



**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



**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>

requiere{listaNoVacia(recursos)  $\wedge$  L igualLargo(recursos, cooperan)}

asegura{igualLargo(res, recursos)  $\wedge$  L  
 $(\forall i : Z)(0 < i < |cooperan| \rightarrow L (cooperan[i] \rightarrow res[i] = totalRecursos/|cooperan| \vee cooperan[i] \rightarrow res[i] = totalRecursos/|cooperan|))$ }

aux TotalDeRecursos(cooperan :seq<Bool>,recursos :seq<R>) : R =  $\sum_{i=0}^{|recursos|-1} (If\ cooperan[i]\ then\ recursos[i]\ else\ 0\ fi)$  ;

**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)  $\wedge$  L esMatriz(trayectoria)  $\wedge$  esMatriz(pagos)  $\wedge$  L listaNoVacia(trayectoria)  $\wedge$  L igualLargo(trayectoria, cooperan, apuestas, pagos, eventos)  $\wedge$  apuestasValidas(apuestas)  $\wedge$  pagosValidos(pagos)}

asegura{igualLargo(res, trayectorias)  $\wedge$  L (  $(\forall j : Z) (|trayectorias[0]| \leq j < |eventos[0]|) \rightarrow L ((\forall i : Z)(0 \leq i < |trayectorias|) \wedge L resP[i] = trayectorias[i][j]-1*apuestas[i][eventos[i][j]]*pagos[i][eventos[i][j]]) \wedge L (trayectorias[i] = old(trayectorias[i]) ++ redistribucionDeLosFrutos (resP, cooperan)[i]))$ }

**1.3) proc trayectoriaExtrañaEscalera** ( in trayectoria : seq<R> ) : Bool

requiere {listaNoVacia(trayectoria)  $\wedge$  todosPositivos(trayectoria)}

asegura{ (  $\exists ! i : Z) ( (0 \leq i < |trayectoria|) \rightarrow L ((|trayectoria| = 1) \vee L ((i=0) \wedge (trayectoria[i] > trayectoria[i+1])) \vee L ((i=|trayectoria|-1) \wedge (trayectoria[i] > trayectoria[i-1])) \vee L ((trayectoria[i] > trayectoria[i+1]) \wedge (trayectoria[i] > trayectoria[i-1]))$  ) }

**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)  $\wedge$  listaNoVacia(recursos)  $\wedge$  esMatriz(pagos)  $\wedge$  esMatriz(eventos)  $\wedge$  apuestasValidas(apuestas)  $\wedge$  pagosValidos(pagos)}

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

```
(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|  $\wedge$  L mismoLargo(recursos, cooperan)  $\wedge$  esMatriz(apuestas)
 $\wedge$  esMatriz(pagos)  $\wedge$  esMatriz(eventos)  $\wedge$  apuestasValidas(apuestas)  $\wedge$  pagosValidos(pagos)
}
```

```
  asegura{ (  $\forall i, j : R$  ) ((i + j = 1)  $\wedge$  (i  $\geq$  0)  $\wedge$  (j  $\geq$  0)  $\wedge$  (  $\exists a, b : R$  ) ((a + b = 1)  $\wedge$  (a  $\geq$  0)  $\wedge$ 
(b  $\geq$  0)  $\wedge$ 
    (trayectoriaDeLosFrutosIndividualesALargoPlazo(recursos, cooperan,
SetAt(apuestas, individuo, (a,b)), pagos, eventos)[individuo][[eventos]-1])
     $\geq$ 
    (trayectoriaDeLosFrutosIndividualesALargoPlazo(recursos, cooperan,
SetAt(apuestas, individuo, (i,j)), pagos, eventos)[individuo][[eventos]-1])) )
```

## Anexo

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

```
pred todosPositivos (in s: seq<T>) { (  $\forall i : Z$  ) (0  $\leq$  i < |s|  $\rightarrow$  L(s[i]  $\geq$  0)) }
```

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

```
pred esMatriz (in s: seq<seq<T>>) { (  $\forall i : Z$  ) (0  $\leq$  i < |m|  $\rightarrow$  L(m[i] > 0  $\wedge$  (  $\forall j : Z$  ) (0  $\leq$  j < |m|
 $\rightarrow$  L(m[i] = m[j])) ) ) }
```

```
pred apuestaValida ( s : seq<seq<R>> ) { esMatriz(s)  $\wedge$  ( (  $\forall i : Z$  ) (0  $\leq$  i < |s|  $\rightarrow$  L (  $\forall k : Z$  ) (0  $\leq$  k <
|s[i]|  $\rightarrow$  L  $\sum$  s[i][k] = 1  $\wedge$  s[i][k] > 0)) }
```

```
pred pagosValidos ( s : seq<seq<R>> ) { esMatriz(s)  $\wedge$  ( (  $\forall i : Z$  ) (0  $\leq$  i < |s|  $\rightarrow$  L
todosPositivos(s[i])) }
```