperan)[k] \lor k = individuo \text{ then } 0 \text{ else } aporteIndividual(s, apuestas, pagos, eventos, n, k) \text{ fi}) + distribucionCoop(s, apuestas, pagos, eventos, old(cooperan), k, individuo) \land p[n][k] = (\text{if } \neg(old(cooperan)[k]) \lor k = individuo \text{ then } aporteIndividual(p, apuestas, pagos, eventos, n, k)$

else 0 fi) + $distribucionNoCoop(p, apuestas, pagos, eventos, old(cooperan), k, individuo))) \wedge_L cooperan[individuo] = p[individuo][|p[individuo]| - 1] \leq s[individuo][|s[individuo]| - 1])\}$

`aux distribucionCoop` (trayectorias: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, apuestas: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, pagos: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, eventos: $seq\langle seq\langle \mathbb{N}\rangle\rangle$, cooperan: $seq\langle \mathsf{Bool}\rangle$, m: $\mathbb{N}$, individuo: $\mathbb{N}$) : $\mathbb{R}$ =
$(\sum_{k=0}^{|cooperan|-1}$ if $cooperan[k] \vee k = individuo$ then $aporteIndividual(trayectorias, apuestas, pagos, eventos, k, m)$ else 0 fi$)/|cooperan|$;

`aux distribucionNoCoop` (trayectorias: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, apuestas: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, pagos: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, eventos: $seq\langle seq\langle \mathbb{N}\rangle\rangle$, cooperan: $seq\langle \mathsf{Bool}\rangle$, m: $\mathbb{N}$, individuo: $\mathbb{N}$) : $\mathbb{R}$ =
$(\sum_{k=0}^{|cooperan|-1}$ if $cooperan[k] \wedge k \neq individuo$ then $aporteIndividual(trayectorias, apuestas, pagos, eventos, k, m)$ else 0 fi$)/|cooperan|$;

1.5. individuoActualizaApuesta

`proc individuoActualizaApuesta` (in individuo: $\mathbb{N}$, in recursos: $seq\langle \mathbb{R}\rangle$, in cooperan: $seq\langle \mathsf{Bool}\rangle$, inout apuestas: $seq\langle seq\langle \mathsf{Bool}\rangle\rangle$, in pagos: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, in eventos: $seq\langle seq\langle \mathbb{N}\rangle\rangle$)

`requiere` $\{apuestas = old(apuestas)\}$
`requiere` $\{|cooperan| = |recursos| = |apuestas| = |pagos| = |eventos|\}$
`requiere` $\{0 \leq individuo < |cooperan|\}$
`asegura` $\{|apuestas| = |old(apuestas)|\}$
`asegura` $\{(\forall i : \mathbb{Z})\ (-1 < i < |apuestas| \longrightarrow_L |apuestas[i]| = |old(apuestas)[i]|)\}$
`asegura` $\{(\exists p, s : seq\langle seq\langle \mathbb{R}\rangle\rangle)(\exists mejorApuesta : seq\langle \mathbb{R}\rangle)((\forall posibleApuesta : seq\langle \mathbb{R}\rangle)\ (ultElem(p, recursos, old(apuestas), pagos, cooperan, eventos, individuo, mejorApuesta) \geq ultElem(s, recursos, old(apuestas), pagos, cooperan, eventos, individuo, posibleApuesta) \longrightarrow_L apuestas[individuo] = mejorApuesta))\}$

`aux ultElem` (t: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, recursos: $seq\langle \mathbb{R}\rangle$, apuestas: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, pagos: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, cooperan: $seq\langle Bool\rangle$, eventos: $seq\langle \mathbb{N}\rangle$, individuo: $\mathbb{N}$, posibleApuesta: $seq\langle \mathbb{N}\rangle$) : $\mathbb{R}$ =
if $trayectoriaPosible(t, recursos, apuestas, pagos, cooperan, eventos, individuo, posibleApuesta)$ then $t[individuo][|t[individuo]| - 1]$ else -1 fi ;

`pred trayectoriaPosible` (t: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, recursos: $seq\langle \mathbb{R}\rangle$, apuestas: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, pagos: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, cooperan: $seq\langle Bool\rangle$, eventos: $seq\langle \mathbb{N}\rangle$, individuo: $\mathbb{N}$, posibleApuesta: $seq\langle \mathbb{N}\rangle$) {
$|posibleApuesta| = |apuestas[individuo]| \wedge sumElem(posibleApuesta) = 1 \wedge todosPositivos(posibleApuesta)|t| = |cooperan| \wedge (\forall i : \mathbb{Z})\ (-1 < i < |t| \wedge_L |t[i]| = (|eventos| + 1) \wedge t[i][0] = recursos[i] \wedge (\forall j : \mathbb{Z})\ (0 \leq j < |t[i]| \longrightarrow_L t[i][j + 1] =$ (if $cooperan[i]$ then 0 else $aporteIndDiferido(t[i], apuestas[i], pagos[i], eventos[i], i, j, individuo, posibleApuesta)$ fi) $+ distribucionDiferida(t[i], apuestas[i], pagos[i], eventos[i], cooperan, i, j, individuo, posibleApuesta)))$
}

`aux aporteIndDiferido` (trayectoria: $seq\langle \mathbb{R}\rangle$, apuestas: $seq\langle \mathbb{R}\rangle$, pagos: $seq\langle \mathbb{R}\rangle$, eventos: $seq\langle \mathbb{N}\rangle$, k: $\mathbb{N}$, m: $\mathbb{N}$, individuo: $\mathbb{N}$, apuestaInd: $seq\langle \mathbb{R}\rangle$) : $\mathbb{R}$ =
if $k = individuo$ then $trayectorias[m] \cdot apuestaInd[eventos[m]] \cdot pagos[eventos[m]]$ else $trayectorias[m] \cdot apuestas[eventos[m]] \cdot pagos[eventos[m]]$ fi ;

`aux distribucionDiferida` (trayectoria: $seq\langle \mathbb{R}\rangle$, apuestas: $seq\langle \mathbb{R}\rangle$, pagos: $seq\langle \mathbb{R}\rangle$, eventos: $seq\langle \mathbb{N}\rangle$, cooperan: $seq\langle Bool\rangle$, k: $\mathbb{N}$, m: $\mathbb{N}$, individuo: $\mathbb{N}$, apuestaInd: $seq\langle \mathbb{R}\rangle$) : $\mathbb{R}$ =
$(\sum_{k=0}^{|cooperan|-1}$ if $cooperan[k]$ then $aporteIndDiferido(trayectorias, apuestas, pagos, eventos, k, m, individuo)$ else 0 fi$)/|cooperan|$;

`aux sumElem` (s: $seq\langle \mathbb{R}\rangle$) : $\mathbb{R}$ =
$\sum_{i=0}^{|s|-1} s[i]$;

Auxiliares y predicados globales

`pred todosPositivos` (s: $seq\langle \mathbb{R}\rangle$) {
$(\forall i : \mathbb{Z})\ (0 \leq i < |s| \longrightarrow_L s[i] > 0)$
}

`aux aporteIndividual` (trayectorias: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, apuestas: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, pagos: $seq\langle seq\langle \mathbb{R}\rangle\rangle$, eventos: $seq\langle seq\langle \mathbb{N}\rangle\rangle$, k: $\mathbb{N}$, m: $\mathbb{N}$) : $\mathbb{R}$ = $trayectorias[k][m] \cdot apuestas[k][eventos[k][m]] \cdot pagos[k][eventos[k][m]]$;

2. Demostraciones de correctitud

Demostrar que la siguiente especificación es correcta respecto de su implementación.

La función **frutoDelTrabajoPuramenteIndividual** calcula, para el ejemplo de apuestas al juego de cara o seca, cuánto se ganaría si se juega completamente solo. Se contempla que el evento True es cuando sale cara.

`proc frutoDelTrabajoPuramenteIndividual` (in recurso: $\mathbb{R}$, in apuesta: $\langle s : \mathbb{R}, c : \mathbb{R}\rangle$, in pago: $\langle s : \mathbb{R}, c : \mathbb{R}\rangle$, in eventos: $seq\langle \mathsf{Bool}\rangle$, out res: $\mathbb{R}$)

`requiere` $\{apuesta_c + apuesta_s = 1 \wedge pago_c > 0 \wedge pago_s > 0 \wedge apuesta_c > 0 \wedge apuesta_s > 0 \wedge recurso > 0\}$
`asegura` $\{res = recurso(apuesta_c pago_c)^{\#apariciones(eventos,T)}(apuesta_s pago_s)^{\#apariciones(eventos,F)}\}$

Donde #apariciones(eventos, T) es el auxiliar utilizado en la teórica, y #(eventos, T) es su abreviación.

```
res := recurso
i := 0
while (i < |eventos|) do
    if eventos[i] then
        res := (res * apuesta.c) * pago.c
    else
        res := (res * apuesta.s) * pago.s
    endif
    i := i + 1
endwhile
```

Decimos que un programa S es correcto respecto de una especificación dada por una precondición P y una postcondición Q, si siempre que el programa comienza en un estado que cumple P, el programa termina su ejecución, y en el estado final se cumple Q. Se denota con la siguiente tripla de Hoare:

$$\{P\}S\{Q\}$$

Como el programa que nos dan cuenta con ciclos, tenemos que probar que $\{Pre\}S1; while...; S3\{Post\}$ es válida, para ello tenemos que cumplir:

1. $Pre \longrightarrow_L wp(S1, P_c)$
2. $P_c \longrightarrow_L wp(while..., Q_c)$
3. $Q_c \longrightarrow_L wp(S3, Post)$

Por monotonía, esto nos permite demostrar que $Pre \longrightarrow_L wp(S1; while...; S3, Post)$ es verdadera.
Vamos a demostrarlas, para ello proponemos lo siguiente:

- $P_c \equiv \{i = 0 \land res = recurso \land apuesta_c + apuesta_s = 1 \land pago_c > 0 \land pago_s > 0 \land apuesta_c > 0 \land apuesta_s > 0 \land recurso > 0\}$
- $Q_c \equiv \{res = recurso * \prod_{j=0}^{|eventos|-1} \textsf{if } eventos[j] \textsf{ then } apuesta.c * pago.c \textsf{ else } apuesta.s * pago.s \textsf{ fi}\}$
- $I \equiv \{0 \leq i \leq |eventos| \land res = recurso * \prod_{j=0}^{i-1} \textsf{if } eventos[j] \textsf{ then } apuesta.c * pago.c \textsf{ else } apuesta.s * pago.s \textsf{ fi}\}$
- $fv \equiv \{|eventos| - i\}$
- $B \equiv \{i < |eventos|\}$

Equivalencias entre Q_c propuesto y *asegura* dado:

$asegura : res = recurso * (apuesta_c * pago_c)^{\#apariciones(eventos,T)} (apuesta_s * pago_s)^{\#apariciones(eventos,F)} \equiv recurso * (apuesta_c * pago_c)^{\sum_{i=0}^{|eventos|-1} \textsf{if } eventos[i]=T \textsf{ then } 1 \textsf{ else } 0 \textsf{ fi}} * (apuesta_s * pago_s)^{\sum_{i=0}^{|eventos|-1} \textsf{if } eventos[i]=F \textsf{ then } 1 \textsf{ else } 0 \textsf{ fi}}$
Por propiedad de potenciación: $x^{f(n)+f(m)} = x^{f(n)} * x^{f(m)}$ luego $x^{\sum_{i=0}^{n} f(i)} = \prod_{i=0}^{n} x^{f(i)}$
$\equiv recurso * \prod_{j=0}^{|eventos|-1} \textsf{if } eventos[j] = T \textsf{ then } (apuesta_c * pago_c) \textsf{ else } 1 \textsf{ fi} * \prod_{j=0}^{|eventos|-1} \textsf{if } eventos[j] = F \textsf{ then } (apuesta_s * pago_s) \textsf{ else } 1 \textsf{ fi}$
Si $A = \{0 \leq j < |eventos| - 1 : eventos[j] = T\}$ y $B = \{0 \leq j < |eventos| - 1 : eventos[j] = F\}$ tengo que $A \cap B = \varnothing$ y como en las productorias el predicado del else es 1 (neutro multiplicativo), vale que si las juntamos queda:
$asegura : res = recurso * \prod_{j=0}^{|eventos|-1} \textsf{if } eventos[j] \textsf{ then } apuesta_c * pago_c \textsf{ else } apuesta_s * pago_s \textsf{ fi}$ ■

Demostración $Pre \longrightarrow_L wp(S1, P_c)$

Vamos a utilizar los axiomas 1 (asignación) y 2 (composicional) vistos en la teórica.
1. $Pre \longrightarrow_L wp(res := recurso; i := 0, P_c) \overset{Axioma\ 2}{\equiv} wp(res := recurso; wp(i := 0, P_c))$
2. $wp(i := 0, \{i = 0 \land res = recurso \land apuesta_c + apuesta_s = 1 \land pago_c > 0 \land pago_s > 0 \land apuesta_c > 0 \land apuesta_s > 0 \land recurso > 0\}) \overset{Axioma\ 1}{\equiv} def(0) \land_L 0 = 0 \land res = recurso \land apuesta_c + apuesta_s = 1 \land pago_c > 0 \land pago_s > 0 \land apuesta_c > 0 \land apuesta_s > 0 \land recurso > 0 \equiv res = recurso \land apuesta_c + apuesta_s = 1 \land pago_c > 0 \land pago_s > 0 \land apuesta_c > 0 \land apuesta_s > 0 \land recurso > 0 \equiv E_1$
3. $wp(res := recurso, E_1) \overset{Axioma\ 1}{\equiv} def(recurso) \land_L recurso = recurso \land apuesta_c + apuesta_s = 1 \land pago_c > 0 \land pago_s > 0 \land apuesta_c > 0 \land apuesta_s > 0 \land recurso > 0 \equiv apuesta_c + apuesta_s = 1 \land pago_c > 0 \land pago_s > 0 \land apuesta_c > 0 \land apuesta_s > 0 \land recurso > 0 \equiv E_2$
Y como $Pre \equiv E_2$ tenemos que $Pre \longrightarrow E_2$ ■

Demostración $P_c \longrightarrow_L wp(while..., Q_c)$

Por Teorema del Invariante, vamos a mostrar que la tripla de Hoare:

$$\{P_c\}while...\{Q_c\}$$

es válida. Entonces tenemos siguiente:

1. $P_c \implies I$
2. $\{I \wedge B\}S\{I\}$
3. $I \wedge \neg B \implies Q_c$
4. $\{I \wedge v_0 = fv\}S\{fv < v_0\}$
5. $I \wedge fv \leq 0 \implies \neg B$

Demostración $P_c \implies I$:

$P_c \equiv \{i = 0 \wedge res = recurso \wedge apuesta_c + apuesta_s = 1 \wedge pago_c > 0 \wedge pago_s > 0 \wedge apuesta_c > 0 \wedge apuesta_s > 0 \wedge recurso > 0\}$
$I \equiv \{0 \leq i \leq |eventos| \wedge res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}\}$
Queremos probar que vale I:

- $0 \leq i \leq |eventos|$
  vale, porque $i = 0$ y trivialmente sabemos que se encuentra entre 0 y la longitud de la secuencia eventos.
- $res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}$
  tenemos que $i = 0$ y $res = recurso$ entonces, como $i = 0$
  $res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi} = recurso * \prod_{j=0}^{0-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi} = recurso * \prod_{j=0}^{-1}$, que es un rango vacío, es decir, valdrá 1 y teníamos en P_c que $res = recurso$ entonces, $recurso = recurso * 1 = recurso$

Tenemos entonces que $P_c \implies I$ ■

Demostración $\{I \wedge B\}S\{I\}$

$B \equiv \{i < |eventos|\}$
$I \equiv \{0 \leq i \leq |eventos| \wedge res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}\}$
Vemos si $I \wedge B \implies wp(if...; i := i+1, I)$

- $wp(i := i+1, I) \equiv def(i+1) \wedge_L I^i_{i+1} \equiv 0 \leq i+1 \leq |eventos| \wedge_L res = recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi} \equiv E_3$
- $wp(if..., E_3) \equiv def(eventos[i]) \wedge_L ((eventos[i] \wedge wp(res := (res * apuesta.c) * pago.c, E_3) \vee (\neg eventos[i] \wedge wp(res := (res * apuesta.s) * pago.s, E_3)))) \equiv$
  $((eventos[i] \wedge (res * apuesta.c) * pago.c = recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}) \vee (\neg eventos[i] \wedge (res * apuesta.s) * pago.s = recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi})) \equiv ((eventos[i] \wedge res = \frac{1}{apuesta.c * pago.c} * recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}) \vee (\neg eventos[i] \wedge res = \frac{1}{apuesta.s * pago.s} * recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}))$
  Necesitamos llevar esto a algo similar al invariante, nos damos cuenta que $\frac{1}{apuesta.c * pago.c}$ y $\frac{1}{apuesta.s * pago.s}$ son equivalentes a $\frac{1}{\text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}}$ respectivamente, entonces reescribimos los términos:
- $0 \leq i+1 \leq |eventos| \wedge_L ((eventos[i] \wedge res = \frac{1}{\text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}} * recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}) \vee (\neg eventos[i] \wedge res = \frac{1}{\text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}} * recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi})) \equiv 0 \leq i+1 \leq |eventos| \wedge_L ((eventos[i] \wedge res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}) \vee (\neg eventos[i] \wedge res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}))$
  Aplico $(P \wedge Q) \vee (\neg P \wedge Q) \equiv Q$
- $0 \leq i+1 \leq |eventos| \wedge_L res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi} \equiv E_4$
  Chequeamos si $I \wedge B \implies E_4$
- $i \leq i+1 \leq |eventos|$
  el invariante afirma que $0 \leq i \leq |eventos|$ y la guarda afirma que $i < |eventos|$
- $res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}$