



# 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⟨ℝ⟩, in cooperan: seq⟨Bool⟩) : seq⟨ℝ⟩
  requiere {−}
  asegura {−}
```

## 1.2. trayectoriaDeLosFrutosIndividualesALargoPlazo

```
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⟨ℝ⟩⟩)
  requiere {−}
  asegura {−}
```

## 1.3. trayectoriaExtrañaEscalera

```
proc trayectoriaExtrañaEscalera (in trayectorias: seq⟨ℝ⟩) : Bool
  requiere {−}
  asegura {−}
```

## 1.4. individuoDecideSiCooperarONo

```
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 {−}
  asegura {−}
```

## 1.5. individuoActualizaApuesta

```
proc individuoActualizaApuesta (in individuo: ℕ, in recursos: seq⟨ℝ⟩, in cooperan: seq⟨Bool⟩, inout apuestas: seq⟨seq⟨Bool⟩⟩, in pagos: seq⟨seq⟨ℝ⟩⟩, in eventos: seq⟨seq⟨ℕ⟩⟩)
  requiere {−}
  asegura {−}
```

# 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: ℝ, in apuesta: ⟨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 teórica, y #(eventos, T) es su abreviación.

```
1 |   res := recursos
2 |   i := 0
3 |   while (i < |eventos|) do
4 |     if eventos[i] then
5 |       res := (res * apuesta.c) * pago.c
6 |     else
7 |       res := (res * apuesta.s) * pago.s
8 |     endif
9 |     i := i + 1
10 |  endwhile
```

Lo principal: las fórmulas. Se puede poner en una línea, como  $x_i = x_{i-1} + x_{i-2}$ , o ponerse más grande:

$$\sum_{i=0}^n i \quad (1)$$

Y se pueden citar ecuaciones con `\eqref{nombreDeEq}`: (1)

Ejemplo de itemizado:

- Item 1
- Item 2
- Item 3

Ejemplo de enumerado con menor distancia entre items:

1. Item 1
2. Item 2
3. Item 3

Podemos escribir mucho texto. Mucho texto. Mucho texto. Mucho texto. Mucho texto. Mucho texto. Mucho texto. Mucho texto. Mucho texto. Mucho texto. Mucho texto. Mucho texto.

Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo. Otro párrafo.

Le agregamos una separación entre párrafos. Le agregamos una separación entre párrafos. Le agregamos una separación entre párrafos. Le agregamos una separación entre párrafos. Le agregamos una separación entre párrafos. Le agregamos una separación entre párrafos.

La tabla 1 es un ejemplo de cómo se hace una tabla.

Col1	Col2	Col2	Col3
1	6	87837	787
2	7	78	5415
3	545	778	7507
4	545	18744	7560
5	88	788	6344

Tabla 1: Ejemplo de tabla

La figura 2 es un ejemplo de cómo se agrega una imagen.



Figura 1: Ejemplo de figura

```

1 | r e s = r e c u r s o s
2 | i = 0
3 | w h i l e ( i < | e v e n t o s | ) d o
4 |     i f e v e n t o s [ i ] t h e n
5 |         r e s = ( r e s * a p u e s t a . c ) * p a g o . c
6 |     e l s e
7 |         r e s = ( r e s * a p u e s t a . s ) * p a g o . s
8 |     e n d i f
9 |     i = i + 1
10 | e n d w h i l e

```

Código 1: Ejemplo de código (usando los estilos de la cátedra, ver las macros para más detalles)

Si se pone un label al `\lstlisting`, se puede referenciar: Código 1.



(a) Logo de LaTeX



(b) Logo de TeX

Figura 2: Ejemplo para poner dos figuras juntas. Y citarlas por separado a (a) y (b).

## 2.1. Macros de la cátedra para especificar

```

proc nombre (in paramIn : N, inout paramInout : seq⟨Z⟩) : tipoRes
  requiere {expresionBooleana1}
  asegura {expresionBooleana2}
  aux auxiliar1 (parametros) : tipoRes = expresion ;
  pred pred1 (parametros) {
    expresion
  }

aux auxiliarSuelto (parametros) : tipoRes = expresion ;
pred predSuelto (parametros) {
  (∀variable : tipo) (algo →L expresion)
}
pred predSuelto (parametros) {
  (∃variable : tipo) (algo ∧L expresion)
}

hola «“¡HEAD chau ===== lawea
»”””»¿16195965f3010a1d75937740513beddf4d768ede

```