



# TP2

2 de mayo de 2021

Algoritmos y Estructuras de Datos II

## Grupo: 6

Integrante	LU	Correo electrónico
Collasius, Federico	164/20	fede.collasius@gmail.com
Fernández Olivares Esnaola, Joaquín	11/20	joaquinfernandezolivares@gmail.com
Totaro, Facundo Ariel	43/20	facutotaro@gmail.com
Venturini, Julia	159/20	juliaaventurini00@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>

# TAD Tablero

géneros tablero

igualdad observacional

$$(\forall t1, t2 : \text{tablero}) \quad t1 =_{\text{obs}} t2 \iff \left( \begin{array}{l} \text{tamaño}(t1) =_{\text{obs}} \\ \text{tamaño}(t2) \Rightarrow_{\text{L}} \\ (\forall pos_X, pos_Y : \text{nat}) \\ ( \\ ( \\ pos_X < \pi_1(\text{tamaño}(t1)) \\ \wedge \\ pos_Y < \pi_2(\text{tamaño}(t1)) \\ ) \\ \Rightarrow_{\text{L}} \\ (\text{esEntrada?}(pos_X, pos_Y, t1) \\ =_{\text{obs}} \\ \text{esEntrada?}(pos_X, pos_Y, t2) \\ \wedge \\ \text{esLlegada?}(pos_X, pos_Y, t1) \\ =_{\text{obs}} \\ \text{esLlegada?}(pos_X, pos_Y, t2) \\ \wedge \\ \text{esPared?}(pos_X, pos_Y, t1) \\ =_{\text{obs}} \\ \text{esPared?}(pos_X, pos_Y, t2) \\ \wedge \\ \text{esFantasma?}(pos_X, pos_Y, t1) \\ =_{\text{obs}} \\ \text{esFantasma?}(pos_X, pos_Y, t2) \\ \wedge \\ \text{esChocolate?}(pos_X, pos_Y, t1) \\ =_{\text{obs}} \\ \text{esChocolate?}(pos_X, pos_Y, t2) \\ ) \end{array} \right)$$

**usa**      Nat, Bool, Tupla

**exporta** genero, observadores, generadores, otras operaciones

## observadores básicos

tamaño	: tablero	→	tupla(nat, nat)
esEntrada?	: nat pos <sub>X</sub> × nat pos <sub>Y</sub> × tablero t	→	bool
esLlegada?	: nat pos <sub>X</sub> × nat pos <sub>Y</sub> × tablero t	→	bool
esPared?	: nat pos <sub>X</sub> × nat pos <sub>Y</sub> × tablero t	→	bool
esFantasma?	: nat pos <sub>X</sub> × nat pos <sub>Y</sub> × tablero t	→	bool
esChocolate?	: nat pos <sub>X</sub> × nat pos <sub>Y</sub> × tablero t	→	bool

## generadores

crearTablero : nat *fil* × nat *col* × nat *en<sub>X</sub>* × nat *en<sub>Y</sub>* × nat *ll<sub>X</sub>* × nat *ll<sub>Y</sub>* → tablero  
 $\{en_X, ll_X < fil \wedge en_Y, ll_Y < col \wedge (en_X \neq ll_X \vee en_Y \neq ll_Y)\}$

AgF : nat pos<sub>X</sub> × nat pos<sub>Y</sub> × tablero t → tablero  
 $\left\{ \begin{array}{l} pos_X < \pi_1(\text{tamaño}(t)) \wedge pos_Y < \pi_2(\text{tamaño}(t)) \wedge_{\text{L}} (\neg \text{esLlegada?}(pos_X, pos_Y, t) \wedge \neg \text{esSalida?}(pos_X, pos_Y, t) \wedge \\ \neg \text{esPared?}(pos_X, pos_Y, t) \wedge \neg \text{esChocolate?}(pos_X, pos_Y, t)) \end{array} \right\}$

AgP : nat pos<sub>X</sub> × nat pos<sub>Y</sub> × tablero t → tablero  
 $\left\{ \begin{array}{l} pos_X \upharpoonright \pi_1(\text{tamaño}(t)) \wedge pos_Y < \pi_2(\text{tamaño}(t)) \wedge_{\text{L}} (\neg \text{esLlegada?}(pos_X, pos_Y, t) \wedge \neg \text{esFantasma?}(pos_X, pos_Y, t) \wedge \\ \neg \text{esChocolate?}(pos_X, pos_Y, t) \wedge \neg \text{esSalida?}(pos_X, pos_Y, t)) \end{array} \right\}$

AgC : nat pos<sub>X</sub> × nat pos<sub>Y</sub> × tablero t → tablero  
 $\left\{ \begin{array}{l} pos_X \upharpoonright \pi_1(\text{tamaño}(t)) \wedge pos_Y < \pi_2(\text{tamaño}(t)) \wedge_{\text{L}} (\neg \text{esLlegada?}(pos_X, pos_Y, t) \wedge \neg \text{esSalida?}(pos_X, pos_Y, t) \wedge \\ \neg \text{esPared?}(pos_X, pos_Y, t) \wedge \neg \text{esFantasma?}(pos_X, pos_Y, t)) \end{array} \right\}$

eliminarChocolate :  $\text{nat } pos_X \times \text{nat } pos_Y \times \text{tablero } t \longrightarrow \text{tablero}$   
 $\{\text{pos}_X < \pi_1(\text{tamaño}(t)) \wedge pos_Y < \pi_2(\text{tamaño}(t)) \wedge \text{esChocolate?}(\text{pos}_X, pos_Y, t)\}$

#### otras operaciones

entrada :  $\text{tablero} \longrightarrow \text{tupla}(\text{nat}, \text{nat})$   
salida :  $\text{tablero} \longrightarrow \text{tupla}(\text{nat}, \text{nat})$   
posicionValida :  $\text{nat} \times \text{nat} \times \text{tablero} \longrightarrow \text{bool}$   
posicionesFantasmas :  $\text{tablero} \longrightarrow \text{conj}(\text{nat}, \text{nat})$

**axiomas**  $\forall pos_X, pos_Y, pos_{X2}, pos_{Y2}, en_X, en_Y, ll_X, ll_Y, fil, col: \text{int} \ , \forall t: \text{tablero}$

posiciónValida( $pos_X, pos_Y, \text{tablero}$ )  $\equiv (0 \leq pos_X < \pi_1(\text{tamaño}(t)) \wedge$   
 $0 \leq pos_Y < \pi_2(\text{tamaño}(t))$

tamaño( $\text{crearTablero}(fil, col, en_X, en_Y, ll_X, ll_Y)$ )  $\equiv \langle fi, col \rangle$

tamaño( $\text{AgF}(pos_X, pos_Y, t)$ )  $\equiv \text{tamaño}(t)$

tamaño( $\text{AgP}(pos_X, pos_Y, t)$ )  $\equiv \text{tamaño}(t)$

tamaño( $\text{AgC}(pos_X, pos_Y, t)$ )  $\equiv \text{tamaño}(t)$

tamaño( $\text{eliminarChocolate}(pos_X, pos_Y, t)$ )  $\equiv \text{tamaño}(t)$

esFantasma?( $pos_X, pos_Y, \text{crearTablero}(fil, col, en_X, en_Y, ll_X, ll_Y)$ )  $\equiv \text{false}$

esFantasma?( $pos_{X2}, pos_{Y2}, \text{AgF}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then true else}$   
 $\text{esFantasma?}(pos_{X2}, pos_{Y2}, t) \text{ fi}$

esFantasma?( $pos_{X2}, pos_{Y2}, \text{AgP}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then}$   
 $\text{false}$

**else**

$\text{esFantasma?}(pos_{X2}, pos_{Y2}, t)$

**fi**

esFantasma?( $pos_{X2}, pos_{Y2}, \text{AgC}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then}$   
 $\text{false}$

**else**

$\text{esFantasma?}(pos_{X2}, pos_{Y2}, t)$

**fi**

esFantasma?( $pos_{X2}, pos_{Y2}, \text{eliminarChocolate}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then}$   
 $\text{false}$

**else**

$\text{esFantasma?}(pos_{X2}, pos_{Y2}, t)$

**fi**

esPared?( $pos_X, pos_Y, \text{crearTablero}(fil, col, en_X, en_Y, ll_X, ll_Y)$ )  $\equiv \text{false}$

esPared?( $pos_{X2}, pos_{Y2}, \text{AgF}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then}$   
 $\text{false}$

**else**

$\text{esPared?}(pos_{X2}, pos_{Y2}, t)$

**fi**

esPared?( $pos_{X2}, pos_{Y2}, \text{AgP}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then}$   
 $\text{true}$

**else**

$\text{esPared?}(pos_{X2}, pos_{Y2}, t)$

**fi**

esPared?( $pos_{X2}, pos_{Y2}, \text{AgC}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then}$   
 $\text{false}$

**else**

$\text{esPared?}(pos_{X2}, pos_{Y2}, t)$

**fi**

esPared?( $pos_{X2}, pos_{Y2}, \text{eliminarChocolate}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then}$   
 $\text{false}$

**else**

$\text{esPared?}(pos_{X2}, pos_{Y2}, t)$

**fi**

esLlegada?( $pos_{X2}, pos_{Y2}, \text{crearTablero}(fil, col, en_X, en_Y, ll_X, ll_Y)$ )  $\equiv \text{if } pos_{X2} = x_2 \wedge pos_{Y2} = y_2 \text{ then true else false fi}$

esLlegada?( $pos_{X2}, pos_{Y2}, \text{AgF}(pos_X, pos_Y, t)$ )  $\equiv \text{if } pos_{X2} = pos_X \wedge pos_{Y2} = pos_Y \text{ then}$   
 $\text{false}$

**else**

$\text{esLlegada?}(pos_{X2}, pos_{Y2}, t)$

**fi**

```

esLlegada?(posX2, posY2, AgP(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esLlegada?(posX2, posY2, t)
fi
esLlegada?(posX2, posY2, AgC(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esLlegada?(posX2, posY2, t)
fi
esLlegada?(posX2, posY2, eliminarChocolate(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esLlegada?(posX2, posY2, t)
fi
esEntrada?(posX2, posY2, crearTablero(fil, col, enX, enY, llX, llY))  ≡  if posX2 = enX ∧ posY2 = enY then
true
else
false
fi
esEntrada?(posX2, posY2, AgF(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esEntrada?(posX2, posY2, t)
fi
esEntrada?(posX2, posY2, AgP(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esEntrada?(posX2, posY2, t)
fi
esEntrada?(posX2, posY2, AgC(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esEntrada?(posX2, posY2, t)
fi
esEntrada?(posX2, posY2, eliminarChocolate(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esEntrada?(posX2, posY2, t)
fi
esChocolate?(posX2, posY2, crearTablero(fil, col, enX, enY, llX, llY))  ≡  if posX2 = enX ∧ posY2 = enY then
true
else
false
fi
esChocolate?(posX2, posY2, AgF(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esChocolate?(posX2, posY2, t)
fi
esChocolate?(posX2, posY2, AgP(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
false
else
esChocolate?(posX2, posY2, t)
fi
esChocolate?(posX2, posY2, AgC(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
true
else
esChocolate?(posX2, posY2, t)
fi

```

```

esChocolate?(posX2, posY2, eliminarChocolate(posX, posY, t))  ≡  if posX2 = posX ∧ posY2 = posY then
    false
else
    esChocolate?(posX2, posY2, t)
fi

entrada(crearTablero(fil,col,enX, enY, llX, llY))  ≡  ⟨enX, enY⟩
entrada(AgF(posX, posY, t))  ≡  entrada(t)
entrada(AgP(posX, posY, t))  ≡  entrada(t)
entrada(AgC(posX, posY, t))  ≡  entrada(t)
entrada(eliminarChocolate(posX, posY, t))  ≡  entrada(t)

llegada(crearTablero(fil,col,enX, enY, llX, llY))  ≡  ⟨llX, llY⟩
llegada(AgF(posX, posY, t))  ≡  llegada(t)
llegada(AgP(posX, posY, t))  ≡  llegada(t)
llegada(AgC(posX, posY, t))  ≡  llegada(t)
llegada(eliminarChocolate(posX, posY, t))  ≡  llegada(t)

posicionesFantasmas(crearTablero(fil,col,enX, enY, llX, llY))  ≡  ∅
posicionesFantasmas(AgF(posX, posY, t))  ≡  Ag(⟨posX, posY⟩, posicionesFantasmas(t))
posicionesFantasmas(AgP(posX, posY, t))  ≡  posicionesFantasmas(t)
posicionesFantasmas(AgC(posX, posY, t))  ≡  posicionesFantasmas(t)
posicionesFantasmas(eliminarChocolate(posX, posY, t))  ≡  posicionesFantasmas(t)

```

**Fin TAD**

**TAD** nombreJugador es String

**Fin TAD**

**TAD** posicion es  $\langle nat, nat \rangle$

**Fin TAD**

**TAD** Pacman

géneros pacman

igualdad observacional

$$(\forall p1, p2 : \text{pacman}) \quad p1 =_{\text{obs}} p2 \iff \left( \begin{array}{l} \text{posActualX}(p1) =_{\text{obs}} \text{posActualX}(p2) \\ \wedge \\ \text{posActualY}(p1) =_{\text{obs}} \text{posActualY}(p2) \\ \wedge \\ \text{dameTablero}(p1) =_{\text{obs}} \\ \text{dameTablero}(p2) \\ \wedge \\ \text{cantidadDeMovimientos}(p1) =_{\text{obs}} \\ \text{cantidadDeMovimientos}(p2) \\ \wedge \\ \text{turnoInmunidad}(p1) =_{\text{obs}} \\ \text{turnoInmunidad}(p2) \\ \wedge \\ \text{nombreJugadorActual}(p1) =_{\text{obs}} \\ \text{nombreJugadorActual}(p2) \\ \wedge \\ \text{ranking}(p1) =_{\text{obs}} \\ \text{ranking}(p2) \end{array} \right)$$

**usa** Tablero, Bool, Int, Nat, posicion, Secuencia, ranking, nombreJugador

**exporta** generos, observadores, generadores, gano?, asustado?, esInmune?, totalDeMovimientos

**observadores básicos**

dameTablero : pacman  $\longrightarrow$  tablero  
cantidadDeMovimientos : pacman  $\longrightarrow$  nat  
turnoInmunidad : pacman  $\longrightarrow$  nat  
posActualX : pacman  $\longrightarrow$  nat  
posActualY : pacman  $\longrightarrow$  nat  
nombreJugadorActual : pacman  $\longrightarrow$  nombreJugador  
ranking : pacman  $\longrightarrow$  ranking

**generadores**

spawnear : tablero  $\times$  nombreJugador  $\longrightarrow$  pacman  
mover : nat  $pos_X \times$  nat  $pos_Y \times$  pacman  $p \longrightarrow$  pacman  $\{ \text{posicionValida}(\text{pos}_X, \text{pos}_Y, \text{dameTablero}(p)) \wedge$   
 $\text{movimientoValido}(\text{pos}_X, \text{pos}_Y, \text{posActualX}(p), \text{posActualY}(p), \text{dameTablero}(p))$   
 $\wedge L \neg (\text{asustado?}(p) \vee \text{gano?}(p)) \}$

**otras operaciones**

gano? : pacman  $\longrightarrow$  bool  
asustado? : pacman  $\longrightarrow$  bool  
movimientoValido : nat  $pos_X \times$  nat  $pos_Y \times$  nat  $\times$  nat  $\times$  tablero  $t \longrightarrow$  bool  $\{ \text{posicionValida}(\text{pos}_X, \text{pos}_Y, t) \}$

esInmune? : pacman  $\rightarrow$  bool  
 totalDeMovimientos : pacman  $\rightarrow$  nat  
 esVecino? : conj(posicion)  $\times$  pacman  $\rightarrow$  bool  
 distanciaManhattan : posicion  $\times$  pacman  $\rightarrow$  nat  
 tieneFantasmasVecinos : pacman  $\rightarrow$  bool  
 averiguarMejor : pacman  $\rightarrow$  nombreJugador  
 averiguarArribaRanking : pacman  $\rightarrow$  nombreJugador  
 puntosParaSuperar : pacman  $\rightarrow$  nat  
 puntajeJugadorActual : pacman  $\rightarrow$  nat  
 averiguarPuntaje : pacman p  $\times$  nombreJugador n  $\rightarrow$  nat

$\{(\exists ju : jugador)(ju \in conjJugadores(ranking(p) \wedge \pi_1(ju) = n))\}$

**axiomas**  $\forall pacman: pacman$  ,  $\forall nat: x,y,posXPacman,posYPacman$  ,  $\forall tablero: tablero$

dameTablero(spawnear(tablero))	$\equiv$ tablero
dameTablero(mover(x,y,pacman))	$\equiv$ <b>if</b> esChocolate(x,y,dameTablero(pacman)) <b>then</b> eliminarChocolate(x,y,dameTablero(pacman)) <b>else</b> dameTablero(pacman) <b>fi</b>
cantidadDeMovimientos(spawnear(tablero))	$\equiv$ 0
cantidadDeMovimientos(mover(x,y,pacman))	$\equiv$ cantidadDeMovimientos(pacman) + 1
posActualX(spawnear(tablero))	$\equiv$ $\pi_1$ (entrada(tablero))
posActualX(mover(x,y,pacman))	$\equiv$ x
posActualY(spawnear(tablero))	$\equiv$ $\pi_2$ (entrada(tablero))
posActualY(mover(x,y,pacman))	$\equiv$ y
turnoInmunidad(spawnear(pacman))	$\equiv$ 0
turnoInmunidad(mover(x,y,pacman))	$\equiv$ <b>if</b> esChocolate?(x,y,dameTablero(pacman)) <b>then</b> 10 <b>else</b> max(0, turnoInmunidad(pacman)-1) <b>fi</b>
nombreJugadorActual(spawnear(tablero, nombreJugador, ranking))	$\equiv$ nombreJugador
nombreJugadorActual(mover(x, y, pacman))	$\equiv$ nombreJugadorActual(pacman)
ranking(spawnear(tablero,nombreJugador,r))	$\equiv$ r
ranking(mover(x,y,p))	$\equiv$ <b>if</b> $\langle x, y \rangle =$ llegada(dameTablero(p)) <b>then</b> AgregarJugador( $\langle$ nombreJugadorActual(p), cantidadMovimientos(p) + 1), ranking(p)) <b>else</b> ranking(p) <b>fi</b>
gano?(pacman)	$\equiv$ $\pi_1$ (llegada(dameTablero(pacman))) = posActualX(pacman) $\wedge$ $\pi_2$ (llegada(dameTablero(pacman))) = posActualY(pacman)
asustado?(pacman)	$\equiv$ $\neg esInmune(pacman) \wedge$ tieneFantasmasVecinos(pacman)
movimientoValido(x,y,posXPacman,posYPacman,tablero)	$\equiv$ (posXPacman - 1 $\leq$ x $\leq$ posXPacman + 1 $\wedge$ (posYPacman - 1 $\leq$ y $\leq$ posYPacman + 1) $\wedge$ $\neg(x = posXPacman + 1 \wedge y = posYPacman + 1)$ $\wedge$ $\neg(x = posXPacman + 1 \wedge y = posYPacman - 1)$ $\wedge$ $\neg(x = posXPacman - 1 \wedge$ y = posYPacman + 1) $\wedge$ $\neg(x = posXPacman - 1 \wedge$ y = posYPacman - 1) $\wedge$ $\neg esPared(x, y, tablero)$ )

esInmune?(pacman)	$\equiv$ turnoInmunidad(pacman) > 0
totalDeMovimientos(spawnear(tablero))	$\equiv$ 0
totalDeMovimientos(mover(x,y,pacman))	$\equiv$ totalDeMovimientos(pacman) + 1
tieneFantasmasVecinos(pacman)	$\equiv$ esVecino?(posicionesDeFantasmas(dameTablero(pacman)),p)
esVecino?( $\emptyset$ , <i>pacman</i> )	$\equiv$ false
esVecino?(conjFant,pacman)	$\equiv$ <b>if</b> distanciaManhattan(dameUno(conjFant,pacman)) < 3 <b>then</b> true <b>else</b> esVecino?(sinUno(conjFant,pacman)) <b>fi</b>
distanciaManhattan(posFan,pacman)	$\equiv$  posActualX(pacman)- $\pi_1$ (posFan)  +  posActualY(pacman)- $\pi_2$ (posFan)
averiguarMejor(pacman)	$\equiv$ $\pi_1$ (mejorPuntaje(ranking(pacman)))
averiguarArribaRanking(pacman)	$\equiv$ $\pi_1$ (JugadorArribaDe(nombreJugadorActual(pacman), ranking(pacman)))
puntosParaSuperar(pacman)	$\equiv$ $\pi_2$ (JugadorArribaDe(nombreJugadorActual(pacman), ranking(pacman)))
averiguarPuntaje(pacman,nombreJugador)	$\equiv$ puntajeActual(nombreJugador,ranking(pacman))
puntajeJugadorActual(pacman)	$\equiv$ puntajeActual(nombreJugadorActual(pacman),ranking(pacman))

**Fin TAD**



**TAD** Jugador Es tupla(string, nat)

**Fin TAD**

**TAD** Ranking

**géneros** ranking

**igualdad observacional**

$$(\forall r1, r2 : \text{ranking}) \ (r1 =_{\text{obs}} r2 \iff (\text{conjJugadores}(r1) =_{\text{obs}} \text{conjJugadores}(r2)))$$

**usa** Jugador, Nat, Bool, conj( $\alpha$ ), *nombreJugador*

**exporta** Observadores, Generadores, mejorPuntaje, puntajeActual, jugadorArribaDe

**observadores básicos**

$$\text{conjJugadores} : \text{ranking} \longrightarrow \text{conj}(\text{jugador})$$

**generadores**

$$\text{armarRanking} : \longrightarrow \text{ranking}$$

$$\text{agregarJugador} : \text{jugador} \times \text{ranking} \longrightarrow \text{ranking}$$

**otras operaciones**

$$\text{mejorPuntaje} : \text{ranking} \text{ r} \longrightarrow \text{jugador} \qquad \{\neg \text{vacio?}(\text{conjJugadores}(\text{r}))\}$$

$$\text{estaEnElRanking?} : \text{nombreJugador} \times \text{ranking} \longrightarrow \text{bool}$$

$$\text{puntajeActual} : \text{nombreJugador n} \times \text{ranking r} \longrightarrow \text{bool} \qquad \{(\exists ju : \text{jugador})(ju \in \text{conjJugadores}(\text{r}) \wedge \pi_1(ju) = \text{n})\}$$

$$\text{jugadorArribaDe} : \text{nombreJugador n} \times \text{ranking r} \longrightarrow \text{jugador} \\ \{(\exists j : \text{jugador})(j \in \text{conjJugadores}(\text{r}) \wedge \pi_1(j) = \text{n} \wedge \text{puntajeActual}(j) < \text{mejorPuntaje}(\text{r}))\}$$

$$\text{jugadorConXPuntos} : \text{nat} \times \text{conj}(\text{jugador}) \longrightarrow \text{conj}(\text{jugador})$$

$$\text{inmediatamenteSuperior} : \text{jugador} \times \text{conj}(\text{jugador}) \longrightarrow \text{nat}$$

$$\text{jugador} : \text{nombreJugador n} \times \text{conj}(\text{jugador}) \text{ conjJug} \longrightarrow \text{jugador} \qquad \{(\exists j : \text{jugador}) j \in \text{conjJug} \wedge \pi_1(j) = \text{n}\}$$

**axiomas**  $\forall \text{ranking}: \text{ranking} \ , \forall \text{jugador}: \text{jugador} \langle \text{string}, \text{nat} \rangle \ , \forall \text{puntaje}: \text{nat} \ , \forall \text{nombre}: \text{nombreJugador} \ ,$   
 $\forall \text{conjjug}: \text{conj}(\text{jugador})$

$$\text{conjJugadores}(\text{armarRanking}) \equiv \emptyset$$

$$\begin{aligned} \text{conjJugadores}(\text{agregarJugador}(\text{jugador}, \text{ranking})) \equiv & \text{if } \text{estaEnElRanking?}(\pi_1(\text{jugador}), \text{ranking}) \\ & \text{then if } \text{puntajeActual}(\pi_1(\text{jugador}), \text{conjJugadores}(\text{ranking})) < \pi_2(\text{j}) \\ & \text{then} \\ & \text{conjJugadores}(\text{ranking}) \\ & \text{else} \\ & \text{Ag}(\text{jugador}, \text{conjJugadores}(\text{ranking})) - \\ & \langle \pi_1(\text{jugador}), \text{puntajeActual}(\pi_1(\text{jugador}), \text{conjJugadores}(\text{ranking})) \rangle \\ & \text{fi} \\ & \text{else} \\ & \text{Ag}(\text{jugador}, \text{conjJugadores}(\text{ranking})) \\ & \text{fi} \end{aligned}$$

$$\text{estaEnElRanking?}(\text{nombre}, \text{armarRanking}) \equiv \text{false}$$

$$\begin{aligned} \text{estaEnElRanking?}(\text{nombre}, \text{agregarJugador}(\text{jugador}, \text{ranking})) \equiv & \text{if } \pi_1(\text{jugador}) = \text{nombre} \text{ then} \\ & \text{true} \\ & \text{else} \\ & \text{estaEnELRanking?}(\text{nombre}, \text{ranking}) \\ & \text{fi} \end{aligned}$$

```

puntajeActual(nombre,conjug)  $\equiv$  if  $\pi_1(\text{dameUno}(\text{conjug})) = \text{nombre}$  then
     $\pi_2(\text{dameUno}(\text{conjug}))$ 
else
    puntajeActual(nombre,sinUno(conjug))
fi

jugadorArribaDe(nombre,ranking)  $\equiv$  dameUno(
    jugadoresConXPuntos(
        inmediatamenteSuperior(
            jugador(nombre,conjJugadores(ranking)),conjJugadores(ranking)-
            jugador(nombre,conjJugadores(ranking))),
        conjJugadores(ranking)))

inmediatamenteSuperior(jugador,conjug)  $\equiv$  if  $\# \text{conjug} = 1$  then
     $\pi_2(\text{jugador})$ 
else
     $\min(\pi_2(\text{dameUno}(\text{conjug})), \text{inmediatamenteSuperior}(\text{jugador}, \text{sinUno}(\text{conjug})))$ 
fi

jugadoresConXPuntos(puntaje,conjug)  $\equiv$  if  $\text{vacio?}(\text{conjug})$  then  $\emptyset$  else if  $\pi_2(\text{dameUno}(\text{conjug})) = x$  then
    Ag(dameUno(conjug),jugadoresConXPuntos(puntaje,sinUno(conjug)))
else
    jugadoresConXPuntos(puntaje,sinUno(conjug))
fi

jugador(nombre,conjug)  $\equiv$  if  $\pi_1(\text{dameUno}(\text{conjug})) = \text{nomJug}$  then
    dameUno(conjug)
else
    jugador(nombre,sinUno(conjug))
fi

```

**Fin TAD**