DEPARTAMENTO DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Practico N°1: Fondo Monetario Común

Grupo: AguanteTaaallere

Integrante	LU	Correo electrónico
Olmos , Francisco	1101/23	francisco.olmos.99@gmail.com
Andina Silva, Augusto	1344/23	augustoandinasilva@gmail.com
López Porto, Gregorio	1376/23	gregoriolopezporto@gmail.com
Quintana , Joaquín Ezequiel	1356/23	joaquin32flores@gmail.com

Nota: Reentregar
- Debe estar hecho en latex
- Faltan requieres
- Utilizan procs en procs !
- No está el punto 2

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

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

y cómo deben ser entre ellos ?

requiere{listaNoVacia(recursos) ∧L igualLargo(recursos, cooperan)}

asegura{igualLargo(res, recursos) ∧L
(∀i : Z)(0< i< |cooperan| –>L (cooperan[i] → res[i] = totalRecursos/|cooperan| v
cooperan[i] → res[i] = totalRecursos/|cooperan|))}

Está mal definida, El implica inicial ya dice todo...

Es esto ?

aux TotalDeRecursos(cooperan :seq⟨Bool⟩,recursos :seq⟨R⟩) : R = ∑_i=0 ^|recursos|−1 (**If** cooperan[i] **then** recursos[i] **else** 0 **fi**) ;

1.2) 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 { listaNoVacia(trayectoria) ∧L esMatriz(trayectoria) ∧ esMatriz(pagos) ∧L listaNoVacia(trayectoria) ∧L igualLargo(trayectoria, cooperan, apuestas, pagos, eventos) ∧ apuestasValidas(apuestas) ∧ pagosValidos(pagos)}

Cómo deben ser las apuestas ?
Y los eventos contra pagos y apuestas?

No liga la entrada

asegura{igualLargo(res, trayectorias) ∧L ((∀j : Z) (|trayectorias[0]| ≤ j < |eventos[0]|) →L ((∀i : Z)(0 ≤ i < |trayectorias|) ∧L resP[i] = trayectorias[i][j-1]*apuestas[i][eventos[i][j]]*pagos[i][eventos[i][j]]) ∧L (trayectorias[i] = old(trayectorias[i]) ++ redistribucionDeLosFrutos (resP, cooperan)[i]))}

Acá no hay res

No pueden usar un proc !

qué es ++ ? No se puede leer

1.3) proc **trayectoriaExtrañaEscalera** (in trayectoria : seq⟨R⟩) : Bool

requiere {listaNoVacia(trayectoria) ∧ todosPositivos(trayectoria)}

asegura{ (∃! i : Z) ((0 ≤ i < |trayectoria|) –>L ((|trayectoria| = 1)
ˇL ((i=0) ∧ (trayectoria[i] > trayectoria[i+1]))
ˇL ((i=|trayectoria|-1) ∧ (trayectoria[i] > trayectoria[i-1]))
ˇL ((trayectoria[i] > trayectoria[i+1]) ∧ (trayectoria[i] > trayectoria[i-1]))}

Puede estar en las puntas y el medio
No funciona

1.4) 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{ mismoLargo(recursos,cooperan) ∧ listaNoVacia(recursos) ∧ esMatriz(pagos) ∧ esMatriz(eventos) ∧ apuestasValidas(apuestas) ∧ pagosValidos(pagos)}

Cómo son los eventos contra los pagos y apuestas ?
¿

asegura{
If (trayectoriaDeLosFrutosIndividualesALargoPlazo(recursos,SetAt(cooperan, individuo, true), apuestas, pagos, eventos)[individuo][|eventos|-1])
>

No se pueden utilizar procs !

(trayectoriaDeLosFrutosIndividualesALargoPlazo(recursos,SetAt(cooperan, individuo,false), apuestas, pagos, eventos)[individuo][|eventos|-1])

then SetAt(cooperan, individuo, true)
else SetAt(cooperan, individuo,false)
fi }

1,5) 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{ individuo < |recursos| ∧L mismoLargo(recursos, cooperan) ∧ esMatriz(apuestas) ∧ esMatriz(pagos) ∧ esMatriz(eventos) ∧ apuestasValidas(apuestas) ∧ pagosValidos(pagos) }

idem anterior

asegura{ (∀ i,j : R)((i + j = 1) ∧ (i ≥ 0) ∧ (j ≥ 0) ∧ (∃ a,b: R) ((a + b = 1) ∧ (a ≥0) ∧ (b≥ 0) ∧
(trayectoriaDeLosFrutosIndividualesALargoPlazo(recursos,cooperan, SetAt(apuestas, individuo, (a,b)), pagos, eventos)[individuo][|eventos|-1])
≥
(trayectoriaDeLosFrutosIndividualesALargoPlazo(recursos, cooperan, SetAt(apuestas, individuo, (i,j)), pagos, eventos)[individuo][|eventos|-1])))

No se pueden utilizar procs!

Anexo

pred listaNoVacia (in s: seq<T>) {|s| > 0}

pred todosPositivos (in s: seq<T>) {(∀i : Z) (0 ≤ i < |s|) →L(s[i] ≥ 0)}

pred mismoLargo (in s: seq<T>, in l: seq<T>) {|s| = |l|}

pred esMatriz (in s: seq<seq<T>>({(∀i : Z)(0 ≤ i < |m| →L |m[i]| > 0∧ (∀j : Z)(0 ≤ j < |m| →L |m[i]| = |m[j]|)) }

pred apuestaValida (s :seq⟨seq⟨R⟩⟩{ esMatriz(s) ∧((∀i : Z)(0 ≤ i < |s|→L (∀k: Z)(0 ≤ k < |s[i]|→L ∑s[i][k]=1 ∧ s [i][k]>0)}

pred pagosValidos (s :seq⟨seq⟨R⟩⟩{ esMatriz(s) ∧((∀i : Z)(0 ≤ i < |s|→L todosPositivos(s[i])}