

Algoritmos y Estructuras de Datos II

Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

TP 1 - Broker System

Grupo 20

Integrante	LU	Correo electrónico
Fernando Gasperi Jabalera	56/09	fgasperijabalera@gmail.com
Esteban Romero	659/06	estebantaborcias@gmail.com
Leandro Tozzi	-	leandro.tozzi@gmail.com
Alfredo Terrile Cendoya	022/11	freddy199_0@hotmail.com

Reservado para la cátedra

Instancia	Docente	Nota
Primera entrega		
Segunda entrega		

Índice

1. TAD BROKER	3
2. TAD PROMESA	9
3. TAD TÍTULO	10

1. TAD BROKER

TAD BROKER

igualdad observacional

$$(\forall c, c' : \text{broker}) (c =_{\text{obs}} c' \iff ())$$

géneros broker

exporta broker, observadores, enAlza

usa BOOL, NAT

observadores básicos

títulos	: broker	\longrightarrow conj(título)	
promesas	: cliente $c \times$ broker	\longrightarrow conj(promesas)	$\{c \in \text{clientes}(b)\}$
clientes	: broker	\longrightarrow conj(clientes)	
acciones	: cliente $c \times$ nombre $n \times$ broker b	\longrightarrow nat	$\{\text{estáDefinido?}(n, \text{títulos}(b)) \wedge c \in \text{clientes}(b)\}$

generadores

nuevo	: conj(título) $ts \times$ conj(cliente) cs	\longrightarrow broker	$\{\neg \text{Vacío?}(cs) \wedge \neg \text{hayNombresRepetidos}(ts)\}$
CambiarCotización	: nombre $n \times$ cotización \times broker b	\longrightarrow broker	$\{\text{estáDefinido?}(n, \text{títulos}(b))\}$
AgPromesa	: cliente $c \times$ promesa $p \times$ broker b	\longrightarrow broker	$\left\{ \begin{array}{l} c \in \text{clientes}(b) \wedge \text{estáDefinido?}(\text{nombreTítulo}(p), \text{títulos}(b)) \wedge_L \\ \text{cantidad}(p) \leq \text{máximoAcciones}(\text{nombreTítulo}(p), \text{títulos}(b)) \wedge \\ \neg \text{HayPromMismoTipo}(\text{nombreTítulo}(p), \text{promesas}(c, b)) \wedge_L (\text{vende?}(p) \Rightarrow \text{puedeVender?}(c, p, b)) \end{array} \right\}$

otras operaciones

hayNombresRepetidos	: conj(título) ts	\longrightarrow bool	
máximoAcciones	: nombre $n \times$ conj(título) ts	\longrightarrow nat	
enAlza	: nombre $n \times$ broker b	\longrightarrow bool	$\{\text{estáDefinido?}(n, \text{títulos}(b))\}$
HayPromMismoTipo	: promesa $p \times$ conj(promesa) $ps \times$ título t	\longrightarrow bool	
cumplirPromesas	: cliente $c \times$ conj(promesa) $ps \times$ nombre \times cotización \times nat \times broker b	\longrightarrow conj(promesa)	
dameMáximo	: nombre \times conj(título)	\longrightarrow nat	
accionesLibres	: título $t \times$ broker b	\longrightarrow nat	
accionesTomadas	: nombre \times conj(cliente) \times broker	\longrightarrow nat	
cumplirVentas	: conj(promesa) \times nombre \times cotización	\longrightarrow conj(promesa)	
existeVenta	: promesa \times nombre \times cotización	\longrightarrow bool	
accLibresDpsVenta	: conj(cliente) \times nombre \times cotización \times nat \times broker	\longrightarrow nat	
vendióAcciones	: conj(promesa) \times nombre \times cotización	\longrightarrow bool	
nuevasAccLibres	: conj(promesa) \times nombre \times cotización	\longrightarrow nat	
cumplirCompras	: cliente \times conj(promesa) \times título \times cotización \times conj(clientes) \times acciones \times broker	\longrightarrow conj(promesas)	
accDspDeComprasDeOtros	: conj(promesa) \times nombre \times cotización \times nat	\longrightarrow nat	
promDspDeVerSiCompro	: conj(promesa) \times nombre \times cotización \times nat	\longrightarrow conj(promesa)	
puedeVender?	: cliente $c \times$ promesa $p \times$ broker b	\longrightarrow bool	

<code>estáDefinido?</code>	<code>: nombre × conj(título)</code>	<code>→ bool</code>
<code>seCumple?</code>	<code>: promesa × broker</code>	<code>→ bool</code>
<code>dameCotización</code>	<code>: nombre × conj(título)</code>	<code>→ nat</code>
<code>títuloNuevaCot</code>	<code>: nombre × cotización × conj(título)</code>	<code>→ conj(título)</code>
<code>accClienteDspVenta</code>	<code>: nat × conj(promesa) × nombre × cotización</code>	<code>→ nat</code>
<code>accDspDeCompras</code>	<code>: cliente × nombre × conj(promesas) × conj(cliente) × nat × nat × broker</code>	<code>→ nat</code>
<code>accClienteDspCompras</code>	<code>: conj(promesa) × nombre × cotización × nat × nat</code>	<code>→ nat</code>

axiomas

$\text{promesas}(c, \text{nuevo}(ts, cs))$	$\equiv \emptyset$
$\text{promesas}(c, \text{AgPromesa}(c_1, p, b))$	$\equiv \text{if seCumple?}(p, b) \text{ then } \text{promesas}(c, b) \text{ else } \text{if } c = c_1 \text{ then } \text{Ag}(p, \text{promesas}(c, b)) \text{ else } \text{promesas}(c, b) \text{ fi}$
$\text{promesas}(c, \text{cambiarCot}(n, ct, b))$	$\equiv \text{cumplirPromesas}(c, \text{promesas}(c, b), n, ct, \text{accionesLibres}(n, b), b)$
$\text{títulos}(\text{nuevo}(ts, cs))$	$\equiv ts$
$\text{títulos}(\text{AgTítulo}(t, b))$	$\equiv \text{Ag}(t, \text{títulos}(b))$
$\text{títulos}(\text{cambiarCot}(n, ct, b))$	$\equiv \text{títuloNuevaCot}(n, ct, \text{títulos}(b))$
$\text{clientes}(\text{nuevo}(ts, cs))$	$\equiv cs$
$\text{clientes}(\text{cambiarCot}(n, ct, b))$	$\equiv \text{clientes}(b)$
$\text{clientes}(\text{AgPromesa}(c, p, b))$	$\equiv \text{clientes}(b)$
$\text{clientes}(\text{AgTítulo}(t, b))$	$\equiv \text{clientes}(b)$
$\text{acciones}(c, t, \text{nuevo}(ts, cs))$	$\equiv 0$
$\text{acciones}(c, t, \text{CambiarCot}(n, ct, b))$	$\equiv \text{if } t = n \text{ then } \text{accDspDeCompras}(c, n, \text{promesas}(c, b), \text{clientes}(b), \text{accClienteDspVenta}(\text{acciones}(c, t, b), \text{promesas}(c, t, b), n, ct), \text{accLibresDspVenta}(\text{clientes}(b), n, ct, \text{accionesLibres}(n, b), b), b) \text{ else } \text{acciones}(c, t, b) \text{ fi}$
$\text{acciones}(c, n, \text{AgPromesa}(c_1, p, b))$	$\equiv \text{if } c = c_1 \wedge \text{seCumple?}(p, b) \text{ then } \text{if vende?}(p) \text{ then } \text{acciones}(c, n, b) - \text{cantidad}(p) \text{ else } \text{acciones}(c, n, b) + \text{cantidad}(p) \text{ fi} \text{ else } \text{acciones}(c, n, b) \text{ fi}$
$\text{hayNombresRepetidos?}(ts)$	$\equiv \text{if vacio?}(ts) \text{ then } \text{false} \text{ else } \text{if estaDefinido}(\text{nombre}(\text{dameUno}(ts)), \text{sinUno}(ts)) \text{ then } \text{true} \text{ else } \text{hayNombresRepetidos}(\text{sinUno}(ts)) \text{ fi} \text{ fi}$
$\text{maximoAcciones}(n, ts)$	$\equiv \text{if } \text{nombre}(\text{dameUno}(ts)) = n \text{ then } \text{maximo}(\text{dameUno}(ts)) \text{ else } \text{maximoAcciones}(n, \text{sinUno}(ts)) \text{ fi}$
$\text{puedeVender?}(c, p, b)$	$\equiv \text{cantidad}(p) \leq \text{acciones}(c, \text{nombre}(p), b)$

HayPromMismoTipo(p, ps)	<pre> ≡ if vacío?(ps) then false else if vende?(dameUno(ps))=vende?(p) ∧ nombre(dameUno(ps))=nombre(p) then true else HayPromMismoTipo($p, sinUno(ps)$) fi fi </pre>
estáDefinido?(n, ts)	<pre> ≡ if vacío?(ts) then false else if nombre(dameUno(ts))= n then true else estáDefinido?($n, sinUno(ts)$) fi fi </pre>
seCumple?(p, b)	<pre> ≡ if vende?(p) then dameCotización(nombreTítulo(p), títulos(b)) < límite(p) else if accionesLibres(nombreTítulo(p), b) > cantidad(p) then dameCotización(nombre(p), títulos(b)) > límite(p) else false fi fi </pre>
dameCotización(n, ts)	<pre> ≡ if nombre(dameUno(ts)) = n then cotización(dameUno(ts)) else dameCotización($n, sinUno(ts)$) fi </pre>
accionesLibres(n, b)	<pre> ≡ dameMáximo(n, títulos(b)) - accionesTomadas(n, clientes(b), b) </pre>
dameMáximo(n, ts)	<pre> ≡ if nombre(dameUno(ts)) = n then máximo(dameUno(ts)) else dameMáximo($n, sinUno(ts)$) fi </pre>
accionesTomadas(n, cs, b)	<pre> ≡ if vacío?(cs) then 0 else acciones(dameUno(cs), n, b) + accionesTomadas(n, sinUno(cs), b) fi </pre>
cumplirPromesas(c, ps, t, ct, a, b)	<pre> ≡ cumplirCompras(c, cumplirVentas(ps, t, ct), t, ct, clientes(b), accLibresDespVenta(clientes(b), t, ct, a, b), b) </pre>
cumplirVentas(ps, n, ct)	<pre> ≡ if vacío?(ps) then ∅ else if existeVenta(dameUno(ps), n, ct) then sinUno(ps) else Ag(dameUno(ps), cumplirVentas(sinUno(ps), n, ct)) fi fi </pre>
existeVenta(p, n, ct)	<pre> ≡ vende?(p) ∧ nombre(p) = n ∧ ct < límite(p) </pre>

```

accLibresDspVenta(cs, n, ct, a, b)    ≡ if vacío?(cs) then
    a
  else
    if vendióAcciones(promesas(dameUno(cs), b), n, ct) then
      nuevasAccLibres(promesas(dameUno(cs), b), n, ct) +
      accLibresDspVenta(sinUno(cs), n, ct, a, b)
    else
      accLibresDspVenta(p, sinUno(cs), n, ct, a)
    fi
  fi

vendióAcciones(ps, n, ct)             ≡ if vacío?(ps) then
    false
  else
    if vende?(dameUno(p)) ∧ nombre(dameUno(ps)) = n ∧
    límite(dameUno(ps)) < ct then
      true
    else
      vendióAcciones(sinUno(ps), n, ct)
    fi
  fi

nuevasAccLibres(ps, n, ct)            ≡ if vacío?(ps) then
    0
  else
    if vende?(dameUno(ps)) ∧ nombre(dameUno(ps)) = n ∧
    límite(dameUno(ps)) < ct then
      cantidad(dameUno(ps))
    else
      nuevasAccLibres(sinUno(p), n, ct)
    fi
  fi

cumplirCompras(c, ps, t, ct, cs, a, b) ≡ if dameUno(cs) = c then
    promDspDeVerSiCompro(ps, t, ct, a)
  else
    cumplirCompras(c, ps, t, ct, sinUno(cs),
    accDspDeComprasDeOtros(promesas(dameUno(cs), b), t, ct,
    a), b)
  fi

accDspDeComprasDeOtros(ps, n, ct, a) ≡ if vacío?(ps) then
    a
  else
    if ¬vende?(dameUno(ps)) ∧ nombre(dameUno(ps)) = n ∧
    límite(dameUno(ps)) < ct then
      a - cantidad(p)
    else
      accDspDeComprasDeOtros(sinUno(p), n, ct, a)
    fi
  fi

promDspDeVerSiCompro(ps, n, ct, a)    ≡ if vacío?(ps) then
    ∅
  else
    if ¬vende?(dameUno(ps)) ∧ nombre(dameUno(ps)) = n ∧
    límite(dameUno(ps)) < ct ∧ a ≥ cantidad(dameUno(ps))
    then
      sinUno(ps)
    else
      Ag(dameUno(ps), promDspDeVerSiCompro(sinUno(ps), n,
      ct, a))
    fi
  fi

```

```

títuloNuevaCot(n, ct, ts)           ≡ if nombre(dameUno(ts)) = n then
                                     Ag(cambiarValor(ct,dameUno(ts)),sinUno(ts))
                                     else
                                     Ag(dameUno(ts),títuloNuevaCot(n,ct,sinUno(ts))
                                     fi

accClienteDspVenta(a, ps, n, ct)    ≡ if vacío?(ps) then
                                     a
                                     else
                                     if existeVenta(dameUno(ps),t,ct) then
                                     a - cantidad(dameUno(ps))
                                     else
                                     accClienteDspVenta(a,sinUno(ps),n,ct)
                                     fi
                                     fi

accDspDeCompras(c, n, ps, cs, a, av, b) ≡ if dameUno(cs) = c then
                                     accClienteDspCompras(ps, t, ct, a, av)
                                     else
                                     accDspDeCompras(c,      n,      ps,      sinUno(cs),      ct,
                                     accDspDeComprasDeOtros(promesas(dameUno(cs),  b),  t,
                                     ct, av), b)
                                     fi

accClienteDspCompras(ps, n, ct, a, av) ≡ if vacío(ps) then
                                     a
                                     else
                                     if ¬vende?(dameUno(ps)) ∧ nombre(dameUno(ps)) = n ∧
                                     límite(dameUno(ps)) < c ∧ av ≤ cantidad(dameUno(ps))
                                     then
                                     a + cantidad(dameUno(ps))
                                     else
                                     promDspDeVerSiCompro(sinUno(ps), n, ct, a, av)
                                     fi
                                     fi

```

Fin TAD

2. TAD PROMESA

TAD PROMESA

igualdad observacional

$$(\forall p, p' : \text{promesa}) \left(p =_{\text{obs}} p' \iff \left(\begin{array}{l} \text{nombreTitulo}(p) =_{\text{obs}} \text{nombreTitulo}(p') \wedge \text{vende?}(p) =_{\text{obs}} \text{vende?}(p') \\ \text{vende?}(p) \wedge \text{cantidad}(p) =_{\text{obs}} \text{cantidad}(p') \wedge \\ \text{limite}(p) =_{\text{obs}} \text{limite}(p') \end{array} \right) \right)$$

géneros promesa

exporta observadores

usa BOOL, NAT

observadores básicos

límite	: promesa	→ nat
nombreTitulo	: promesa	→ nombre
vende?	: promesa	→ bool
cantidad	: promesa	→ nat

generadores

nueva	: límite × título × bool × nat	→ promesa
-------	--------------------------------	-----------

axiomas $\forall l, c: \text{nat}, \forall v: \text{bool}, \forall t: \text{titulo}$

límite(nueva(l, t, v, c))	$\equiv l$
nombreTitulo(nueva(l, t, v, c))	$\equiv t$
vende?(nueva(l, t, v, c))	$\equiv v$
cantidad(nueva(l, t, v, c))	$\equiv c$

Fin TAD

