# Trabajo Practico I, Alta Seguridad nos cuida

Algoritmos y Estructuras de Datos II, DC, UBA.

## Índice

1. TAD Universidad 2

2. TAD Agente 3

# 1. TAD Universidad

**TAD** Universidad

**géneros** uni

**exporta** uni, Generadores, Observadores Basicos

**usa** Nat, Conju($\alpha$), Bool, Tupla($\alpha_1, \ldots, \alpha_n$)

**igualdad observacional**

$$(\forall u, u' : \text{uni}) \left( u =_{\text{obs}} u' \iff \begin{pmatrix} \text{alto?(u)} =_{\text{obs}} \text{alto?(u')} \wedge \text{ancho?(u)} =_{\text{obs}} \text{ancho?(u')} \wedge \text{obstaculos?(u)} =_{\text{obs}} \text{obstaculos?(u')} \wedge \text{agentes?(u)} =_{\text{obs}} \text{agentes?(u')} \wedge \\ \text{estudiantes?(u)} =_{\text{obs}} \text{estudiantes?(u')} \wedge \text{hippies?(u)} =_{\text{obs}} \text{hippies?(u')} \end{pmatrix} \right)$$

**observadores básicos**

alto? : uni $\longrightarrow$ nat

ancho? : uni $\longrightarrow$ nat

obstaculos? : uni $\longrightarrow$ conj($pos$)

agentes? : uni $\longrightarrow$ conj($\langle as, pos \rangle$)

estudiantes? : uni $\longrightarrow$ conj($\langle est, pos \rangle$)

hippies? : uni $\longrightarrow$ conj($pos$)

**generadores**

nuevaUni : nat $\times$ nat $\times$ conj($pos$) $\times$ conj($\langle as \times pos \rangle$) $\longrightarrow$ uni

agregarE : uni $\times$ ($\langle est \times pos \rangle$) $\longrightarrow$ uni

agregarH : uni $\times$ pos $\longrightarrow$ uni

**otras operaciones**

cuantosE : uni $\longrightarrow$ nat

cuantosH : uni $\longrightarrow$ nat

masVigilante : uni $\longrightarrow$ As

queHay : uni $\times$ pos $\longrightarrow$ tipoEnum

moverTodo : conj($\langle as \times pos \rangle$) $\times$ conj($pos$) $\times$ conj($\langle est \times pos \rangle$) $\times$ nat $\times$ nat $\times$ conj(pos) $\longrightarrow$ $\langle conj(\langle as, pos \rangle), conj(p$

**axiomas** $\forall$ :

Observadores Basicos

agentes?(nuevaUni(al,an,co,cAs)) $\equiv$ $\Pi_1$(moverTodo(cAs, $\emptyset$, $\emptyset$, al, an, co))

agentes?(agregarH(uni,pos)) $\equiv$ $\Pi_1$ (moverTodo(agentes?(uni), pos U hippies?(uni), estudiantes?(uni), alto?(uni), ancho?(uni), obstaculos?(uni)))

agentes?(agregarE(uni,$\langle est, pos \rangle$)) $\equiv$ $\Pi_1$ (moverTodo(agentes?(uni), hippies?(uni), $\langle est, pos \rangle$ U estudiantes?(uni), alto?(uni), ancho?(uni), obstaculos?(uni)))

hippies?(nuevaUni(al,an,co,cAs)) $\equiv$ $\emptyset$

hippies?(agregarH(uni,pos)) $\equiv$ $\Pi_2$(moverTodo(agentes?(uni), pos U hippies?(uni), estudiantes?(uni), alto?(uni), ancho?(uni), obstaculos?(uni)))

hippies?(agregarE(uni,$\langle est, pos \rangle$)) $\equiv$ $\Pi_2$(moverTodo(agentes?(uni), hippies?(uni), $\langle est, pos \rangle$ U estudiantes?(uni), alto?(uni), ancho?(uni), obstaculos?(uni)))

estudiantes?(nuevaUni(al,an,co,cAs)) $\equiv$ $\emptyset$

estudiantes?(agregarH(uni,pos)) $\equiv$ $\Pi_3$(moverTodo(agentes?(uni), pos U hippies?(uni), estudiantes?(uni), alto?(uni), ancho?(uni), obstaculos?(uni)))

estudiantes?(agregarE(uni,⟨*est*, *pos*⟩)) ≡ $\Pi_3$(moverTodo(agentes?(uni), hippies?(uni), ⟨*est*, *pos*⟩ U estudiantes?(uni), alto?(uni), ancho?(uni), obstaculos?(uni)))

alto?(nuevaUni(al,an,co,cAs)) ≡ al

alto?(agregarH(uni,pos)) ≡ alto?(uni)

alto?(agregarE(uni,⟨*est*, *pos*⟩)) ≡ alto?(uni)

ancho?(nuevaUni(al,an,co,cAs)) ≡ an

ancho?(agregarH(uni,pos)) ≡ ancho?(uni)

ancho?(agregarE(uni,⟨*est*, *pos*⟩)) ≡ ancho?(uni)

obstaculos?(nuevaUni(al,an,co,cAs)) ≡ co

obstaculos?(agregarH(uni,pos)) ≡ obstaculos?(uni)

obstaculos?(agregarE(uni,⟨*est*, *pos*⟩)) ≡ obstaculos?(uni)

Otras Operaciones

cuantosE(uni) ≡ # estudantes?(uni)

cuantosH(uni) ≡ # hippies?(uni)

masVigilante(uni) ≡ maxAtrapados(agentes?(uni))

maxAtrapados(cAs) ≡ max( hippiesAtrapados(dameUno(cAs)), maxAtrapados(sinUno(cAs)) )

**Fin TAD**

## 2. TAD Agente

**TAD** Agente

**géneros** as

**exporta** as, Generadores, Observadores Basicos, hippiesAtrapados

**usa** Nat, Bool

**igualdad observacional**

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

**observadores básicos**

numPlaca : as ⟶ nat

hippiesAtrapados : as ⟶ nat

numSanciones : as ⟶ nat

**generadores**

nuevoAs : nat ⟶ as

capturarH : as a ⟶ as

sancionar : as ⟶ as

**otras operaciones**

inactivo? : as ⟶ bool

**axiomas** ∀ :

Observadores Basicos

numPlaca(nuevoAs(n)) ≡ n

numPlaca(capturarH(a)) ≡ numPlaca(a)

numPlaca(sancionar(a)) $\equiv$ numPlaca(a)

hippiesAtrapados(nuevoAs(n)) $\equiv$ 0

hippiesAtrapados(capturarH(a)) $\equiv$ 1+ hippiesAtrapados(a)

hippiesAtrapados(sancionar(a)) $\equiv$ hippiesAtrapados(a)

numSanciones(nuevoAs(n)) $\equiv$ 0

numSanciones(capturarH(a)) $\equiv$ numSanciones(a)

numSanciones(sancionar(a)) $\equiv$ 1+ numSanciones(a)

Otras Operaciones

inactivo?(a) $\equiv$ **if** numSanciones(a) $> 3$ **then** true **else** false **fi**

**Fin TAD**