



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

TP 1.1

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 Posición Es Tupla(nat,nat)

usa POSICIÓN, BOOL,NAT

igualdad observacional

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

género: grilla

observadores básicos

HayRio? : grilla \times Posición \longrightarrow bool

generadores

CrearGrilla : \longrightarrow Grilla

AgRio : grilla \times nat \times bool \longrightarrow Grilla

axiomas

HayRio?(CrearGrilla,pos) \equiv False

HayRio?(AgRio(g,coordenada,eje),pos) \equiv (eje \wedge (pos₁ = coordenada))
 \vee (\neg eje \wedge (pos₂ = coordenada))
 \vee (hayRio?(g,pos))

Fin TAD

2. TAD Partida

TAD Partida

Posición Es Tupla(nat,nat)

usa POSICIÓN,GRILLA,BOOL,NAT

género: partida

igualdad observacional

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

observadores básicos

HayCasa? : partida \times Posición \longrightarrow Bool

HayComercio? : partida \times Posición \longrightarrow Bool

Nivel : partida $p \times$ Posición $pos \longrightarrow$ nat

$\{\text{HayCasa?}(\text{pos}, p) \vee \text{HayComercio?}(\text{pos}, p)\}$

VerGrilla : partida \longrightarrow grilla

SeConstruyoEsteTurno? : partida \longrightarrow Bool

generadores

NuevaPartida	: grilla	→ partida	
PasarTurno	: partida p	→ partida	{SeConstruyoEsteTurno?(p)}
CrearCasa	: partida $p \times$ Posición pos	→ partida	{SePuedeConstruir?(p,pos)}
CrearComercio	: partida $p \times$ Posición pos	→ partida	{SePuedeConstruir?(p,pos)}

otras operaciones

SePuedeConstruir?	: partida \times Posición	→ Bool	
mayorNivelCasa	: partida \times conj(Posición) \times nat	→ nat	
PosicionesManhattan	: nat $dist \times$ Posición pos	→ conj(Posición)	{0 <= dist}
Alejar	: conj(Posición)	→ conj(Posición)	
AlejarUnaPosición	: Posición	→ conj(Posición)	

axiomas

$mayorNivelCasa(p,c,maxNivel) \equiv$ **if** $\emptyset?(c)$ **then** maxNivel **else**
 if $(0 \leq DameUno(c)_1 \wedge 0 \leq DameUno(c)_2) \wedge_L$
 $hayCasa?(p,DameUno(c)) \wedge_L maxNivel < nivel(p,DameUno(c))$
 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
 $Alejar(Posiciones(dist - 1,pos)) \cup PosicionesManhattan(dist - 1,pos)$
 fi

$Alejar(c) \equiv$ **if** $\emptyset?(c)$ **then**
 \emptyset
 else
 $AlejarseUnaPosición(DameUno(c)) \cup Alejar(SinUno(c))$
 fi

$AlejarUnaPosición(pos) \equiv \{ (pos_1 + 1, pos_2), (pos_1, pos_2 + 1), (pos_1 - 1, pos_2), (pos_1, pos_2 - 1) \}$

$SePuedeConstruir?(p,pos) \equiv$ $\neg HayCasa?(p,pos) \wedge$
 $\neg HayComercio?(p,pos) \wedge$
 $\neg HayRio?(VerGrilla(p),pos)$

$HayCasa?(NuevaPartida(g),pos) \equiv false$
 $HayCasa?(PasarTurno(p),pos) \equiv HayCasa?(p,pos)$
 $HayCasa?(CrearCasa(p,pos'),pos) \equiv$ **if** $pos = pos'$ **then** true **else** $HayCasa?(p,pos)$ **fi**
 $HayCasa?(CrearComercio(p,pos'),pos) \equiv HayCasa?(p,pos)$

$HayComercio?(NuevaPartida(g),pos) \equiv false$
 $HayComercio?(PasarTurno(p),pos) \equiv HayComercio?(p,pos)$
 $HayComercio?(CrearCasa(p,pos'),pos) \equiv HayComercio?(p,pos)$

```

HayComercio?(CrearComercio(p,pos'),pos)  ≡  if pos' = pos then
                                             true
                                             else
                                             HayComercio?(p,pos)
                                             fi
Nivel(PasarTurno(p),pos)                  ≡  nivel(p,pos) + 1
Nivel(CrearCasa(p,pos'),pos)              ≡  if pos = pos' then 0 else Nivel(p,pos) fi
Nivel(CrearComercio(p,pos'),pos)          ≡  if pos = pos' then
                                             mayorNivelCasa(PosicionesManhattan(3,pos),0)
                                             else
                                             Nivel(p,pos)
                                             fi
VerGrilla(NuevaPartida(g))                ≡  g
VerGrilla(PasarTurno(p))                  ≡  grilla (p)
VerGrilla(CrearCasa(p,pos))               ≡  grilla(p)
VerGrilla(CrearComercio(p,pos))           ≡  grilla(p)

SeConstruyoEsteTurno?(NuevaPartida(g))    ≡  false
SeConstruyoEsteTurno?(PasarTurno(p))      ≡  false
SeConstruyoEsteTurno?(CrearCasa(p,pos))    ≡  true
SeConstruyoEsteTurno?(CrearComercio(p,pos)) ≡  true

```

Fin TAD