



**DEPARTAMENTO
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

TP 1.2

Algoritmos y Estructuras de Datos II

Grupo: 18

Integrante	LU	Correo electrónico
Castro Russo, Matias Nahuel	203/19	castronahuel14@gmail.com
Torsello, Juan Manuel	248/19	juantorsello@gmail.com
Capelo, Gianluca	83/19	gianluca.capelo@gmail.com
Yazlle, Máximo	310/19	myazlle99@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. TAD Grilla

TAD Grilla

TAD Eje es Bool

TAD Posición Es Tupla(nat,nat)

usa POSICIÓN, EJE,NAT

igualdad observacional

$(\forall g, g' : \text{Grilla}) (g =_{\text{obs}} g' \iff (\forall p : \text{Posicion})(\text{HayRío?}(g,p) =_{\text{obs}} \text{HayRío?}(g',p)))$

género: grilla

observadores básicos

HayRío? : grilla \times Posición \longrightarrow bool

generadores

CrearGrilla : \longrightarrow Grilla

UnirGrillas : grilla \times grilla \longrightarrow Grilla

En el parámetro Eje el cual es un renombre para el TAD Bool , tomamos que si es true el río va a ser agregado en sentido horizontal y en cambio si fuese false vertical.

AgRío : grilla \times nat \times Eje \longrightarrow Grilla

axiomas

HayRío?(CrearGrilla,pos) \equiv False

HayRío?(AgRio(g,coordenada,eje),pos) \equiv (eje \wedge (π_1 (pos) = coordenada))
 \vee (\neg eje \wedge (π_2 (pos) = coordenada))
 \vee (HayRío?(g,pos))

HayRío?(UnirGrillas(g_1, g_2),pos) \equiv HayRío?(g_1 ,pos) \vee HayRío?(g_2 ,pos)

Fin TAD

2. TAD Partida

TAD Partida

Posición Es Tupla(nat,nat)

usa POSICIÓN,GRILLA,BOOL,NAT

género: partida

igualdad observacional

$(\forall p, p' : \text{partida}) \left(p =_{\text{obs}} p' \iff \left(\begin{array}{l} \text{VerGrilla}(p) =_{\text{obs}} \text{VerGrilla}(p') \wedge \\ \text{CantDeUniones}(p) =_{\text{obs}} \text{CantDeUniones}(p') \wedge \\ \text{SeConstruyoEsteTurno?}(p) =_{\text{obs}} \\ \text{SeConstruyoEsteTurno}(p') \wedge \\ (\forall \text{pos:Posicion}) (\text{hayCasa?}(p,\text{pos}) \\ =_{\text{obs}} \text{hayCasa?}(p',\text{pos})) \wedge \\ \text{hayComercio?}(p,\text{pos}) =_{\text{obs}} \text{hayComercio?}(p',\text{pos}) \\) \wedge_L \\ (\forall \text{pos: Posicion}) (\text{hayComercio?}(p,\text{pos}) \vee \text{hayCa-} \\ \text{sa?}(p,\text{pos}) \Rightarrow_L \text{Nivel}(p',\text{pos}) =_{\text{obs}} \text{Nivel}(p,\text{pos})) \end{array} \right) \right)$

observadores básicos

HayCasa?	: partida \times Posición	\longrightarrow Bool	
HayComercio?	: partida \times Posición	\longrightarrow Bool	
Nivel	: partida $p \times$ Posición pos	\longrightarrow nat	$\{ \text{HayCasa?}(pos,p) \vee \text{HayComercio?}(pos,p) \}$
VerGrilla	: partida	\longrightarrow grilla	
SeConstruyoEsteTurno?	: partida	\longrightarrow Bool	
CantDeUniones	: partida	\longrightarrow Nat	

generadores

NuevaPartida	: grilla	\longrightarrow partida	
PasarTurno	: partida p	\longrightarrow partida	$\{ \text{SeConstruyoEsteTurno?}(p) \}$
CrearCasa	: partida $p \times$ Posición pos	\longrightarrow partida	$\{ \text{SePuedeConstruir?}(p,pos) \}$
CrearComercio	: partida $p \times$ Posición pos	\longrightarrow partida	$\{ \text{SePuedeConstruir?}(p,pos) \}$
UnirPartidas	: partida $a \times$ partida b	\longrightarrow partida	$\{ \text{AptoParaUnion} \}$

otras operaciones

SePuedeConstruir?	: partida \times Posición	\longrightarrow Bool
mayorNivelCasa	: partida \times conj(Posición) \times nat	\longrightarrow nat
PosicionesManhattan	: nat $dist \times$ Posición pos	\longrightarrow conj(Posición)
Alejar	: conj(Posición)	\longrightarrow conj(Posición)
AlejarUnaPosición	: Posición	\longrightarrow conj(Posición)
CantDeTurnos	: Partida	\longrightarrow Nat
HayConstrucción?	: partida \times Posición	\longrightarrow Bool
EstaADist3Manhattan?	: Posición \times Posición	\longrightarrow Bool

axiomas

EstaADist3Manhattan?(pos1,pos2)	$\equiv \pi_1(pos1) - \pi_1(pos2) + \pi_2(pos1) - \pi_2(pos2) = 3$
mayorNivelCasa(p,c,maxNivel,pos)	\equiv if $\emptyset?(c)$ then maxNivel else if EstaADist3Manhattan?(pos,DameUno(c)) \wedge hayCasa?(p,DameUno(c)) \wedge maxNivel < nivel(p,DameUno(c)) then mayorNivelCasa(p,SinUno(c),nivel(p,DameUno(c))) else mayorNivelCasa(p,SinUno(c),maxNivel) fi fi

”PosicionesManhattan devuelve todas las posiciones a ’dist’ casilleros de distancia manhattan de una posición.”

PosicionesManhattan(dist,pos)	\equiv if dist = 0 then {pos} else Alejar(PosicionesManhattan(dist - 1,pos)) \cup PosicionesManhattan(dist -1,pos) fi
Alejar(c)	\equiv if $\emptyset?(c)$ then \emptyset else AlejarseUnaPosición(DameUno(c)) \cup Alejar(SinUno(c)) fi

$$\begin{aligned} \text{AlejarUnaPosición}(\text{pos}) \equiv & \{(\pi_1(\text{pos})+1, \pi_2(\text{pos})), (\pi_1(\text{pos}), \pi_2(\text{pos})+1)\} \cup \\ & \text{if } 0 < \pi_1(\text{pos}) \wedge 0 < \pi_2(\text{pos}) \text{ then} \\ & \quad \{(\pi_1(\text{pos})-1, \pi_2(\text{pos})), (\pi_1(\text{pos}), \pi_2(\text{pos})-1)\} \\ & \text{else} \\ & \quad \text{if } 0 < \pi_1(\text{pos}) \text{ then} \\ & \quad \quad \{(\pi_1(\text{pos})-1, \pi_2(\text{pos}))\} \\ & \quad \text{else} \\ & \quad \quad \text{if } 0 < \pi_2(\text{pos}) \text{ then} \\ & \quad \quad \quad \{(\pi_1(\text{pos}), \pi_2(\text{pos})-1)\} \text{ else } \emptyset \text{ fi} \\ & \text{fi} \end{aligned}$$

$$\begin{aligned} \text{SePuedeConstruir?}(p, \text{pos}) \equiv & \neg \text{HayCasa?}(p, \text{pos}) \wedge \\ & \neg \text{HayComercio?}(p, \text{pos}) \wedge \\ & \neg \text{HayRío?}(\text{VerGrilla}(p), \text{pos}) \end{aligned}$$

$$\begin{aligned} \text{HayConstrucción?}(p, \text{pos}) \equiv \\ \text{HayCasa}(p, \text{pos}) \vee \text{HayComercio}(p, \text{pos}) \end{aligned}$$

Axiomatizamos CantidadTurnos sobre los generadores teniendo en cuenta que se deduce de los observadores. La cantidad de turnos es igual al nivel máximo que haya en la partida , ya que es obligatorio construir al menos una vez en cada turno.

$$\begin{aligned} \text{CantDeTurnos}(\text{NuevaPartida}(g)) & \equiv 0 \\ \text{CantDeTurnos}(\text{PasarTurno}(p)) & \equiv \text{CantDeTurnos}(p) + 1 \\ \text{CantDeTurnos}(\text{CrearCasa}(p, \text{pos})) & \equiv \text{CantDeTurnos}(p) \\ \text{CantDeTurnos}(\text{CrearComercio}(p, \text{pos})) & \equiv \text{CantDeTurnos}(p) \\ \text{CantDeTurnos}(\text{UnirPartidas}(p_a, p_b)) & \equiv \max(\text{CantDeTurnos}(p_a), \text{CantDeTurnos}(p_b)) \end{aligned}$$

$$\begin{aligned} \text{HayCasa?}(\text{NuevaPartida}(g), \text{pos}) & \equiv \text{false} \\ \text{HayCasa?}(\text{PasarTurno}(p), \text{pos}) & \equiv \text{HayCasa?}(p, \text{pos}) \\ \text{HayCasa?}(\text{CrearCasa}(p, \text{pos}'), \text{pos}) & \equiv (\text{pos}' = \text{pos}) \vee \text{HayCasa?}(p, \text{pos}) \\ \text{HayCasa?}(\text{CrearComercio}(p, \text{pos}'), \text{pos}) & \equiv \text{HayCasa?}(p, \text{pos}) \\ \text{HayCasa?}(\text{UnirPartidas}(p_1, p_2), \text{pos}) & \equiv \text{HayCasa?}(p_1, \text{pos}) \vee \text{HayCasa?}(p_2, \text{pos}) \end{aligned}$$

$$\begin{aligned} \text{HayComercio?}(\text{NuevaPartida}(g), \text{pos}) & \equiv \text{false} \\ \text{HayComercio?}(\text{PasarTurno}(p), \text{pos}) & \equiv \text{HayComercio?}(p, \text{pos}) \\ \text{HayComercio?}(\text{CrearCasa}(p, \text{pos}'), \text{pos}) & \equiv \text{HayComercio?}(p, \text{pos}) \\ \text{HayComercio?}(\text{CrearComercio}(p, \text{pos}'), \text{pos}) & \equiv (\text{pos}' = \text{pos}) \vee \text{HayComercio?}(p, \text{pos}) \\ \text{HayComercio?}(\text{UnirPartidas}(p_1, p_2), \text{pos}) & \equiv \neg(\text{HayCasa?}(p_1, \text{pos}) \vee \text{HayCasa?}(p_2, \text{pos})) \wedge \\ & \quad (\text{HayComercio?}(p_1, \text{pos}) \vee \text{HayComercio?}(p_2, \text{pos})) \end{aligned}$$

$$\begin{aligned} \text{Nivel}(\text{PasarTurno}(p), \text{pos}) & \equiv \text{nivel}(p, \text{pos}) + 1 \\ \text{Nivel}(\text{CrearCasa}(p, \text{pos}'), \text{pos}) & \equiv \text{if } \text{pos} = \text{pos}' \text{ then } 0 \text{ else } \text{Nivel}(p, \text{pos}) \text{ fi} \\ \text{Nivel}(\text{CrearComercio}(p, \text{pos}'), \text{pos}) & \equiv \text{if } \text{pos} = \text{pos}' \text{ then} \\ & \quad \text{mayorNivelCasa}(p, \text{PosicionesManhattan}(3, \text{pos}), 0, \text{pos}) \\ & \text{else} \\ & \quad \text{Nivel}(p, \text{pos}) \\ & \text{fi} \end{aligned}$$

$$\begin{aligned} \text{Nivel}(\text{UnirPartidas}(p_1, p_2), \text{pos}) &\equiv \text{if } ((\text{HayConstrucción?}(p_1, \text{pos}) \wedge \neg \text{HayConstrucción?}(p_2, \text{pos})) \vee \\ &\quad (\text{HayCasa?}(p_1, \text{pos}) \wedge \text{HayComercio?}(p_2, \text{pos})) \text{ then} \\ &\quad \quad \text{Nivel}(p_1, \text{pos}) \\ &\text{else} \\ &\quad \text{if } (\neg \text{HayConstrucción?}(p_1, \text{pos}) \wedge \text{HayConstrucción?}(p_2, \text{pos})) \\ &\quad \vee (\text{HayComercio?}(p_1, \text{pos}) \wedge \text{HayCasa?}(p_2, \text{pos})) \text{ then} \\ &\quad \quad \text{Nivel}(p_2, \text{pos}) \\ &\quad \text{else} \\ &\quad \quad \text{max}(\text{Nivel}(p_1, \text{pos}), \text{Nivel}(p_2, \text{pos})) \\ &\quad \text{fi} \\ &\text{fi} \end{aligned}$$

$$\begin{aligned} \text{VerGrilla}(\text{NuevaPartida}(g)) &\equiv g \\ \text{VerGrilla}(\text{PasarTurno}(p)) &\equiv \text{VerGrilla}(p) \\ \text{VerGrilla}(\text{CrearCasa}(p, \text{pos})) &\equiv \text{VerGrilla}(p) \\ \text{VerGrilla}(\text{CrearComercio}(p, \text{pos})) &\equiv \text{VerGrilla}(p) \\ \text{VerGrilla}(\text{UnirPartidas}(p_1, p_2)) &\equiv \text{UnirGrillas}(\text{VerGrilla}(p_1), \text{VerGrilla}(p_2)) \end{aligned}$$

$$\begin{aligned} \text{SeConstruyoEsteTurno?}(\text{NuevaPartida}(g)) &\equiv \text{false} \\ \text{SeConstruyoEsteTurno?}(\text{PasarTurno}(p)) &\equiv \text{false} \\ \text{SeConstruyoEsteTurno?}(\text{CrearCasa}(p, \text{pos})) &\equiv \text{true} \\ \text{SeConstruyoEsteTurno?}(\text{CrearComercio}(p, \text{pos})) &\equiv \text{true} \\ \text{SeConstruyoEsteTurno?}(\text{UnirPartidas}(p_1, p_2)) &\equiv \text{SeConstruyoEsteTurno?}(p_1) \vee \text{SeConstruyoEsteTurno?}(p_2) \end{aligned}$$

$$\begin{aligned} \text{CantDeUniones}(\text{NuevaPartida}(g)) &\equiv 0 \\ \text{CantDeUniones}(\text{PasarTurno}(p)) &\equiv \text{CantDeUniones}(p) \\ \text{CantDeUniones}(\text{CrearCasa}(p, \text{pos})) &\equiv \text{CantDeUniones}(p) \\ \text{CantDeUniones}(\text{CrearComercio}(p, \text{pos})) &\equiv \text{CantDeUniones}(p) \\ \text{CantDeUniones}(\text{UnirPartidas}(p_1, p_2)) &\equiv \text{CantDeUniones}(p_1) + \text{CantidadDeUniones}(p_2) + 1 \end{aligned}$$

Reemplazos sintácticos:

AptoParaUnión :

$$\begin{aligned} (\forall \text{pos:Posicion}) &(\text{HayRío?}(p_1, \text{pos}) \Rightarrow \neg \text{HayConstrucción?}(p_2, \text{pos}) \wedge \\ &\quad \text{HayRío?}(p_2, \text{pos}) \Rightarrow \neg \text{HayConstrucción?}(p_1, \text{pos}) \wedge \\ &\quad \text{HayConstrucción?}(p_1, \text{pos}) \wedge \text{Nivel}(p_1, \text{pos}) = \text{CantTurnos}(p_1) \Rightarrow \neg \text{HayConstrucción?}(p_2, \text{pos}) \wedge \\ &\quad \text{HayConstrucción?}(p_2, \text{pos}) \wedge \text{Nivel}(p_2, \text{pos}) = \text{CantTurnos}(p_2) \Rightarrow \neg \text{HayConstrucción?}(p_1, \text{pos})) \end{aligned}$$

Decisiones tomadas:

En caso de haber una casa y un comercio en la misma posición, decidimos que cuando se unan dos partidas, se conserve la casa, ya que se puede construir un comercio con un nivel inicial diferente de 0. Si hay dos contrucciones del mismo tipo prevalecerá la de mayor nivel.

Fin TAD