



UNIVERSIDAD DE BUENOS AIRES

Departamento de Computación

Facultad de Ciencias Exactas y Naturales

Algoritmos y Estructuras de Datos II

Primer cuatrimestre de 2013

Trabajo Práctico de TADS

17 de abril de 2013

Integrante	LU	Correo electrónico
Almansi, Emilio Guido	674/12	ealmansi@gmail.com
Vilerino, Silvio Fernando	106/12	svilerino@gmail.com
Chapresto, Matias Nahuel	201/12	matiaschapresto@gmail.com
Nasif, Lucas	391/06	nasif.lucas@gmail.com

Índice

1. Renombres	3
2. TAD GREMIO	4
3. TAD REUNION	5
4. TAD HISTORIAL	6
5. TAD DATOS SISTEMA PARITARIAS	8
6. TAD SISTEMA PARITARIAS	10

1. Renombres

TAD NOMBRE es STRING

TAD TRABAJADOR es STRING

TAD EMPRESA es STRING

TAD ALIANZA es CONJUNTO(GREMIO)

2. TAD GREMIO

TAD GREMIO

igualdad observacional

$$(\forall g, g' : \text{gremio}) \left(g =_{\text{obs}} g' \iff \left(\begin{array}{l} \text{nombre}(g) =_{\text{obs}} \text{nombre}(g') \wedge \\ \text{trabajadores}(g) =_{\text{obs}} \text{trabajadores}(g') \wedge \\ \text{empresas}(g) =_{\text{obs}} \text{empresas}(g') \end{array} \right) \right)$$

géneros gremio

exporta gremio, generadores, observadores, cantTrabajadores, nombresDistintos, empresasDisjuntas, trabajadoresDisjuntos

usa BOOL, NAT, CONJUNTO(α)

generadores

CrearGremio : nombre $n \times \text{conj}(\text{trabajador}) \text{ ct} \times \text{conj}(\text{empresa}) \text{ ce} \longrightarrow \text{gremio} \quad \{\neg \emptyset?(ct) \wedge \neg \emptyset?(ce)\}$

observadores básicos

nombre : gremio \longrightarrow nombre

trabajadores : gremio $\longrightarrow \text{conj}(\text{trabajador})$

empresas : gremio $\longrightarrow \text{conj}(\text{empresa})$

otras operaciones

cantTrabajadores : gremio $\longrightarrow \text{nat}$

nombresDistintos : gremio \times gremio $\longrightarrow \text{bool}$

empresasDisjuntas : gremio \times gremio $\longrightarrow \text{bool}$

trabajadoresDisjuntos : gremio \times gremio $\longrightarrow \text{bool}$

axiomas

$\forall n : \text{nat}, \forall ct : \text{conj}(\text{trabajador}), \forall ce : \text{conj}(\text{empresa}), \forall g : \text{gremio}$

$\text{nombre}(\text{CrearGremio}(n, ct, ce)) \equiv n$

$\text{trabajadores}(\text{CrearGremio}(n, ct, ce)) \equiv ct$

$\text{empresas}(\text{CrearGremio}(n, ct, ce)) \equiv ce$

$\text{cantTrabajadores}(g) \equiv \#(\text{trabajadores}(g))$

$\text{nombresDistintos}(g1, g2) \equiv \text{nombre}(g1) \neq \text{nombre}(g2)$

$\text{empresasDisjuntas}(g1, g2) \equiv \emptyset?(\text{empresas}(g1) \cap \text{empresas}(g2))$

$\text{trabajadoresDisjuntos}(g1, g2) \equiv \emptyset?(\text{trabajadores}(g1) \cap \text{trabajadores}(g2))$

Fin TAD

3. TAD REUNION

TAD REUNION

igualdad observacional

$$(\forall r, r' : \text{reunion}) \left(r =_{\text{obs}} r' \iff \begin{pmatrix} \text{gremio}(r) =_{\text{obs}} \text{gremio}(r') \wedge \\ \text{empresas}(r) =_{\text{obs}} \text{empresas}(r') \wedge \\ \text{gremioPidio}(r) =_{\text{obs}} \text{gremioPidio}(r') \wedge \\ \text{empresasOfrecieron}(r) =_{\text{obs}} \text{empresasOfrecieron}(r') \wedge \\ \text{huboAcuerdo}(r) =_{\text{obs}} \text{huboAcuerdo}(r') \wedge \\ (\text{huboAcuerdo}(r) \Rightarrow_{\text{L}} \text{aumento}(r) =_{\text{obs}} \text{aumento}(r')) \end{pmatrix} \right)$$

géneros reunion

exporta reunion, observadores, generadores

usa NAT, BOOL, CONJUNTO(α)

observadores básicos

gremio : reunion \longrightarrow nombre

empresas : reunion \longrightarrow conj(empresa)

gremioPidio : reunion \longrightarrow nat

empresasOfrecieron : reunion \longrightarrow nat

huboAcuerdo : reunion \longrightarrow bool

aumento : reunion \longrightarrow nat

{huboAcuerdo(r)}

generadores

ReunionSinAcuerdo : nombre \times conj(empresa) \times nat \times nat \longrightarrow reunion

ReunionConAcuerdo : nombre \times conj(empresa) \times nat \times nat \times nat \longrightarrow reunion

axiomas

\forall ng : nombre, \forall ce : conj(empresa), \forall gp : nat, \forall eo : nat, \forall au : nat

gremio(ReunionSinAcuerdo(ng, ce, gp, eo)) \equiv ng

gremio(ReunionConAcuerdo(ng, ce, gp, eo, au)) \equiv ng

empresas(ReunionSinAcuerdo(ng, ce, gp, eo)) \equiv ce

empresas(ReunionConAcuerdo(ng, ce, gp, eo, au)) \equiv ce

gremioPidio(ReunionSinAcuerdo(ng, ce, gp, eo)) \equiv gp

gremioPidio(ReunionConAcuerdo(ng, ce, gp, eo, au)) \equiv gp

empresasOfrecieron(ReunionSinAcuerdo(ng, ce, gp, eo)) \equiv eo

empresasOfrecieron(ReunionConAcuerdo(ng, ce, gp, eo, au)) \equiv eo

huboAcuerdo(ReunionSinAcuerdo(ng, ce, gp, eo)) \equiv false

huboAcuerdo(ReunionConAcuerdo(ng, ce, gp, eo, au)) \equiv true

aumento(ReunionConAcuerdo(ng, ce, gp, eo, au)) \equiv au

Fin TAD


```
cantidadDeReaperturas(hi)  ≡  if algunaVezAbrio(hi) then  
                                cantidadDeAperturas(hi) - 1  
                                else  
                                0  
                                fi
```

Fin TAD

5. TAD DATOS SISTEMA PARITARIAS

A partir de un conjunto de alianzas (o, lo que es igual, conjunto de conjuntos de gremios), se determinan todos los gremios que pertenecen al sistema de paritarias y cuáles son las relaciones de alianza entre ellos. En cada conjunto de gremios, se considera que cada uno de los elementos está aliado con todos los del grupo (salvo consigo mismo). Por construcción, esto garantiza que la relación de alianzas sea transitiva y simétrica, y sin casos reflexivos. Además, la totalidad de los gremios del sistema es simplemente la enumeración de todos los gremios que pertenecen a alguno de los conjuntos. Esto también admite tener gremios sin aliados, ya que se los puede incluir como elementos únicos en un conjunto.

TAD DATOS SISTEMA PARITARIAS

igualdad observacional

$$(\forall d, d' : \text{datos}) (d =_{\text{obs}} d' \iff (\text{alianzas}(d) =_{\text{obs}} \text{alianzas}(d')))$$

géneros datos

exporta datos, observadores, generadores, gremiosRegistrados, aliadosRegistrados, aplanar

usa BOOL, CONJUNTO(α), GREMIO

observadores básicos

alianzas : datos \longrightarrow conj(alianza)

generadores

CrearDatos : conj(alianza) $ca \longrightarrow$ datos

$\{\neg \emptyset?(ca) \wedge$

$(\forall a : \text{alianza} \mid a \in ca) \neg \emptyset?(a) \wedge$

$(\forall a1 : \text{alianza} \mid a1 \in ca) (\forall a2 : \text{alianza} \mid a2 \in ca - \{a1\}) \emptyset?(a1 \cap a2) \wedge$

$(\forall g1 : \text{gremio} \mid g1 \in \text{aplanar}(ca)) (\forall g2 : \text{gremio} \mid g2 \in \text{aplanar}(ca) - \{g1\}) \text{nombresDistintos}(g1, g2) \wedge$

$(\forall g1 : \text{gremio} \mid g1 \in \text{aplanar}(ca)) (\forall g2 : \text{gremio} \mid g2 \in \text{aplanar}(ca) - \{g1\}) \text{empresasDisjuntas}(g1, g2) \wedge$

$(\forall g1 : \text{gremio} \mid g1 \in \text{aplanar}(ca)) (\forall g2 : \text{gremio} \mid g2 \in \text{aplanar}(ca) - \{g1\}) \text{trabajadoresDisjuntos}(g1, g2) \}$

otras operaciones

gremiosRegistrados : datos \longrightarrow conj(gremio)

aplanar : conj(alianza) \longrightarrow conj(gremio)

aliadosRegistrados : nombre \times datos \longrightarrow conj(nombre)

aliadosRegistradosAux : nombre \times conj(alianza) \longrightarrow conj(nombre)

obtenerNombres : conj(gremio) \longrightarrow conj(nombre)

axiomas

$\forall ca : \text{conj(alianza)}, \forall cg : \text{conj(gremio)}, \forall d : \text{datos}, \forall ng : \text{nombre}$

$\text{alianzas}(\text{CrearDatos}(ca)) \equiv ca$

$\text{gremiosRegistrados}(d) \equiv \text{aplanar}(\text{alianzas}(d))$

$\text{aplanar}(ca) \equiv \text{if } \emptyset?(ca) \text{ then}$

\emptyset

else

$\text{dameUno}(ca) \cup \text{aplanar}(\text{sinUno}(ca))$

fi

$\text{aliadosRegistrados}(ng, d) \equiv \text{aliadosRegistradosAux}(ng, \text{alianzas}(d))$


```
aliadosRegistradosAux(ng, ca)  ≡ if  $\emptyset?(ca)$  then
                                    $\emptyset$ 
                                   else
                                       if  $ng \in \text{obtenerNombres}(\text{dameUno}(ca))$  then
                                            $\text{obtenerNombres}(\text{dameUno}(ca)) - \{ ng \}$ 
                                       else
                                            $\text{aliadosRegistradosAux}(g, \text{sinUno}(ca))$ 
                                       fi
                                   fi

obtenerNombres(cg)  ≡ if  $\emptyset?(cg)$  then
                                    $\emptyset$ 
                                   else
                                        $\text{Ag}(\text{nombre}(\text{dameUno}(cg)), \text{obtenerNombres}(\text{sinUno}(cg)))$ 
                                   fi
```

Fin TAD

6. TAD SISTEMA_{PARITARIAS}

Para generar una instancia de *SistemaParitarias* es necesario proveer la información estática del mismo a través de una instancia de *DatosSistemaParitarias*. Después, la dinámica de las paritarias se maneja (re)abriendo paritarias mediante el generador *AbrirParitaria*, o registrando el acontecimiento de una reunión mediante el generador *HuboReunion*. Las operaciones *empresasNegociando* y *gremiosNegociando* permiten conocer las empresas y gremios con paritarias abiertas, y la operación *gremioInconformista* permite conocer alguno de los gremios que más veces reabrió paritarias.

TAD SISTEMA_{PARITARIAS}

igualdad observacional

$$(\forall si, si' : \text{sistema}) \left(si =_{\text{obs}} si' \iff \left(\begin{array}{l} \text{gremios}(si) =_{\text{obs}} \text{gremios}(si') \wedge \\ ((\forall ng : \text{nombre} \mid ((\exists g : \text{gremio} \mid g \in \text{gremios}(si)) \mid \\ \text{nombre}(g) = ng)) \text{aliados}(ng, si) =_{\text{obs}} \text{aliados}(ng, si')) \\ \wedge \text{historial}(ng, si) =_{\text{obs}} \text{historial}(ng, si')) \end{array} \right) \right)$$

géneros sistema

exporta sistema, observadores, generadores, gremiosNegociando, empresasNegociando, #trabajadoresNegociando, gremioInconformista

usa BOOL, NAT, DATOS, HISTORIAL, CONJUNTO(α)

observadores básicos

gremios : sistema \longrightarrow conj(gremio)

aliados : nombre $ng \times$ sistema $si \longrightarrow$ conj(nombre)

$$\{ (\exists g : \text{gremio} \mid g \in \text{gremios}(si) \mid \text{nombre}(g) = ng) \}$$

historial : nombre $ng \times$ sistema $si \longrightarrow$ historial

$$\{ (\exists g : \text{gremio} \mid g \in \text{gremios}(si) \mid \text{nombre}(g) = ng) \}$$

generadores

CrearSistema : datos $d \longrightarrow$ sistema

AbrirParitaria : nombre $ng \times$ sistema $si \longrightarrow$ sistema $\{ \text{cerrado}(\text{historial}(ng, si)) \}$

HuboReunion : nombre $ng \times$ reunion $rn \times$ sistema $si \longrightarrow$ sistema $\{ \text{abierto}(\text{historial}(ng, si)) \wedge ((\forall nga : \text{nombre} \mid nga \in \text{aliados}(ng, si)) \mid \text{gremioPidio}(rn) > \text{maximoAumento}(\text{historial}(nga, si))) \}$

otras operaciones

gremiosNegociando : sistema $si \longrightarrow$ conj(gremio)

gremiosNegociandoAux : conj(gremio) $cg \times$ sistema $si \longrightarrow$ conj(gremio)

empresasNegociando : sistema $si \longrightarrow$ conj(empresa)

empresasNegociandoAux : conj(gremio) $cg \longrightarrow$ conj(gremio)

#trabajadoresNegociando : sistema $si \longrightarrow$ nat

#trabajadoresNegociandoAux : conj(gremio) $cg \longrightarrow$ conj(gremio)

gremioInconformista : sistema $si \longrightarrow$ gremio

gremioInconformistaAux : conj(gremio) $cg \times$ sistema $si \longrightarrow$ nat

cantReaperturaGremio : gremio $g \times$ sistema $si \longrightarrow$ nat

maximaCantReapertura : sistema $si \longrightarrow$ nat

maximaCantReaperturaAux : conj(gremio) $cg \times$ sistema $si \longrightarrow$ nat

hayQueReabrirHistorial : nombre $ng1 \times$ nombre $ng2 \times$ reunion $rng2 \times$ sistema $si \longrightarrow$ bool

axiomas

$\forall d : \text{datos}, \forall g : \text{gremio}, \forall rn : \text{reunion}, \forall ng : \text{nombre}, \forall ng1 : \text{nombre}, \forall ng2 : \text{nombre}, \forall cg : \text{conj(gremio)}, \forall si : \text{sistema}$

```

gremios( CrearSistema(d) )      ≡ gremiosRegistrados(d)
gremios( AbrirParitaria(g, si) ) ≡ gremios(si)
gremios( HuboReunion(g, rn, si) ) ≡ gremios(si)

aliados(ng, CrearSistema(d) )    ≡ aliadosRegistrados(ng, d)
aliados(ng1, AbrirParitaria(ng2, si) ) ≡ aliados(ng1, si)
aliados(ng1, HuboReunion(ng2, rn, si) ) ≡ aliados(ng1, si)

historial(ng, CrearSistema(d) )    ≡ CrearHistorial
historial(ng1, AbrirParitaria(ng2, si) ) ≡ if ng1 = ng2 then
                                         AbrirHistorial( historial(ng1, si) )
                                         else
                                         historial(ng1, si)
                                         fi

historial(ng1, HuboReunion(ng2, rn, si) ) ≡ if ng1 = ng2 then
                                         RegistrarReunion(rn, historial(ng1, si) )
                                         else
                                         if hayQueReabrirHistorial(ng1, ng2, rn, si) then
                                             AbrirHistorial( historial(ng1, si) )
                                         else
                                             historial(ng1, si)
                                         fi
                                         fi

gremiosNegociando(si)            ≡ gremiosNegociandoAux(gremios(si), si)
gremiosNegociandoAux(cg, si) ≡ if vacio?(cg) then
                                ∅
                                else
                                if abierto(historial(nombre(dameUno(cg)), si)) then
                                    Ag(dameUno(cg), gremiosNegociandoAux(sinUno(cg), si))
                                else
                                    gremiosNegociandoAux(sinUno(cg), si)
                                fi
                                fi

empresasNegociando(si)          ≡ empresasNegociandoAux(gremiosNegociando(si))
empresasNegociandoAux(cg) ≡ if vacio?(cg) then
                                ∅
                                else
                                empresas(dameUno(cg)) ∪ empresasNegociandoAux(sinUno(cg))
                                fi

#trabajadoresNegociando(si)      ≡ #trabajadoresNegociandoAux(gremiosNegociando(si))

```

```

#trabajadoresNegociandoAux(cg)  ≡ if vacio?(cg) then
                                0
                                else
                                cantTrabajadores(dameUno(cg))  +  #trabajadoresNegociandoAux(sinUno(cg))
                                fi

gremioInconformista(si)         ≡ gremioInconformistaAux(gremios(si), si)
gremioInconformistaAux(cg, si)  ≡ if cantReaperturasGremio(dameUno(cg), si) = maximaCantReaperturas(si) then
                                dameUno(cg)
                                else
                                gremioInconformistaAux(sinUno(cg), si)
                                fi

cantReaperturasGremio(g, si)    ≡ cantidadDeReaperturas(historial(nombre(g), si))
maximaCantReaperturas(si)(g, si) ≡ maximaCantReaperturasAux(gremios(si), si)
maximaCantReaperturasAux(cg, si) ≡ if ∅?( cg ) then
                                0
                                else
                                max(cantReaperturasGremio(dameUno(cg), si), maximaCantReaperturasAux(sinUno(cg), si))
                                fi

hayQueReabrirHistorial(ng1, ng2, rng2, si) ≡ cerrado(historial(ng1, si)) ∧ ng1 ∈ aliados(ng2, si) ∧
                                huboAcuerdo(rng2)

```

Fin TAD