



**DEPARTAMENTO
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

Simcity

Trabajo practico grupal N

31 de mayo de 2020

Algoritmos y Estructuras de Datos II

Grupo 17

Integrante	LU	Correo electrónico
Rodriguez Celma, Guido	374/19	guido.rodriguez@outlook.com.ar
Itzcovitz, Ryan	169/19	ryanitzcovitz@gmail.com
Rodriguez, Miguel	57/19	mmiguerodriguez@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

<https://exactas.uba.ar>

TAD RIO es TUPLA<BOOL, NAT>

TAD POS es TUPLA<NAT, NAT>

TAD SimCity

géneros simCity

igualdad observacional

$$(\forall s, s' : \text{simCity}) \left(s =_{\text{obs}} s' \iff \left(\begin{array}{l} \text{antigüedad}(s) = \text{antigüedad}(s') \wedge \\ \text{popularidad}(s) = \text{popularidad}(s') \wedge \\ \text{rios}(s) =_{\text{obs}} \text{rios}(s') \wedge \text{construcciones}(s) \\ =_{\text{obs}} \text{construcciones}(s') \wedge_{\text{L}} (\forall c: \\ \text{construccion})(c \in \text{construcciones}(s) \\ \Rightarrow_{\text{L}} \text{nivelDeConstruccion}(s, c) = \\ \text{nivelDeConstruccion}(s', c)) \end{array} \right) \right)$$

observadores básicos

rios	: simCity	→ conj(rio)	
construcciones	: simCity	→ conj(constr)	
nivelDeConstruccion	: simCity $s \times \text{constr } c$	→ nat	$\{c \in \text{construcciones}(s)\}$
popularidad	: simCity	→ nat	
antigüedad	: simCity	→ nat	

generadores

generar	: conj(rio)	→ simCity	
nuevoTurno	: simCity s	→ simCity	$\{\# \text{construidasUltimoTurno}(s) > 0\}$
construir	: simCity $s \times \text{constr } c$	→ simCity	$\{\neg \text{hayConstruccion}(s, \text{posicion}(c)) \wedge \neg \text{hayRio}(s, \text{posicion}(c))\}$
unir	: simCity $s \times \text{simCity } s'$	→ simCity	$\{\neg \text{hayRiosConflictivos}(s, s')\}$

otras operaciones

hayRio	: simCity $\times \text{pos}$	→ bool	
hayConstruccion	: simCity $\times \text{pos}$	→ bool	
nivelMaxCercanos	: simCity $s \times \text{constr } c$	→ nat	$\{\text{tipo}(c) = \text{"comercio"}\}$
$\# \text{construidasUltimoTurno}$: simCity	→ nat	
construccionesUltimoTurno	: simCity	→ conj(constr)	
hayRiosConflictivos	: simCity $\times \text{simCity}$	→ bool	

axiomas

rios(generar(r))	$\equiv r$
rios(nuevoTurno(s))	$\equiv \text{rios}(s)$
rios(construir(s, c))	$\equiv \text{rios}(s)$
rios(unir(s, s'))	$\equiv \text{rios}(s) \cup \text{rios}(s')$
construccion(generar(r))	$\equiv \emptyset$
construcciones(nuevoTurno(s))	$\equiv \text{construcciones}(s)$
construcciones(construir(s, c))	$\equiv \text{Ag}(c, \text{construcciones}(s))$

$\text{construcciones}(\text{unir}(s, s'))$	$\equiv \text{unirConstrucciones}(s, s', \text{construcciones}(s), \text{construcciones}(s'))$
$\text{hayRio}(s, p)$	$\equiv \text{hayRioAux}(\text{rios}(s), p)$
$\text{hayConstruccion}(s, p)$	$\equiv \text{hayConstruccionAux}(\text{construcciones}(s), p)$
$\# \text{construidasUltimoTurno}(\text{generar}(r))$	$\equiv 0$
$\# \text{construidasUltimoTurno}(\text{nuevoTurno}(s))$	$\equiv 0$
$\# \text{construidasUltimoTurno}(\text{construir}(s, c))$	$\equiv 1 + \# \text{construidasUltimoTurno}(s)$
$\# \text{construidasUltimoTurno}(\text{unir}(s, s'))$	$\equiv \#(\text{construccionesUltimoTurno}(\text{unir}(s, s')))$
$\text{construccionesUltimoTurno}(\text{generar}(r))$	$\equiv \emptyset$
$\text{construccionesUltimoTurno}(\text{nuevoTurno}(s))$	$\equiv \emptyset$
$\text{construccionesUltimoTurno}(\text{construir}(s, c))$	$\equiv \text{Ag}(c, \text{construccionesUltimoTurno}(s))$
$\text{construccionesUltimoTurno}(\text{unir}(s, s'))$	$\equiv \text{unirConstrucciones}(s, \text{construccionesUltimoTurno}(s), \text{construccionesUltimoTurno}(s')), s',$
$\text{nivelDeConstruccion}(\text{nuevoTurno}(s), c)$	$\equiv 1 + \text{nivelDeConstruccion}(s, c)$
$\text{nivelDeConstruccion}(\text{construir}(s, c'), c)$	$\equiv \text{if } c' = c \text{ then}$ $\quad \text{if tipo}(c) = \text{"casa"} \text{ then}$ $\quad \quad 0$ $\quad \text{else}$ $\quad \quad \text{nivelMaxCercanos}(s, c)$ $\quad \text{fi}$ else $\quad \text{nivelDeConstruccion}(s, c)$ fi
$\text{nivelDeConstruccion}(\text{unir}(s, s'), c)$	$\equiv \text{if tipo}(c) = \text{"casa"} \text{ then}$ $\quad \text{if } c \in \text{construcciones}(s) \text{ then}$ $\quad \quad \text{nivelDeConstruccion}(s, c)$ $\quad \text{else}$ $\quad \quad \text{nivelDeConstruccion}(s', c)$ $\quad \text{fi}$ else $\quad \text{if } c \in \text{construcciones}(s) \text{ then}$ $\quad \quad \max(\text{nivelMaxCercanos}(\text{unir}(s, s'), c),$ $\quad \quad \text{nivelDeConstruccion}(s, c))$ $\quad \text{else}$ $\quad \quad \max(\text{nivelMaxCercanos}(\text{unir}(s, s'), c),$ $\quad \quad \text{nivelDeConstruccion}(s', c))$ $\quad \text{fi}$
$\text{antigüedad}(\text{generar}(s))$	$\equiv 0$
$\text{antigüedad}(\text{nuevoTurno}(s))$	$\equiv \text{antigüedad}(s) + 1$
$\text{antigüedad}(\text{construir}(s, c))$	$\equiv \text{antigüedad}(s)$
$\text{antigüedad}(\text{unir}(s, s'))$	$\equiv \max(\text{antigüedad}(s), \text{antigüedad}(s'))$
$\text{popularidad}(\text{generar}(sr))$	$\equiv 0$
$\text{popularidad}(\text{nuevoTurno}(s))$	$\equiv \text{popularidad}(s)$
$\text{popularidad}(\text{construir}(s, c))$	$\equiv \text{popularidad}(s)$
$\text{popularidad}(\text{unir}(s, s'))$	$\equiv \text{popularidad}(s) + \text{popularidad}(s') + 1$
$\text{hayRiosConflictivos}(s, s')$	$\equiv \text{hayRiosConflictivosAux}(\text{rios}(s), \text{rios}(s'), \text{construcciones}(s), \text{construcciones}(s'))$

nivelMaxCercanos(s, c) \equiv nivelMaxCercanosAux(s , construcciones(s), c)

Fin TAD

auxiliares

```

hayRioAux( $rs, p$ )  $\equiv$  if vacio?( $rs$ ) then
    false
else
    if [ $\pi_1(dameUno(rs)) = 0 \wedge \pi_2(dameUno(rs)) = \pi_2(p)$ ]  $\vee$ 
        [ $\pi_1(dameUno(rs)) = 1 \wedge \pi_2(dameUno(rs)) = \pi_1(p)$ ] then
        true
    else
        hayRioAux(sinUno( $rs$ ),  $p$ )
    fi
fi

hayConstruccionAux( $cs, p$ )  $\equiv$  if vacio?( $cs$ ) then
    false
else
    if  $\pi_1(dameUno(cs)) = \pi_1(p) \wedge \pi_1(dameUno(cs)) = \pi_2(p)$  then
        true
    else
        hayConstruccionAux(sinUno( $cs$ ),  $p$ )
    fi
fi

distancia( $c, c'$ )  $\equiv$   $\max(\pi_1(posicion(c)), \pi_1(posicion(c')))$  -  $\min(\pi_1(posicion(c)), \pi_1(posicion(c')))$ 
    +  $\max(\pi_2(posicion(c)), \pi_2(posicion(c')))$  -  $\min(\pi_2(posicion(c)), \pi_2(posicion(c')))$ 

nivelMaxCercanosAux( $s, cs, c$ )  $\equiv$  if vacio?( $cs$ ) then
    0
else
    if distancia(dameUno( $cs$ ),  $c$ )  $\leq 3 \wedge$  tipo(dameUno( $cs$ )) = "casa" then
        max(nivelDeConstruccion( $s$ , dameUno( $cs$ )), nivelMaxCercanosAux( $s$ , sinUno( $cs$ ),  $c$ ))
    else
        nivelMaxCercanosAux( $s$ , sinUno( $cs$ ),  $c$ )
    fi
fi

hayRiosConflictivosAux( $rs, rs', cs, cs'$ )  $\equiv$  if  $\neg$ vacio?( $cs$ ) then
    if hayRioAux( $rs'$ , posicion(dameUno( $cs$ ))) then
        true
    else
        hayRiosConflictivos( $rs, rs', sinUno(cs), cs'$ )
    fi
else
    if  $\neg$ vacio?( $cs'$ ) then
        if hayRioAux( $rs$ , posicion(dameUno( $cs'$ ))) then
            true
        else
            hayRiosConflictivos( $rs, rs', cs, sinUno(cs')$ )
        fi
    else
        false
    fi
fi

```

```

unirConstrucciones( $s, s', cs, cs'$ )  $\equiv$  if vacio?( $cs$ ) then
     $cs'$ 
else
    if vacio?( $cs'$ ) then
         $cs$ 
    else
        if hayConstruccionAux( $cs'$ , posicion(dameUno( $cs$ )))
        then
            Ag(decidir( $s, s', dameUno(cs),$ 
            enEstaPosicion( $cs', posicion(dameUno(cs))$ )),
            unirConstrucciones(sinUno( $cs, cs' \setminus$ 
            enEstaPosicion( $cs', posicion(dameUno(cs))$ )))
        else
            Ag(dameUno( $cs$ ), unirConstrucciones(sinUno( $cs,$ 
             $cs'$ )))
        fi
    fi
fi

decidir( $c, c'$ )  $\equiv$  if nivelDeConstruccion( $s, c$ ) = nivelDeConstruccion( $s', c'$ ) then
    if tipo( $c$ ) == "casa" then  $c$  else  $c'$  fi
else
    if nivelDeConstruccion( $s, c$ ) > nivelDeConstruccion( $s', c'$ ) then
         $c$ 
    else
         $c'$ 
    fi
fi

enEstaPosicion( $cs, p$ )  $\equiv$  if posicion(dameUno( $cs$ )) =  $p$  then
     $cs$ 
else
    enEstaPosicion(sinUno( $cs$ ),  $p$ )
fi

```

TAD Construcción

géneros constr

igualdad observacional

$$(\forall c, c' : \text{constr}) \ (c =_{\text{obs}} c' \iff (\text{tipo}(c) = \text{tipo}(c') \wedge \text{posicion}(c) = \text{posicion}(c')))$$

observadores básicos

$\text{tipo} \quad : \text{constr} \longrightarrow \text{string}$

$\text{posicion} : \text{constr} \longrightarrow \text{pos}$

generadores

$\text{generar} : \text{string} \longrightarrow \text{pos}$

otras operaciones

axiomas

$\text{tipo}(\text{generar}(t, p)) \quad \equiv t$

$\text{posicion}(\text{generar}(t, p)) \quad \equiv p$

Fin TAD

Aclaraciones Finales

En caso de querer unir dos SimCity's en que se solapen dos construcciones en la misma coordenada, ante igualdad de niveles elegimos la construcción que sea de tipo “casa” (de haberla), sino elegimos la construcción de mayor nivel.