



Trabajo práctico 1: Especificación y WP

21 de abril de 2024

Algoritmos y Estructuras de Datos

Grupo JDDBVCNLUCVDVTZLNSIH

Integrante	LU	Correo electrónico
Labastíe, Gaspar	660/23	gaspilabastie@gmail.com
Rugo, Julian	1414/23	julianrugo22@gmail.com
Torres, Emiliano	80/23	emilianomtorres1@gmail.com
Vanotti, Franco	464/23	fvanotti15@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>

0. Aclaraciones generales

- Los índices de las listas recursos, cooperan, trayectorias, apuestas, pagos, eventos representa el identificador de los individuos.
- recursos: $\text{seq}\langle\mathbb{R}\rangle$ es la lista con el recurso de cada individuo.
- cooperan: $\text{seq}\langle\text{Bool}\rangle$ es la lista que indica T rue si el individuo en dicha posición coopera.
- trayectorias: $\text{seq}\langle\text{seq}\langle\mathbb{R}\rangle\rangle$ indica para cada individuo, en cada paso de tiempos, cuántos recursos (\mathbb{R}) cuenta.
- eventos: $\text{seq}\langle\text{seq}\langle\mathbb{N}\rangle\rangle$ indica para cada individuo, en cada paso temporal, qué evento le ha tocado.
- apuestas: $\text{seq}\langle\text{seq}\langle\mathbb{R}\rangle\rangle$ indica para cada individuo, para cada evento posible (numerados desde 0), cuánto apostará.
- pagos: $\text{seq}\langle\text{seq}\langle\mathbb{R}\rangle\rangle$ indica para cada individuo, para cada evento, cuánto se pagará. Notar que a diferencia del ejemplo, estamos resolviendo un caso más general donde el pago de cada evento puede diferir por individuo.
- Las personas que no *cooperan* no aportan nada al fondo monetario común.
- Los *recursos* iniciales son positivos.
- Todos los *pagos* son positivos.
- Las *apuestas* de los individuos representan la proporción de los recursos que los individuos invierten a cada una de los eventos posibles. Notar nuevamente que a diferencia del ejemplo, en este caso más general, podríamos tener apuestas distintas para cada evento por cada individuo.
- Cada individuo apuesta siempre el mismo porcentaje por cada evento posible (es decir, el mismo número en cada paso temporal). Por ejemplo, si tenemos dos eventos; cara y ceca y apuesta 0, 4 por cara y 0,6 por seca, en cada paso temporal apostará esas proporciones.
- Se considera al número 0 como parte de \mathbb{N} .
- Debido a la ambigüedad presente en el ejercicio sobre especificación en cuanto a si un jugador puede o no puede apostar todo sus recursos a un unico evento, y por ende que la apuesta de dicho jugador a un evento sea igual a 0, nosotros tomamos bajo nuestro criterio personal que eso si es posible, y se llevo a cabo el ejercicio considerando dicha posibilidad.

1. Especificación

1. **redistribucionDeLosFrutos**: Calcula los recursos que obtiene cada uno de los individuos luego de que se redistribuyen los recursos del fondo monetario común en partes iguales. El fondo monetario común se compone de la suma de *recursos* iniciales aportados por todas las personas que *cooperan*. La salida es la lista de recursos que tendrá cada jugador.

```
proc redistribucionDeLosFrutos (in recursos : seq⟨ℝ⟩, in cooperan : seq⟨Bool⟩) : seq⟨ℝ⟩
```

```
  requiere { |recursos| > 0 ∧ |recursos| = |cooperan| ∧  
  (∀i : ℤ) (0 ≤ i < |recursos| →L recursos[i] > 0) }
```

```
  asegura { |res| = |recursos| ∧L nuevosRecursosCooperan(recursos, cooperan, res) ∧  
  nuevosRecursosNoCooperan(recursos, cooperan, res) }
```

```
pred nuevosRecursosCooperan (recursos : seq⟨ℝ⟩, cooperan : seq⟨Bool⟩, res : seq⟨ℝ⟩) {
```

```
  (∀i : ℤ) (0 ≤ i < |recursos| ∧L cooperan[i] = true →L res[i] = distribuciónFondoComún(recursos, cooperan))
```

```
}
```

```
pred nuevosRecursosNoCooperan (recursos : seq⟨ℝ⟩, cooperan : seq⟨Bool⟩, res : seq⟨ℝ⟩) {
```

```
  (∀i : ℤ) (0 ≤ i < |recursos| ∧L cooperan[i] = false →L res[i] = distribuciónFondoComún(recursos, cooperan) +  
  recursos[i])
```

```
}
```

```
aux distribuciónFondoComún (recursos : seq⟨ℝ⟩, cooperan : seq⟨Bool⟩) : ℝ =
```

```
(∑i=0|recursos|-1)(if cooperan[i] = true then  $\frac{\text{recursos}[i]}{|\text{recursos}|}$  else 0 fi);
```

2. **trayectoriaDeLosFrutosIndividualesALargoPlazo**: Actualiza (In/Out) la lista de *trayectorias* de los recursos de cada uno de los individuos. Inicialmente, cada una de las trayectorias (listas de recursos) contiene un único elemento que representa los recursos iniciales del individuo. El procedimiento agrega a las *trayectorias* los recursos que los individuos van obteniendo a medida que se van produciendo los resultados de los *eventos* en función de la lista de *pagos* que le ofrece la naturaleza (o casa de apuestas) a cada uno de los individuos, las *apuestas* (o inversiones) que realizan los individuos en cada paso temporal, y la lista de individuos que *cooperan* aportando al fondo monetario común.

```

proc trayectoriaDeLosFrutosIndividualesALargoPlazo (inout trayectorias: seq⟨seq⟨R⟩⟩, in cooperan : seq⟨Bool⟩,
in apuestas: seq⟨seq⟨R⟩⟩, in pagos: seq⟨seq⟨R⟩⟩, in eventos: seq⟨seq⟨N⟩⟩)

  requiere { |trayectorias| = |cooperan| = |apuestas| = |pagos| = |eventos| ∧
  trayectorias = Trayectorias0 ∧L
  (∀i : Z)(0 ≤ i < |trayectorias| →L (|trayectorias[i]| = 1 ∧ trayectorias[i][0] > 0)) ∧ |pagos| > 0 ∧
  (∀k, l : Z)(0 ≤ k, l < |apuestas| →L |apuestas[k]| = |apuestas[l]|) ∧
  (∀i : Z)((0 ≤ i < |apuestas| →L sumarApuestasIndividuo(apuestas[i]) = 1) ∧L
  (∀j : Z)(0 ≤ j < |pagos[i]| →L 0 ≤ apuestas[i][j] ≤ 1)) ∧
  (∀k, l : Z)(0 ≤ k, l < |pagos| →L |pagos[k]| = |pagos[l]|) ∧
  (∀i : Z)(0 ≤ i < |pagos| →L (∀j : Z)(0 ≤ j < |pagos[i]| →L 0 < pagos[i][j])) ∧
  (∀i : Z)(0 ≤ i < |pagos| →L (∀j : Z)(0 ≤ j < |apuestas| →L |pagos[i]| = |apuestas[j]|)) ∧
  (∀i : Z)(0 ≤ i < |eventos| →L |eventos[i]| > 0) ∧
  (∀i : Z)(0 ≤ i < |eventos| →L |eventos[0]| = |eventos[i]|) ∧
  (∀i : Z)(0 ≤ i < |eventos| →L (∀j : Z)(0 ≤ j < |eventos[i]| →L 0 ≤ eventos[i][j] < |pagos[i]|)) }

  asegura { |trayectorias| = |eventos| ∧L (∀i : Z)(0 ≤ i < |trayectorias| →L trayectorias[i][0] = Trayectorias0[i][0]) ∧

  (∀i : Z)(0 ≤ i < |trayectorias| →L |trayectorias[i]| = |eventos[i]| + 1) ∧
  (∀i : Z)(0 ≤ i < |trayectorias| →L (∀j : Z)(0 < j < |trayectorias[i]| →L
  (trayectoriasCooperan(trayectorias, cooperan, apuestas, pagos, eventos) ∧
  trayectoriasNoCooperan(trayectorias, cooperan, apuestas, pagos, eventos)))) }

pred trayectoriasCooperan (trayectorias : seq⟨seq⟨R⟩⟩, cooperan : seq⟨Bool⟩, apuestas : seq⟨seq⟨R⟩⟩, pagos : seq⟨seq⟨R⟩⟩,
eventos : seq⟨seq⟨Z⟩⟩) {
  cooperan[i] = true →L trayectorias[i][j] = distribuciónFondoComúnTrayectoria(trayectorias, cooperan, apuestas,
pagos, eventos, j)
}

pred trayectoriasNoCooperan (trayectorias : seq⟨seq⟨R⟩⟩, cooperan : seq⟨Bool⟩, apuestas : seq⟨seq⟨R⟩⟩, pagos :
seq⟨seq⟨R⟩⟩, eventos : seq⟨seq⟨Z⟩⟩) {
  cooperan[i] = false →L trayectorias[i][j] = distribuciónFondoComúnTrayectoria(trayectorias, cooperan, apuestas,
pagos, eventos, j) + trayectoria[i][j - 1] * gananciaIndividuo(apuestas[i], pagos[i],
eventos[i][j - 1])
}

aux sumarApuestasIndividuo (apuestasIndividuo : seq⟨R⟩) : R =
(∑n=0|apuestasIndividuo|-1)(apuestasIndividuo[n]);

aux gananciaIndividuo (apuestaIndividuo: seq⟨R⟩, pagosIndividuo: seq⟨R⟩, resultadoEventoIndividuo: R) : R =
apuestaIndividuo[resultadoEventoIndividuo] * pagosIndividuo[resultadoEventoIndividuo];

aux recolecciónFondoComún (trayectorias: seq⟨seq⟨R⟩⟩, cooperan : seq⟨Bool⟩, apuestas: seq⟨seq⟨R⟩⟩, pagos: seq⟨seq⟨R⟩⟩,
eventos: seq⟨seq⟨N⟩⟩, j : Z) : R =
(∑i=0j-1)(if cooperan[i] = true then trayectorias[i][j - 1] * gananciaIndividuo(apuestas[i], pagos[i], eventos[i][j - 1]) else 0 fi)

aux distribuciónFondoComúnTrayectoria (trayectorias: seq⟨seq⟨R⟩⟩, cooperan : seq⟨Bool⟩, apuestas: seq⟨seq⟨R⟩⟩,
pagos: seq⟨seq⟨R⟩⟩, eventos: seq⟨seq⟨N⟩⟩, j : Z) : R =
(recolecciónFondoComún(trayectorias, cooperan, apuestas, pagos, eventos, j)) ;
|trayectorias|

```

3. **trayectoriaExtrañaEscalera** Esta función devuelve *True* sii en la trayectoria de un individuo existe un único punto mayor a sus vecinos (llamado máximo local). Un elemento es máximo local si es mayor estricto que sus vecinos inmediatos.

```

proc trayectoriaExtrañaEscalera (in trayectoria: seq⟨ℝ⟩) : Bool
  requiere {|trayectoria| > 0 ∧ (∀i : ℤ) (0 ≤ i < |trayectoria| →L trayectoria[i] ≥ 0)}
  asegura {res = true ↔
    (∃m : ℤ)((∀i : ℤ)((0 ≤ i, m < |trayectoria| ∧ i ≠ m) →L trayectoria[m] > trayectoria[i]) ∧L
    (∀n : ℤ)(0 ≤ n ≤ m →L trayectoria[n] > trayectoria[n - 1]) ∧
    (∀n : ℤ)(m ≤ n < |trayectoria| - 1 →L trayectoria[n] > trayectoria[n + 1]))}

```

4. **individuoDecideSiCooperarONo** Un *individuo* actualiza su comportamiento cooperativo / no-cooperativo (*cooperan[individuo]*) en función de los recursos iniciales, de quienes *cooperan*, de los *pagos* que se le ofrecen a cada individuo, de las inversiones o *apuestas* de cada individuo, y del resultado de los *eventos* que recibe cada individuo, eligiendo el comportamiento que maximiza sus recursos individuales luego de que ocurren todos los eventos.

```

proc individuoDecideSiCooperarONo (in individuo:ℕ, in recursos: seq⟨ℝ⟩, inout cooperan: seq⟨Bool⟩, in apuestas:
seq⟨seq⟨ℝ⟩⟩, in pagos: seq⟨seq⟨ℝ⟩⟩, in eventos: seq⟨seq⟨ℕ⟩⟩)
  requiere {|recursos| > 0 ∧ |recursos| = |cooperan| = |apuestas| = |pagos| = |eventos| ∧
    cooperan = Cooperan0 ∧
    0 ≤ individuo < |recursos| ∧L
    (∀i : ℤ)(0 ≤ i < |recursos| →L recursos[i] > 0) ∧
    (∀i : ℤ)(0 ≤ i < |eventos| →L |eventos[i]| > 0) ∧
    (∀i : ℤ)(0 ≤ i < |eventos| →L |eventos[i]| = |eventos[0]|) ∧
    (∀i : ℤ)(0 ≤ i < |eventos| →L (∀j : ℤ)(0 ≤ j < |eventos[i]| →L 0 ≤ eventos[i][j] < |pagos[i]|)) ∧
    (∀k, l : ℤ)(0 ≤ k, l < |apuestas| →L |apuestas[k]| = |apuestas[l]|) ∧
    (∀i : ℤ)((0 ≤ i < |apuestas| →L sumarApuestasIndividuo(apuestas[i]) = 1) ∧L
    (∀j : ℤ)(0 ≤ j < |apuestas[i]| →L 0 ≤ apuestas[i][j] ≤ 1)) ∧
    (∀k, l : ℤ)(0 ≤ k, l < |pagos| →L |pagos[k]| = |pagos[l]|) ∧
    (∀i : ℤ)(0 ≤ i < |pagos| →L (∀j : ℤ)(0 ≤ j < |pagos[i]| →L 0 < pagos[i][j])) ∧
    (∀i : ℤ)(0 ≤ i < |pagos| →L (∀j : ℤ)(0 ≤ j < |apuestas| →L |pagos[i]| = |apuestas[j]|))}
  asegura {|cooperan| = |Cooperan0| ∧L
    (∀i : ℤ)(0 ≤ i < |Cooperan0| ∧ i ≠ individuo →L cooperan[i] = Cooperan0[i]) ∧
    (∃cooperanAlt : seq⟨Bool⟩)(|cooperanAlt| = |Cooperan0| ∧L
    (∀i : ℤ)(0 ≤ i < |cooperanAlt| ∧ i ≠ individuo →L cooperanAlt[i] = Cooperan0[i]) ∧
    (cooperanAlt[individuo] = ¬Cooperan0[individuo])) ∧
    (∃trayectoria : seq⟨seq⟨ℝ⟩⟩)(|trayectoria| = |eventos| ∧L primeroTieneAREcursos(recursos, trayectoria) ∧
    moduloEventosMasUno(eventos, trayectoria) ∧ elementosDeT(apuestas, pagos, eventos, Cooperan0, trayectoria) ∧
    (∃trayectoriaAlt : seq⟨seq⟨ℝ⟩⟩)(|trayectoriaAlt| = |eventos| ∧L primeroTieneAREcursos(recursos, trayectoriaAlt) ∧
    moduloEventosMasUno(eventos, trayectoriaAlt) ∧
    elementosDeT(apuestas, pagos, eventos, cooperanAlt, trayectoriaAlt) ∧L
    cooperan[individuo] = mejorEleccion(trayectoria, trayectoriaAlt, Individuo))}

pred primeroTieneAREcursos (recursos:seq⟨ℝ⟩, trayectoria:seq⟨seq⟨ℝ⟩⟩) {
  (∀i : ℤ)(0 ≤ i < |trayectoria| →L trayectoria[i][0] = recursos[i])
}

pred moduloEventosMasUno (eventos:seq⟨seq⟨ℝ⟩⟩, trayectoria:seq⟨seq⟨ℝ⟩⟩) {
  (∀i : ℤ)(0 ≤ i < |trayectoria| →L |trayectoria[i]| = |eventos| + 1)
}

pred elementosDeT (apuestas:seq⟨seq⟨ℝ⟩⟩, pagos:seq⟨seq⟨ℝ⟩⟩, eventos:seq⟨seq⟨ℝ⟩⟩,
cooperan:seq⟨Bool⟩, trayectoria:seq⟨seq⟨ℝ⟩⟩) {
  (∀i : ℤ)(0 ≤ i < |trayectoria| →L (∀j : ℤ)(0 ≤ j < ||trayectoria[i]|| →L
    trayectoriasCooperan(trayectorias, cooperan, apuestas, pagos, eventos) ∧
    trayectoriasNoCooperan(trayectorias, cooperan, apuestas, pagos, eventos)))
}

aux mejorEleccion (trayectoria:seq⟨seq⟨ℝ⟩⟩, trayectoriaAlt:seq⟨seq⟨ℝ⟩⟩, Individuo) : Bool =
  (if trayectoria[individuo][|trayectoria[individuo]| - 1] ≥ trayectoriaAlt[individuo][|trayectoriaAlt[individuo]| - 1] then
    Cooperan0[individuo] else CooperanAlt[individuo] fi);

```

5. **individuoActualizaApuesta** Un *individuo* actualiza su apuesta ($apuestas[individuo]$) en función de los *recursos* iniciales, de la lista de individuos que *cooperan*, de los *pagos* que se le ofrecen a cada individuo, de las inversiones o *apuestas* de cada individuo y del resultado de los eventos que recibe cada individuo, eligiendo la apuesta que maximiza sus recursos individuales luego de que ocurren todos los eventos.

proc individuoActualizaApuesta (in individuo:N, in recursos: seq(R), in cooperan: seq(Bool), inout apuestas: seq(seq(R)), in pagos: seq(seq(R)), in eventos: seq(seq(N)))

requiere $\{|recursos| > 0 \wedge |recursos| = |cooperan| = |apuestas| = |pagos| = |eventos| \wedge_L$
 $(\forall i : \mathbb{Z})(0 \leq i < |recursos| \rightarrow_L recursos[i] > 0) \wedge$
 $apuestas = Apuestas_0 \wedge$
 $(\forall i, j : \mathbb{Z})(0 \leq i, j < |apuestas| \rightarrow_L |apuestas[i]| = |apuestas[j]|) \wedge$
 $(\forall i : \mathbb{Z})(0 \leq i < |apuestas| \rightarrow_L sumarApuestasIndividuo(apuestas[i] = 1)) \wedge$
 $(\forall i, j : \mathbb{Z})(0 \leq i < |apuestas| \rightarrow_L 0 \leq apuestas[i][j] \geq 1) \wedge$
 $(\forall i, j : \mathbb{Z})(0 \leq i < |pagos| \wedge 0 \leq j < |pagos[i]| \rightarrow_L pagos[i][j] > 0) \wedge$
 $(\forall i, j : \mathbb{Z})(0 \leq i, j < |eventos| \rightarrow_L |eventos[i]| = |eventos[j]|) \wedge$
 $(\forall i, j : \mathbb{Z})(0 \leq i < |eventos| \wedge 0 \leq j < |eventos[i]| \rightarrow_L 0 \leq eventos[i][j] \geq |pagos[i]|)\}$
 asegura $\{(\exists m : seq(seq(R)))(esApuestaVariante(m, individuo) \wedge_L (\exists trayectoriam : seq(seq(R)))$
 $(esTrayectoria(trayectoriam, recursos, eventos, pagos, cooperan, m) \wedge_L$
 $(\forall A : seq(seq(R)))(esApuestaVariante(A, individuo) \rightarrow_L$
 $(\forall trayectoria : seq(seq(R)))(esTrayectoria(trayectoria, recursos, eventos, pagos, cooperan, A) \rightarrow_L$
 $trayectoriam[individuo][|trayectoriam[individuo]| - 1] \geq trayectoria[individuo][|trayectoria[individuo]| - 1])))\}$

pred esApuestaVariante (A : seq(seq(R)), individuo : N) {

$|A| = |Apuestas_0| \wedge (\forall i : \mathbb{Z})(0 \leq i < |Apuestas_0| \wedge i \neq individuo \rightarrow_L A[i] = Apuestas_0[i]) \wedge$
 $|A[individuo]| = |Apuestas_0[0]| \wedge sumarApuestasIndividuo(A[individuo]) = 1 \wedge$
 $(\forall j : \mathbb{Z})(0 \leq j < |A[individuo]| \rightarrow_L 0 \leq A[individuo][j] \leq 1)$

}

pred esTrayectoria (trayectoria : seq(seq(R)), recursos : seq(R), eventos : seq(seq(Z)), pagos : seq(seq(R)), cooperan : seq(Bool), apuestas : seq(seq(R))) {

$primeroTieneAREcursos(recursos, trayectoria) \wedge moduloEventosMasUno(eventos, trayectoria) \wedge$
 $elementosDeT(apuestas, pagos, eventos, cooperan, trayectoria)$

}

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 sello, cuánto se ganaría si se juega completamente solo. Se contempla que el evento *True* es cuando sale cara

```
proc frutoDelTrabajoPuramenteIndividual (in recursos:seq⟨ℝ⟩, in apuestas:⟨s: ℝ, c:ℝ⟩, in pago:⟨s: ℝ, c:ℝ⟩, in eventos:seq⟨Bool⟩,
out res: ℝ)
```

```
  requiere {apuestac + apuestas = 1 ∧ pagoc > 0 ∧ pagos > 0 ∧ apuestac > 0 ∧ apuestas > 0 ∧ recurso > 0}
  asegura {res = recurso(apuestacpagoc)#apariciones(eventos, T)(apuestaspagos)#apariciones(eventos, F)}
```

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

```
res = recursos
i = 0
While (i < |eventos|) do
  if eventos[i] then
    res = (res * apuestac) * pagoc
  else
    res = (res * apuestas) * pagos
  endif
  i = i + 1
endwhile
```

Para probar la correctitud de este código usamos el teorema de corrección de un ciclo, para lo cual se propone el siguiente invariante:

$$I \equiv 0 \leq i \leq |\text{eventos}| \wedge$$

$$\text{res} = \text{recurso}(\text{apuesta}_c \text{pago}_c)^{(\sum_{j=0}^{i-1} (\text{if } \text{eventos}[j] = \text{true then } 1 \text{ else } 0 \text{ fi}))} (\text{apuesta}_s \text{pago}_s)^{(\sum_{j=0}^{i-1} (\text{if } \text{eventos}[j] = \text{false then } 1 \text{ else } 0 \text{ fi}))}$$

Y la siguiente funcion decreciente:

$$f_v \equiv |\text{eventos}| - i$$

Entonces hay que probar:

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

1. Veamos que se cumple 1 ($P_c \longrightarrow I$):

$$P_c \equiv \text{res} = \text{recursos} \wedge i = 0$$

$$I \equiv 0 \leq i \leq |\text{eventos}| \wedge$$

$$\text{res} = \text{recurso}(\text{apuesta}_c \text{pago}_c)^{(\sum_{j=0}^{i-1} (\text{if } \text{eventos}[j] = \text{true then } 1 \text{ else } 0 \text{ fi}))} (\text{apuesta}_s \text{pago}_s)^{(\sum_{j=0}^{i-1} (\text{if } \text{eventos}[j] = \text{false then } 1 \text{ else } 0 \text{ fi}))}$$

Podemos remplazar P_c en I para ver que es tautologico.

$$I \equiv 0 \leq 0 \leq |\text{eventos}| \wedge$$

$$\text{recurso} = \text{recurso}(\text{apuesta}_c \text{pago}_c)^{(\sum_{j=0}^{0-1} (\text{if } \text{eventos}[j] = \text{true then } 1 \text{ else } 0 \text{ fi}))} (\text{apuesta}_s \text{pago}_s)^{(\sum_{j=0}^{0-1} (\text{if } \text{eventos}[j] = \text{false then } 1 \text{ else } 0 \text{ fi}))}$$

$$\equiv \text{true} \wedge \text{res} = \text{recurso}(\text{apuesta}_c \text{pago}_c)^0 (\text{apuesta}_s \text{pago}_s)^0$$

$$\equiv \text{true} \wedge \text{recurso} = \text{recurso} \equiv \text{true}$$

Entonces se puede observar que $P_c \longrightarrow I$ se cumple.

2. Veamos que se cumple $2 \{I \wedge B\} S \{I\}$:

$$I \equiv 0 \leq i \leq |eventos| \wedge$$

$$res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}$$

$$B : i < |eventos|$$

Primero vemos si $I \wedge B \longrightarrow wp(if..., i := i + 1, I)$

$$wp(i := i + 1, I) \equiv def(i + 1) \wedge_L I_{i+1} \equiv (-1 \leq i \leq |eventos| - 1) \wedge$$

$$(res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^i)(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^i)(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}) \\ \equiv E1$$

Entonces:

$$wp(if..., E1) \equiv def(eventos[i]) \wedge ((eventos[i] \wedge wp(res := resapuesta_c pago_c, E1)) \vee$$

$$(\neg eventos[i] \wedge wp(res := resapuesta_s pagos, E1)))$$

$$\equiv (0 \leq i \leq |eventos|) \wedge$$

$$((eventos[i] \wedge$$

$$resapuesta_c pago_c = recurso(apuesta_c pago_c)^{(\sum_{j=0}^i)(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^i)(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}) \\ \vee$$

$$(\neg eventos[i] \wedge$$

$$resapuesta_s pago_s = recurso(apuesta_c pago_c)^{(\sum_{j=0}^i)(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^i)(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}) \\)$$

Luego, en la ecuacion:

$$resapuesta_c pago_c = recurso(apuesta_c pago_c)^{(\sum_{j=0}^i)(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^i)(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}$$

se puede pasar dividiendo el término $apuesta_c pago_c$ dividiendo, ya que tanto $apuesta_c$ como $pago_c$ son mayores a 0 por lo pedido en el requiere.

Lo mismo ocurre en la ecuacion:

$$resapuesta_s pago_s = recurso(apuesta_c pago_c)^{(\sum_{j=0}^i)(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^i)(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}$$

donde se puede pasar dividiendo el término $apuesta_s pago_s$ dividiendo por la misma razón.

De esta forma, se obtiene la siguiente expresión:

$$wp(if..., E1) \equiv (0 \leq i \leq |eventos|) \wedge$$

$$((eventos[i] \wedge$$

$$res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}) \vee$$

$$(\neg eventos[i] \wedge$$

$$res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^i)(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)})$$

Por otro lado, tambien cabe destacar que en las ecuaciones:

$$res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^i)(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)} \text{ y}$$

$$res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^i)(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}$$

se pueden reemplazar

$$(apuesta_s pago_s)^{(\sum_{j=0}^i)(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)} \text{ por } (apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)} \text{ y}$$

$$(apuesta_c pago_c)^{(\sum_{j=0}^i)(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)} \text{ por } (apuesta_c pago_c)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}$$

respectivamente, ya que en que ambos casos, el último elemento es igual a 0 y por ende no afecta al resultado de la sumatoria.

De ahí que se obtiene la siguiente expresión:

$$wp(if..., E1) \equiv (0 \leq i \leq |eventos|) \wedge$$

$$((eventos[i] \wedge$$

$$res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}) \vee$$

$$(\neg eventos[i] \wedge$$

$$res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)})$$

$$\equiv (0 \leq i \leq |eventos|) \wedge ((eventos[i] \vee \neg eventos[i]) \wedge$$

$$(res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)}))$$

$$\equiv (0 \leq i \leq |eventos|) \wedge$$

$$(res = recurso(apuesta_c pago_c)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)}(apuesta_s pago_s)^{(\sum_{j=0}^{i-1})(if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)})$$

$$\equiv I$$

Entonces se puede observar que $\{I \wedge B\} S \{I\}$ se cumple.

3. Veamos que se cumple 3 ($I \wedge \neg B \longrightarrow Q_c$):

$$\begin{aligned}
Q_c &: res = recurso(apuesta_c pago_c) \# apariciones(eventos, T) (apuesta_s pago_s) \# apariciones(eventos, F) \\
\neg B &: i \geq |eventos| \\
I \wedge \neg B &\longrightarrow Q_c \\
I \wedge \neg B &\equiv 0 \leq i \leq |eventos| \wedge i \geq |eventos| \\
res &= recurso(apuesta_c pago_c) (\sum_{j=0}^{i-1} (if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)) (apuesta_s pago_s) (\sum_{j=0}^{i-1} (if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)) \\
I \wedge \neg B &\equiv i = |eventos| \wedge \\
res &= recurso(apuesta_c pago_c) (\sum_{j=0}^{|eventos|-1} (if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)) (apuesta_s pago_s) (\sum_{j=0}^{|eventos|-1} (if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)) \\
I \wedge \neg B &\equiv i = |eventos| \wedge res = recurso(apuesta_c pago_c) \# apariciones(eventos, T) (apuesta_s pago_s) \# apariciones(eventos, F)
\end{aligned}$$

Entonces se puede observar que $I \wedge \neg B \longrightarrow Q_c$ se cumple.

4. Veamos que se cumple 4 ($\{I \wedge B \wedge v_0 = f_v\} \ S \ \{f_v < v_0\}$):

$$\begin{aligned}
I &\equiv 0 \leq i \leq |eventos| \wedge \\
res &= recurso(apuesta_c pago_c) (\sum_{j=0}^{i-1} (if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)) (apuesta_s pago_s) (\sum_{j=0}^{i-1} (if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)) \\
B &\equiv i < |eventos| \\
f_v &= |evento| - i \\
I \wedge B \wedge v_0 = f_v &\equiv 0 \leq i \leq |eventos| \wedge \\
res &= recurso(apuesta_c pago_c) (\sum_{j=0}^{i-1} (if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)) (apuesta_s pago_s) (\sum_{j=0}^{i-1} (if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)) \wedge \\
i &< |eventos| \wedge |evento| - i = v_0 \\
\equiv 0 \leq i &< |eventos| \wedge |evento| - i = v_0 \wedge \\
res &= recurso(apuesta_c pago_c) (\sum_{j=0}^{i-1} (if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)) (apuesta_s pago_s) (\sum_{j=0}^{i-1} (if\ eventos[j]=false\ then\ 1\ else\ 0\ fi))
\end{aligned}$$

Primero hay que buscar la precondition más débil del ciclo:

$$wp(i := i + 1, f_v < v_0) \equiv def(i + 1) \wedge Q_{i+1}^i \equiv def(i) \wedge |evento| - (i + 1) < v_0 \equiv def(i) \wedge |evento| - i \leq v_0$$

Luego tenemos la siguiente expresión:

$$\begin{aligned}
&wp((if\ eventos[i]\ then\ res = (res * apuesta_c) pago_c\ else\ res = (res * apuesta_s) pago_s\ fi), Q) \\
&\equiv def(eventos[i]) \wedge_L ((eventos[i] \wedge wp(res := res(apuesta_c pago_c), Q)) \vee (\neg eventos[i] \wedge wp(res := res(apuesta_s pago_s), Q))) \\
&\equiv 0 \leq i < |evento| \wedge_L ((eventos[i] \wedge def(res(apuesta_c pago_c)) \wedge (def(i) \wedge |evento| - i \leq v_0)) \vee \\
&(\neg eventos[i] \wedge def(res(apuesta_s pago_s)) \wedge (def(i) \wedge |evento| - i \leq v_0))) \\
&\equiv 0 \leq i < |evento| \wedge_L ((eventos[i] \wedge def(res) \wedge def(apuesta_c) \wedge def(pago_c) \wedge def(i) \wedge |evento| - i \leq v_0) \vee \\
&(\neg eventos[i] \wedge def(res) \wedge def(apuesta_s) \wedge def(pago_s) \wedge def(i) \wedge |evento| - i \leq v_0)) \\
&\equiv 0 \leq i < |evento| \wedge_L ((eventos[i] \wedge |evento| - i \leq v_0) \vee (\neg eventos[i] \wedge |evento| - i \leq v_0)) \\
&\equiv 0 \leq i < |evento| \wedge_L ((eventos[i] \vee \neg eventos[i]) \wedge |evento| - i \leq v_0) \\
&\equiv 0 \leq i < |evento| \wedge |evento| - i \leq v_0
\end{aligned}$$

Finalmente, se puede observar:

$$\begin{aligned}
&(0 \leq i < |eventos| \wedge |evento| - i = v_0 \wedge \\
res &= recurso(apuesta_c pago_c) (\sum_{j=0}^{i-1} (if\ eventos[j]=true\ then\ 1\ else\ 0\ fi)) (apuesta_s pago_s) (\sum_{j=0}^{i-1} (if\ eventos[j]=false\ then\ 1\ else\ 0\ fi)) \\
&\longrightarrow (0 \leq i < |eventos| \wedge |evento| - i = v_0) \longrightarrow (0 \leq i < |evento| \wedge |evento| - i \leq v_0) \\
&\text{y por ende, } \{I \wedge B \wedge v_0 = f_v\} \ S \ \{f_v < v_0\} \text{ se cumple.}
\end{aligned}$$

5. Veamos que se cumple 5 ($I \wedge f_v \leq 0 \longrightarrow \neg B$):

$$\begin{aligned}
f_v &\equiv |eventos| - i \\
\neg B &\equiv i \geq |eventos| \\
I \wedge f_v \leq 0 &\equiv I \wedge |eventos| - i \leq 0 \equiv I \wedge |eventos| \leq i \equiv I \wedge \neg B \\
\text{Luego, } I \wedge \neg B &\longrightarrow \neg B
\end{aligned}$$

Entonces se puede observar que $I \wedge f_v \leq 0 \longrightarrow \neg B$ se cumple.