

TAD CIUDADROBOTICA

géneros CiudadRobotica

exporta CiudadRobotica, Generadores, Observadores, Otras Operaciones

usa Nat, Bool, Conj(α), Estacion, Senda, Robot

igualdad observacional

$(\forall, :) (=_{\text{obs}} \iff ())$

observadores básicos

robotsC : CiudadRobotica \longrightarrow conj(Robot)

sendasC : CiudadRobotica \longrightarrow conj(Sendas)

ubicacionR : Robot $r \times$ CiudadRobotica $cr \longrightarrow$ estacion

$\{r \in \text{robotsC}(cr)\}$

infraccionesR : CiudadRobotica \times Robot \longrightarrow Nat

$\{r \in \text{robotsC}(cr)\}$

generadores

NuevaCiudad : conj(Sendas) $cs \longrightarrow$ CiudadRobotica

$\{\neg (cs =_{\text{obs}} \emptyset)\}$

AgRobot : Robot $r1 \times$ Estacion $e \times$ CiudadRobotica $cr \longrightarrow$ CiudadRobotica

$\left\{ \begin{aligned} &(\nexists r2: \text{Robot}) (r2 \in \text{robotsC}(cr)) \Rightarrow \text{RUR}(r1) =_{\text{obs}} \text{RUR}(r2) \wedge \\ &(\exists e2 : \text{Estacion}) e2 \in \text{estacionesC}(\text{sendasC}(cr)) \Rightarrow_{\text{L}} \text{Permitido?}(r, \text{conectadasPor}(e, e2, cr)) \end{aligned} \right\}$

MoverRobot : Robot $r \times$ Estacion $e \times$ CiudadRobotica $cr \longrightarrow$ CiudadRobotica

$\{(\exists s: \text{Senda}) s \in \text{sendasC}(cr) \Rightarrow_{\text{L}} s =_{\text{obs}} \text{conectadasPor}(e, \text{ubicacionR}(r, cr), cr)\}$

otras operaciones

estacionesC : conj(Sendas) \longrightarrow conj(estacion)

conectadasPor : Estacion $e1 \times$ Estacion $e2 \times$ CiudadRobotica $cr \longrightarrow$ Senda

$\left\{ (\exists s : \text{Senda}) s \in \text{sendasC}(cr) \Rightarrow_{\text{L}} (e1 =_{\text{obs}} \text{estacion1}(s) \wedge e2 =_{\text{obs}} \text{estacion2}(s)) \vee (e2 =_{\text{obs}} \text{estacion1}(s) \wedge e1 =_{\text{obs}} \text{estacion2}(s)) \right\}$

conectadasPorAux : Estacion $e1 \times$ Estacion $e2 \times$ conj(Senda) $cs \longrightarrow$ Senda

$\{\neg (cs =_{\text{obs}} \emptyset)\}$

inspeccionLunes : CiudadRobotica \times Estacion \longrightarrow CiudadRobotica

hayInfractor? : CiudadRobotica \times conj(Robot) \longrightarrow Bool

eliminarRobot : Robot \times CiudadRobotica \longrightarrow CiudadRobotica

seleccionarInfractor : CiudadRobotica \times Estacion \longrightarrow Robot

filtroRobEst : CiudadRobotica \times conj(Robot) \times Estacion \longrightarrow conj(Robot)

Max : CiudadRobotica \times conj(Robot) \times Nat \longrightarrow Nat

maximosInf : CiudadRobotica \times conj(Robot) \times Nat \longrightarrow conj(Robot)

: \longrightarrow

axiomas

robotsC(NuevaCiudad(cs)) $\equiv \emptyset$

robotsC(AgRobot(r, e, cr)) $\equiv \text{Ag}(e, \text{robotsC}(cr))$

robotsC(MoverRobot(r, e, cr)) $\equiv \text{robotsC}(cr)$

sendasC(NuevaCiudad(cs)) $\equiv cs$

sendasC(AgRobot(r, e, cr)) $\equiv \text{sendasC}(cr)$

sendasC(MoverRobot(r, e, cr)) $\equiv \text{sendasC}(cr)$

ubicacionR(r1, AgRobot(r2, e, cr)) \equiv **if** $r1 =_{\text{obs}} r2$ **then** e **else** ubicacionR(r1, cr) **fi**

ubicacionR(r1, MoverRobot(r2, e, cr)) \equiv **if** $r1 =_{\text{obs}} r2$ **then** e **else** ubicacionR(r1, cr) **fi**

infraccionesR(NuevaCiudad(cs), r) $\equiv 0$

infraccionesR(AgRobot(r1, e, cr), r2) $\equiv \text{infraccionesR}(cr)$

infraccionesR(MoverRobot(r1, e, cr), r2) \equiv **if** $r1 =_{\text{obs}} r2$

then if $\neg \text{Permitido?}(\text{conectadasPor}(e, \text{ubicacionR}(r2, cr), cr), r2)$

then $1 + \text{infraccionesR}(cr, r2)$

else $\text{infraccionesR}(cr, r2)$

else $\text{infraccionesR}(cr, r2)$

estacionesC(cs) \equiv **if** $cs =_{\text{obs}} \emptyset$ **then** \emptyset

else $\text{Ag}(\text{estacion1}(\text{dameUno}(cs)), \text{Ag}(\text{estacion2}(\text{dameUno}(cs)), \text{estacionesC}(\text{sinUno}(cs))))$

conectadasPor(e1, e2, cr) $\equiv \text{conectadasPorAux}(e1, e2, \text{sendasC}(cr))$

conectadasPorAux(e1, e2, cs) \equiv **if** $(e1 =_{\text{obs}} \text{estacion1}(\text{dameUno}(cs)) \wedge e2 =_{\text{obs}} \text{estacion2}(\text{dameUno}(cs)))$

$\vee (e2 =_{\text{obs}} \text{estacion1}(\text{dameUno}(cs)) \wedge e1 =_{\text{obs}} \text{estacion2}(\text{dameUno}(cs)))$

then $\text{dameUno}(cs)$

else $\text{conectadasPorAux}(e1, e2, \text{sinUno}(cs))$

```

inspeccionLunes(cr, e)  $\equiv$  if hayInfractor?(cr, filtroRobEst(cr, robotsC(cr), e))
                        then eliminarRobot(seleccionarInfractor(cr, e), cr)
                        else cr
hayInfractor?(cr, c)  $\equiv$  if c =obs  $\emptyset$  then false
                        else if  $\neg$ (infraccionesR(dameUno(c)) =obs 0)
                        then true
                        else hayInfractor?(cr, sinUno(c))
seleccionarInfractor(cr, e)  $\equiv$  dameUno(maximosInf(cr, filtroRobEst(cr, robotsC(cr), e),
                        Max(cr, filtroRobEst(cr, robotsC(cr), e), 0)))
filtroRobEst(cr, c, e)  $\equiv$  if robotsC(cr) =obs  $\emptyset$  then  $\emptyset$ 
                        else if ubicacionR(dameUno(c), cr) =obs e
                        then Ag(dameUno(c), filtroRobEst(cr, sinUno(c), e))
                        else filtroRobEst(cr, sinUno(c), e)
Max(cr, c, n)  $\equiv$  if c =obs  $\emptyset$  then n
                else if infraccionesR(dameUno(c), cr) > n
                then Max(cr, sinUno(c), infraccionesR(dameUno(c), cr))
                else Max(cr, sinUno(c), n)
maximosInf(cr, c, n)  $\equiv$  if c =obs  $\emptyset$  then  $\emptyset$ 
                    else if infraccionesR(cr, dameUno(c)) =obs n
                    then Ag(dameUno(c), maximosInf(cr, sinUno(c), n))
                    else maximosInf(cr, sinUno(c), n)
eliminarRobot(r, NuevaCiudad(cs))  $\equiv$  NuevaCiudad(cs)
eliminarRobot(r1, AgRobot(r2, e, cr))  $\equiv$  if r1 =obs r2 then cr else eliminarRobot(r1, cr)
eliminarRobot(r1, MoverRobot(r2, e, cr))  $\equiv$  if r1 =obs r2 then MoverRobot(r2, e, eliminarRobot(r1, cr))
                        else eliminarRobot(r1, cr)

```

Fin TAD

TAD ROBOT

géneros Robot

exporta Robot, Generadores, Observadores, Otras Operaciones

usa Nat, Bool, Caracteristica, Conj(α)

igualdad observacional

$$(\forall r1, r2 : \text{Robot}) \left(r1 =_{\text{obs}} r2 \iff \left(\text{RUR}(r1) =_{\text{obs}} \text{RUR}(r2) \wedge \text{CaracteristicasR}(r1) =_{\text{obs}} \text{CaracteristicasR}(r2) \right) \right)$$

observadores básicos

RUR : Robot \longrightarrow Nat

CaracteristicasR : Robot \longrightarrow Conj(Caracteristica)

generadores

NuevoRobot : Nat \longrightarrow Robot

AgCaracteristica : Robot \times Caracteristica \longrightarrow Robot

otras operaciones

TieneCaracteristica? : Robot \times Caracteristica \longrightarrow Bool

axiomas $\forall r : \text{Robot}$

RUR(NuevoRobot(n)) \equiv n

RUR(AgCaracteristica(r, c)) \equiv RUR(r)

CaracteristicasR(NuevoRobot(n)) $\equiv \emptyset$

CaracteristicasR(AgCaracteristica(r, c)) \equiv Ag(c, CaracteristicasR(r))

TieneCaracteristica?(r, c) \equiv c \in CaracteristicasR(r)

Fin TAD

TAD SENDA

géneros Senda

exporta Senda, Generadores, Observadores, Otras Operaciones

usa Bool, Robot, Restriccion, Estacion

igualdad observacional

$(\forall, :) (=_{\text{obs}} \iff ())$

observadores básicos

estacion1 : Senda \longrightarrow Estacion

estacion2 : Senda \longrightarrow Estacion

RestriccionS : Senda \longrightarrow Restriccion

generadores

SendaNueva : Estacion $e1 \times$ Estacion $e2 \times$ Restriccion \longrightarrow Senda

$\{\neg(e1 =_{\text{obs}} e2)\}$

otras operaciones

Permitido? : Senda \times Robot \longrightarrow Bool

axiomas $\forall r: \text{Robot}, \forall s: \text{Senda}, \forall e1, e2: \text{Estacion}$

estacion1(SendaNueva($e1, e2, r$)) $\equiv e1$

estacion2(SendaNueva($e1, e2, r$)) $\equiv e2$

RestriccionS(SendaNueva($e1, e2, r$)) $\equiv r$

Permitido?(s, r) \equiv CumpleCondicion?(RestriccionS(s), r)

Fin TAD

TAD RESTRICCION

géneros Restriccion

exporta Generadores, Observadores

usa Bool, Robot

igualdad observacional

$(\forall, :) (=_{\text{obs}} \iff ())$

observadores básicos

CumpleCondicion? : Restriccion \times robot \longrightarrow bool

generadores

Caract : Caracteristica \longrightarrow Restriccion

• OR • : Restriccion \times Restriccion \longrightarrow Restriccion

• AND • : Restriccion \times Restriccion \longrightarrow Restriccion

NOT • : Restriccion \longrightarrow Restriccion

axiomas $\forall a, b: \text{Restriccion}, \forall s: \text{Caracteristica}$

CumpleCondicion?(Caract(s), r) \equiv TieneCaracteristica?(s, r)

CumpleCondicion?(a OR b, r) \equiv CumpleCondicion?(a, r) \vee CumpleCondicion?(b, r)

CumpleCondicion?(a AND b, r) \equiv CumpleCondicion?(a, r) \wedge CumpleCondicion?(b, r)

CumpleCondicion?(NOT a, r) $\equiv \neg$ CumpleCondicion?(a, r)

Fin TAD