DEPARTAMENTO DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

Trabajo práctico 1

Especificación y WP

21/4/2024 AED

SCARAMOUCHE & LOS FANDANGO

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

`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` $\{(\forall j : \mathbb{Z})\ (0 \leq j < |apuestas| \longrightarrow_L \sum_{i=0}^{|eventos[j]|-1} apuestas[j][i] = 1)\}$

`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` $\{res = true \iff elUnicoEsElMaximo(trayectoria) \lor$
$elMaximoEstaEnLosBordes(trayectoria) \lor$
$elMaximoEstaEnElInterior(trayectoria)\}$

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

`pred` `elUnicoEsElMaximo` (t: $seq\langle\mathbb{R}\rangle$) {
$|trayectoria| = 1$
}

`pred` `elMaximoEstaEnLosBordes` (t: $seq\langle\mathbb{R}\rangle$) {
$elPrimeroEsElMaximo(t) \lor elUltimoEsElMaximo(t)$
}

`pred` `elMaximoEstaEnElInterior` (t: $seq\langle\mathbb{R}\rangle$) {
$(\exists i : \mathbb{Z})\ (0 < i < |t|-1 \land_L (t[i] > t[i+1] \land t[i] > t[i-1]) \land (\forall j : \mathbb{Z})\ (0 < j < |t|-1 \land_L (t[j] > t[j+1] \land t[j] > t[j-1]) \longrightarrow j = i))$
}

`pred` `elPrimeroEsElMaximo` (t: $seq\langle\mathbb{R}\rangle$) {
$t[0] > t[1] \land \neg maximoLocal(t) \land t[|t|-1] < t[|t|-2]$
}

`pred` `elUltimoEsElMaximo` (t: $seq\langle\mathbb{R}\rangle$) {
$t[|t|-1] > t[|t|-2] \land \neg maximoLocal(t) \land t[0] < t[1]$
}

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

`pred` `trayectoriaCoop` (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: $seq\langle\mathsf{Bool}\rangle$, eventos: $seq\langle\mathbb{N}\rangle$, individuo: $\mathbb{N}$) {
$(\forall n : \mathbb{Z})\ (0 \leq n < |t| \wedge_L |t[n]| = (|eventos| + 1) \wedge t[n][0] = recursos[n] \wedge (\forall k : \mathbb{Z})\ (0 < k < |t[n]| \longrightarrow_L t[n][k] = (\text{if } cooperan[k] \vee k = individuo \text{ then } 0 \text{ else } aporteIndividual(t, apuestas, pagos, eventos, n, k) \text{ fi}) + distribucionCoop(t, apuestas, pagos, eventos, cooperan, k, individuo)))$
}

`pred` `trayectoriaNoCoop` (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: $seq\langle\mathsf{Bool}\rangle$, eventos: $seq\langle\mathbb{N}\rangle$, individuo: $\mathbb{N}$) {
$(\forall n : \mathbb{Z})\ (0 \leq n < |t| \wedge_L |t[n]| = (|eventos| + 1) \wedge t[n][0] = recursos[n] \wedge (\forall k : \mathbb{Z})\ (0 < k < |t[n]| \longrightarrow_L t[n][k] = (\text{if } \neg cooperan[k] \vee k = individuo \text{ then } aporteIndividual(t, apuestas, pagos, eventos, n, k) \text{ else } 0 \text{ fi}) + distribucionNoCoop(t, apuestas, pagos, eventos, cooperan, k, individuo)))$
}

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

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

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

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

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

`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` $\{apuesta_c + apuesta_s = 1 \land pago_c > 0 \land pago_s > 0 \land apuesta_c > 0 \land apuesta_s > 0 \land recurso > 0\}$

`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
```

Como el programa que nos dan cuenta con ciclos, tenemos que probar lo siguiente:

1. $Pre \longrightarrow_L wp$(código previo al ciclo, P_c)
2. $P_c \longrightarrow_L wp$(ciclo, Q_c)
3. $Q_c \longrightarrow_L Post$

Para ello proponemos lo siguiente:

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

Equivalencias entre Q_c propuesto y $asegura$ dado:

$asegura : res = recurso * (apuesta_c * pago_c)^{\#apariciones(eventos,T)}(apuesta_s * pago_s)^{\#apariciones(eventos,F)} \equiv recurso * (apuesta_c * pago_c)^{\sum_{i=0}^{|eventos|-1} \text{if } eventos[i]=T \text{ then } 1 \text{ else } 0 \text{ fi}} * (apuesta_s * pago_s)^{\sum_{i=0}^{|eventos|-1} \text{if } eventos[i]=F \text{ then } 1 \text{ else } 0 \text{ fi}}$
Por propiedad de potenciación: $x^{f(n)+f(m)} = x^{f(n)} * x^{f(m)}$ luego $x^{\sum_{i=0}^{n} f(i)} = \prod_{i=0}^{n} x^{f(i)}$
$\equiv recurso * \prod_{j=0}^{|eventos|-1} \text{if } eventos[j] = T \text{ then } (apuesta_c * pago_c) \text{ else } 1 \text{ fi} * \prod_{j=0}^{|eventos|-1} \text{if } eventos[j] = F \text{ then } (apuesta_s * pago_s) \text{ else } 1 \text{ fi}$
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:
$asegura : res = recurso * \prod_{j=0}^{|eventos|-1} \text{if } eventos[j] \text{ then } apuesta_c * pago_c \text{ else } apuesta_s * pago_s \text{ fi}$ ■

Demostración $Pre \longrightarrow_L wp$(código previo al ciclo, P_c)

Vamos a utilizar los axiomas 1 (asignación) y 3 (composicional) vistos en la teórica.
1. $Pre \longrightarrow_L wp(res := recurso; i := 0, P_c) \overset{Axioma\ 3}{\equiv} wp(res := recurso; wp(i := 0, P_c))$
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 $P_c \longrightarrow_L wp$(ciclo, 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. $\{I \land B\}S\{I\}$
3. $I \land \neg B \implies Q_c$
4. $\{I \land v_0 = fv\}S\{fv < v_0\}$
5. $I \land fv \leq 0 \implies \neg B$

Demostración $P_c \implies I$:

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

Vamos a utilizar los axiomas 1 (asignación) y 4 (guardas) vistos en la teórica
$B \equiv \{i < |eventos|\}$
$I \equiv \{0 \leq i \leq |eventos| \land res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}\}$
Vemos si $I \land B \implies wp(if...; i := i + 1, I)$

- $wp(i := i+1, I) \overset{Axioma\ 1}{\equiv} def(i+1) \wedge_L I^i_{i+1} \equiv 0 \leq i+1 \leq |eventos| \wedge_L res = recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi} \equiv E_3$

- $wp(if..., E_3) \overset{Axioma\ 4}{\equiv} def(eventos[i]) \wedge_L ((eventos[i] \wedge wp(res := (res * apuesta.c) * pago.c, E_3) \vee (\neg eventos[i] \wedge wp(res := (res * apuesta.s) * pago.s, E_3)))) \equiv$
  $((eventos[i] \wedge (res * apuesta.c) * pago.c = recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}) \vee$
  $(\neg eventos[i] \wedge (res * apuesta.s) * pago.s = recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi})) \equiv$
  $((eventos[i] \wedge res = \frac{1}{apuesta.c * pago.c} * recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}) \vee$
  $(\neg eventos[i] \wedge res = \frac{1}{apuesta.s * pago.s} * recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}))$
  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:

- $0 \leq i+1 \leq |eventos| \wedge_L ((eventos[i] \wedge res = \frac{1}{\text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}} * recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}) \vee (\neg eventos[i] \wedge res = \frac{1}{\text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}} * recurso * \prod_{j=0}^{i} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi})) \equiv 0 \leq i+1 \leq |eventos| \wedge_L ((eventos[i] \wedge res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}) \vee (\neg eventos[i] \wedge res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}))$
  Aplicamos $(P \wedge Q) \vee (\neg P \wedge Q) \equiv Q$

- $0 \leq i+1 \leq |eventos| \wedge_L res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi} \equiv E_4$
  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 $i < |eventos|$

- $res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[i] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}$, que lo afirma el invariante.
  Entonces $\{I \wedge B\} S \{I\}$ ■

Demostración $I \wedge \neg B \implies Q_c$

$I \equiv \{0 \leq i \leq |eventos| \wedge res = recurso * \prod_{j=0}^{i-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}\}$
$\neg B \equiv \{i \geq |eventos|\}$
$Q_c \equiv \{res = recurso * \prod_{j=0}^{|eventos|-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}\}$
Queremos probar que vale Q_c:
$res = recurso * \prod_{j=0}^{|eventos|-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}$

- $0 \leq i \leq |eventos| \wedge i \geq |eventos|$, luego $i = |eventos|$
  Y al reemplazar el valor de i en I, obtenemos:
  $res = recurso * \prod_{j=0}^{|eventos|-1} \text{if } eventos[j] \text{ then } apuesta.c * pago.c \text{ else } apuesta.s * pago.s \text{ fi}$

Luego $I \wedge \neg B \implies Q_c$ ■

Demostración $\{I \wedge v_0 = fv\} S \{fv < v_0\}$

Vamos a utilizar los axiomas 1 (asignación), 3 (composicional) y 4(guardas) vistos en la teórica
$\{I \wedge B \wedge v_0 = |eventos| - i\} S \{|eventos| - i < v_0\}$
Vemos si $I \wedge B \wedge v_0 = |eventos| - i \implies wp(if...; i := i+1, |eventos| - i < v_0) \overset{Axioma\ 3}{\equiv} wp(if...; wp(i := i+1, |eventos| - i < v_0))$
$wp(i := i+1, |eventos| - i < v_0) \overset{Axioma\ 1}{\equiv} def(i+1) \wedge_L |eventos| - i - 1 < v_0 \equiv |eventos| - i < v_0 + 1$
$wp(if..., |eventos| - i < v_0 + 1) \overset{Axioma\ 4}{\equiv} def(eventos[i]) \wedge_L ((eventos[i] \wedge wp(res := res * apuesta.c * pago.c, |eventos| - i < v_0 + 1)) \vee (\neg eventos[i] \wedge wp(res := res * apuesta.s * pago.s, |eventos| - i < v_0 + 1))) \equiv ((eventos[i] \wedge |eventos| - i < v_0 + 1) \vee (\neg eventos[i] \wedge |eventos| - i < v_0 + 1))$
Aplicamos $(P \wedge Q) \vee (\neg P \wedge Q) \equiv Q$
Vemos si vale la implicación
$I \wedge i < |eventos| \wedge |eventos| - i = v_0 \implies |eventos| - i < v_0 + 1$, porque $A = B \implies A < B + 1$
Entonces $\{I \wedge v_0 = fv\} S \{fv < v_0\}$ ■

Demostración $I \wedge fv \leq 0 \implies \neg B$

Por la definición de fv tenemos:

- $I \wedge |eventos| - i \leq 0$
  si sumamos i de ambos lados de la igualdad,
  $I \wedge |eventos| - i + i \leq 0 + i \equiv I \wedge |eventos| \leq i$

Como $|eventos| \leq i \equiv \neg B$ es una implicación del tipo $P \wedge Q \implies P$, tenemos $I \wedge \neg B \implies \neg B$ ■

Demostración $Q_c \longrightarrow_L Post = asegura \equiv Q_c$

Como la implementación termina con el ciclo y el Q_c equivale al asegura (como lo hemos demostrado anteriormente), por monotonía sabemos que $Pre \implies wp(\text{programa completo}, Post)$, por lo tanto decimos que la especificación es correcta respecto a su implementación. ■