

Algoritmos y Estructuras de Datos II

Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Trabajo Práctico 1: "Pacalgo2"

Grupo: tomarAgua()

Integrante	LU	Correo electrónico
Reyna Maciel, Guillermo José	393/20	guille.j.reyna@gmail.com
Casado Farall, Joaquin	072/20	joakinfarall@gmail.com
Fernández Spandau, Luciana	368/20	fernandezspandau@gmail.com
Chumacero, Carlos Nehemias	492/20	chumacero2013@gmail.com

Reservado para la cátedra

Instancia	Docente	Nota
Primera entrega		
Segunda entrega		

$\text{hayFantasmasCerca}(c, f) \equiv \text{if } \text{vacío?}(f) \text{ then } \text{false}$
 $\quad \text{else } |\pi_1(c) - \pi_1(\text{dameUno}(f))| + |\pi_2(c) - \pi_2(\text{dameUno}(f))| < 3$
 $\quad \vee \text{hayFantasmasCerca}(\text{sinUno}(f))$
 $\text{esMovimientoValido}(p, x, y) \equiv \neg \text{terminóElJuego}(p) \wedge \text{enRango?}(\langle x, y \rangle, \text{dimensión}(\text{mapa}(p))) \wedge \neg(\langle x, y \rangle \in \text{paredes}(\text{mapa}(p)))$
 $\quad \text{fi}$

Fin TAD

TAD MAPA

géneros mapa
exporta mapa, generadores, observadores, enRango?, estaOcupado?
usa BOOL, NAT, CONJUNTO(α), COORDENADAS, DIMENSIÓN
igualdad observacional

$$(\forall m, m' : \text{mapa}) \left(m =_{\text{obs}} m' \iff \left(\begin{array}{l} \text{dimensión}(m) =_{\text{obs}} \text{dimensión}(m') \wedge \\ \text{paredes}(m) =_{\text{obs}} \text{paredes}(m') \wedge \\ \text{fantasmas}(m) =_{\text{obs}} \text{fantasmas}(m') \wedge \\ \text{puntoDeSalida}(m) =_{\text{obs}} \text{puntoDeSalida}(m') \wedge \\ \text{puntoDeLlegada}(m) =_{\text{obs}} \text{puntoDeLlegada}(m') \end{array} \right) \right)$$

observadores básicos

$\text{dimensión} : \text{mapa} \longrightarrow \text{dimensión}$
 $\text{paredes} : \text{mapa} \longrightarrow \text{conj}(\text{coordenadas})$
 $\text{fantasmas} : \text{mapa} \longrightarrow \text{conj}(\text{coordenadas})$
 $\text{puntoDeSalida} : \text{mapa} \longrightarrow \text{coordenadas}$
 $\text{puntoDeLlegada} : \text{mapa} \longrightarrow \text{coordenadas}$

generadores

$\text{nuevoMapa} : \text{dimensión } d \times \text{coordenadas } \textit{inicio} \longrightarrow \text{mapa} \quad \{\pi_1(d) * \pi_2(d) \geq 2 \wedge \neg(\textit{inicio} = \textit{fin}) \wedge \text{enRango?}(\textit{inicio}, d) \wedge \text{enRango?}(\textit{fin}, d)\}$
 $\quad \times \text{coordenadas } \textit{fin}$
 $\text{agregarFantasma} : \text{mapa } m \times \text{coordenadas } c \longrightarrow \text{mapa}$
 $\quad \{\text{enRango?}(c, \text{dimensión}(m)) \wedge \neg \text{estaOcupado?}(c, m)\}$
 $\text{agregarPared} : \text{mapa } m \times \text{coordenadas } c \longrightarrow \text{mapa}$
 $\quad \{\text{enRango?}(c, \text{dimensión}(m)) \wedge \neg \text{estaOcupado?}(c, m)\}$

otras operaciones

$\text{esRango?} : \text{coordenadas } \times \text{dimensión} \longrightarrow \text{bool}$
 $\text{estaOcupado?} : \text{coordenadas } c \times \text{mapa } m \longrightarrow \text{bool} \quad \{\text{enRango?}(c, \text{dimension}(m))\}$

axiomas $\forall m: \text{mapa}, \forall i, f, c: \text{coordenadas}, \forall d: \text{dimensión}$

$\text{dimension}(\text{nuevoMapa}(d, i, f)) \equiv d$
 $\text{dimension}(\text{agregarFantasma}(m, c)) \equiv \text{dimension}(m)$
 $\text{dimension}(\text{agregarPared}(m, c)) \equiv \text{dimension}(m)$

 $\text{paredes}(\text{nuevoMapa}(d, i, f)) \equiv \emptyset$
 $\text{paredes}(\text{agregarFantasma}(m, c)) \equiv \text{paredes}(m)$
 $\text{paredes}(\text{agregarPared}(m, c)) \equiv \text{Ag}(c, \text{paredes}(m))$

 $\text{fantasmas}(\text{nuevoMapa}(d, i, f)) \equiv \emptyset$
 $\text{fantasmas}(\text{agregarFantasma}(m, c)) \equiv \text{Ag}(c, \text{paredes}(m))$
 $\text{fantasmas}(\text{agregarPared}(m, c)) \equiv \text{paredes}(m)$

 $\text{puntoDeSalida}(\text{nuevoMapa}(d, i, f)) \equiv i$
 $\text{puntoDeSalida}(\text{agregarFantasma}(m, c)) \equiv \text{puntoDeSalida}(m)$
 $\text{puntoDeSalida}(\text{agregarPared}(m, c)) \equiv \text{puntoDeSalida}(m)$

 $\text{puntoDeLlegada}(\text{nuevoMapa}(d, i, f)) \equiv f$
 $\text{puntoDeLlegada}(\text{agregarFantasma}(m, c)) \equiv \text{puntoDeLlegada}(m)$
 $\text{puntoDeLlegada}(\text{agregarPared}(m, c)) \equiv \text{puntoDeLlegada}(m)$

enRango?(c, d)

$\equiv \pi_1(c) < \pi_1(d) \wedge \pi_2(c) < \pi_2(d)$

estaOcupado?(c, m)

$\equiv \neg(c \in (\text{paredes}(m) \cup \text{fantasmas}(m) \cup \{\text{puntoDeSalida}(m), \text{puntoDeLlegada}(m)\}))$

Fin TAD

2. Ejercicios - Parte 2

TAD DIMENSIÓN, COORDENADAS

extiende TUPLA(NAT, NAT)

otras operaciones

$\bullet = \bullet : \text{tupla}(\text{nat} \times \text{nat}) \times \text{tupla}(\text{nat} \times \text{nat}) \longrightarrow \text{bool}$

axiomas $\forall \text{tupla}(\text{nat}, \text{nat}): a_1, a_2$

$a_1 = a_2 \equiv \pi_1(a_1) = \pi_1(a_2) \wedge \pi_2(a_1) = \pi_2(a_2)$

Fin TAD

TAD PARTIDA

géneros partida

exporta partida, generadores, observadores, seAsustó?, ganó?

usa BOOL, NAT, CONJUNTO(α), MAPA, COORDENADAS, DIMENSIÓN

igualdad observacional

$$(\forall p, p' : \text{partida}) \left(p =_{\text{obs}} p' \iff \begin{pmatrix} \text{jugador}(p) =_{\text{obs}} \text{jugador}(p') \wedge \\ \text{mapa}(p) =_{\text{obs}} \text{mapa}(p') \wedge \\ \text{chocolates}(p) =_{\text{obs}} \text{chocolates}(p') \wedge \\ \text{movimientosConInmunidad}(p) \\ =_{\text{obs}} \text{movimientosConInmunidad}(p') \wedge \\ \text{puntaje}(p) =_{\text{obs}} \text{puntaje}(p') \end{pmatrix} \right)$$

observadores básicos

jugador	: partida	\longrightarrow coordenadas
mapa	: partida	\longrightarrow mapa
chocolates	: partida	\longrightarrow conj(coordenadas)
movimientosConInmunidad	: partida	\longrightarrow nat
puntaje	: partida	\longrightarrow nat

generadores

iniciarPartida	: mapa $m \times \text{conj}(\text{coordenadas})$ <i>chocolates</i>	\longrightarrow partida
		$\left\{ \begin{array}{l} (\forall c: \text{coordenadas}) (c \in \text{chocolates} \Rightarrow \text{enRango?}(c, \text{dimensión}(m))) \wedge_L \\ \neg(c \in \text{puntoDeSalida}(m) \vee c \in \text{paredes}(m)) \end{array} \right\}$
arriba	: partida p	\longrightarrow partida
		$\{\text{esMovimientoValido}(p, \pi_1(\text{jugador}(p)), \pi_2(\text{jugador}(p))+1)\}$
abajo	: partida p	\longrightarrow partida
		$\{\text{esMovimientoValido}(p, \pi_1(\text{jugador}(p)), \pi_2(\text{jugador}(p))-1)\}$
derecha	: partida p	\longrightarrow partida
		$\{\text{esMovimientoValido}(p, \pi_1(\text{jugador}(p))+1, \pi_2(\text{jugador}(p)))\}$
izquierda	: partida p	\longrightarrow partida
		$\{\text{esMovimientoValido}(p, \pi_1(\text{jugador}(p))-1, \pi_2(\text{jugador}(p)))\}$

otras operaciones

seAsustó?	: partida	\longrightarrow bool
ganó?	: partida	\longrightarrow bool
terminóElJuego?	: partida	\longrightarrow bool
hayFantasmasCerca	: coordenadas \times conj(coordenadas)	\longrightarrow bool
esMovimientoValido	: partida \times nat \times nat	\longrightarrow bool

inmuneAlMover : partida \times coordenadas \longrightarrow nat

axiomas $\forall p$: partida, $\forall m$: mapa, $\forall c$: coordenadas, $\forall f, c$: conj(coordenadas)

jugador(iniciarPartida(m))	\equiv puntoDeSalida(m)
jugador(arriba(p))	$\equiv \langle \pi_1(\text{jugador}(p)), \pi_2(\text{jugador}(p))+1 \rangle$
jugador(abajo(p))	$\equiv \langle \pi_1(\text{jugador}(p)), \pi_2(\text{jugador}(p))-1 \rangle$
jugador(derecha(p))	$\equiv \langle \pi_1(\text{jugador}(p))+1, \pi_2(\text{jugador}(p)) \rangle$
jugador(izquierda(p))	$\equiv \langle \pi_1(\text{jugador}(p))-1, \pi_2(\text{jugador}(p)) \rangle$
mapa(iniciarPartida(m))	$\equiv m$
mapa(arriba(p))	$\equiv \text{mapa}(p)$
mapa(abajo(p))	$\equiv \text{mapa}(p)$
mapa(izquierda(p))	$\equiv \text{mapa}(p)$
mapa(derecha(p))	$\equiv \text{mapa}(p)$
chocolates(iniciarPartida(m, c))	$\equiv c$
chocolates(arriba(p))	$\equiv \text{chocolates}(p) - \{\text{jugador}(\text{arriba}(p))\}$
chocolates(abajo(p))	$\equiv \text{chocolates}(p) - \{\text{jugador}(\text{abajo}(p))\}$
chocolates(izquierda(p))	$\equiv \text{chocolates}(p) - \{\text{jugador}(\text{izquierda}(p))\}$
chocolates(derecha(p))	$\equiv \text{chocolates}(p) - \{\text{jugador}(\text{derecha}(p))\}$
movimientosConInmunidad(iniciarPartida(m, c))	$\equiv 0$
movimientosConInmunidad(arriba(p))	$\equiv \text{inmuneAlMover}(p, \text{jugador}(\text{arriba}(p)))$
movimientosConInmunidad(abajo(p))	$\equiv \text{inmuneAlMover}(p, \text{jugador}(\text{abajo}(p)))$
movimientosConInmunidad(izquierda(p))	$\equiv \text{inmuneAlMover}(p, \text{jugador}(\text{izquierda}(p)))$
movimientosConInmunidad(derecha(p))	$\equiv \text{inmuneAlMover}(p, \text{jugador}(\text{derecha}(p)))$
puntaje(iniciarPartida(m, c))	$\equiv 0$
puntaje(arriba(p))	$\equiv \text{puntaje}(p)+1$
puntaje(abajo(p))	$\equiv \text{puntaje}(p)+1$
puntaje(izquierda(p))	$\equiv \text{puntaje}(p)+1$
puntaje(derecha(p))	$\equiv \text{puntaje}(p)+1$
seAsustó?(p)	$\equiv \text{hayFantasmasCerca}(\text{jugador}(p), \text{fantasmas}(\text{mapa}(p))) \wedge$ $\text{movimientosConInmunidad}(p) = 0$
ganó?(p)	$\equiv \text{jugador}(p) = \text{puntoDeLlegada}(\text{mapa}(p))$
terminóElJuego?(p)	$\equiv \text{seAsustó}(p) \vee \text{ganó}(p)$
hayFantasmasCerca(c, f)	\equiv if vacío?(f) then false else $ \pi_1(c) - \pi_1(\text{dameUno}(f)) +$ $ \pi_2(c) - \pi_2(\text{dameUno}(f)) < 3$ $\vee \text{hayFantasmasCerca}(\text{sinUno}(f))$ fi
esMovimientoValido(p, x, y)	$\equiv \neg \text{terminóElJuego}(p) \wedge$ $\text{enRango?}(\langle x, y \rangle, \text{dimensión}(\text{mapa}(p))) \wedge$ $\neg(\langle x, y \rangle \in \text{paredes}(\text{mapa}(p)))$
inmuneAlMover(p, c)	\equiv if $c \in \text{chocolates}(p)$ then 10 else $\text{max}(0, \text{movimientosConInmunidad}(p)-1)$ fi

Fin TAD

TAD MAPA

géneros	mapa
exporta	mapa, generadores, observadores, enRango?, estaOcupado?
usa	BOOL, NAT, CONJUNTO(α), COORDENADAS, DIMENSIÓN

igualdad observacional

$$(\forall m, m' : \text{mapa}) \left(m =_{\text{obs}} m' \iff \left(\begin{array}{l} \text{dimensión}(m) =_{\text{obs}} \text{dimensión}(m') \wedge \\ \text{paredes}(m) =_{\text{obs}} \text{paredes}(m') \wedge \\ \text{fantasmas}(m) =_{\text{obs}} \text{fantasmas}(m') \wedge \\ \text{puntoDeSalida}(m) =_{\text{obs}} \text{puntoDeSalida}(m') \wedge \\ \text{puntoDeLlegada}(m) =_{\text{obs}} \text{puntoDeLlegada}(m') \end{array} \right) \right)$$

observadores básicos

dimensión	: mapa	→	dimensión
paredes	: mapa	→	conj(coordenadas)
fantasmas	: mapa	→	conj(coordenadas)
puntoDeSalida	: mapa	→	coordenadas
puntoDeLlegada	: mapa	→	coordenadas

generadores

nuevoMapa	: dimensión $d \times$ coordenadas $\text{inicio} \rightarrow$ mapa \times coordenadas fin	
		$\left\{ \pi_1(d) * \pi_2(d) \geq 2 \wedge \neg (\text{inicio} = \text{fin}) \wedge \right.$ $\left. \text{enRango?}(\text{inicio}, d) \wedge \text{enRango?}(\text{fin}, d) \right\}$
agregarFantasma	: mapa $m \times$ coordenadas $c \rightarrow$ mapa	$\{ \text{enRango?}(c, \text{dimensión}(m)) \wedge \neg \text{estaOcupado?}(c, m) \}$
agregarPared	: mapa $m \times$ coordenadas $c \rightarrow$ mapa	$\{ \text{enRango?}(c, \text{dimensión}(m)) \wedge \neg \text{estaOcupado?}(c, m) \}$

otras operaciones

esRango?	: coordenadas \times dimensión	→	bool
estaOcupado?	: coordenadas $c \times$ mapa m	→	bool $\{ \text{enRango?}(c, \text{dimension}(m)) \}$

axiomas $\forall m: \text{mapa}, \forall i, f, c: \text{coordenadas}, \forall d: \text{dimensión}$

dimension(nuevoMapa(d, i, f))	$\equiv d$
dimension(agregarFantasma(m, c))	$\equiv \text{dimension}(m)$
dimension(agregarPared(m, c))	$\equiv \text{dimension}(m)$
paredes(nuevoMapa(d, i, f))	$\equiv \emptyset$
paredes(agregarFantasma(m, c))	$\equiv \text{paredes}(m)$
paredes(agregarPared(m, c))	$\equiv \text{Ag}(c, \text{paredes}(m))$
fantasmas(nuevoMapa(d, i, f))	$\equiv \emptyset$
fantasmas(agregarFantasma(m, c))	$\equiv \text{Ag}(c, \text{paredes}(m))$
fantasmas(agregarPared(m, c))	$\equiv \text{paredes}(m)$
puntoDeSalida(nuevoMapa(d, i, f))	$\equiv i$
puntoDeSalida(agregarFantasma(m, c))	$\equiv \text{puntoDeSalida}(m)$
puntoDeSalida(agregarPared(m, c))	$\equiv \text{puntoDeSalida}(m)$
puntoDeLlegada(nuevoMapa(d, i, f))	$\equiv f$
puntoDeLlegada(agregarFantasma(m, c))	$\equiv \text{puntoDeLlegada}(m)$
puntoDeLlegada(agregarPared(m, c))	$\equiv \text{puntoDeLlegada}(m)$
enRango?(c, d)	$\equiv \pi_1(c) < \pi_1(d) \wedge \pi_2(c) < \pi_2(d)$
estaOcupado?(c, m)	$\equiv \neg(c \in (\text{paredes}(m) \cup \text{fantasmas}(m) \cup \{ \text{puntoDeSalida}(m), \text{puntoDeLlegada}(m) \}))$

Fin TAD

3. Aclaraciones

- Estamos utilizando un sistema de coordenadas al estilo cartesiano, donde la primera coordenada representa el eje horizontal y aumenta hacia la derecha y la segunda coordenada representa el eje vertical y aumenta hacia arriba.
- Por conveniencia, interpretamos que los chocolates no pueden solaparse con el punto de salida (porque para comer un chocolate el jugador debe moverse a la coordenada en donde se encuentra el chocolate) ni con paredes. Interpretamos que sí pueden solaparse con fantasmas y con el punto de llegada.
- En la axiomatización del observador chocolates de partida, para simplificar la escritura utilizamos la propiedad de que $c - \{a\} = c$ cuando $a \notin c$, pues si la intersección de dos conjuntos es vacía, la resta no los modifica.
- Interpretamos que los movimientos con inmunidad no son acumulativos, sino que se recargan. Es decir, comer un chocolate mientras el jugador ya tiene inmunidad de otro chocolate no le suma 10 movimientos más de inmunidad, sino que le recarga los movimientos con inmunidad a un máximo de 10 (similar a la funcionalidad de la inmunidad en el Pac-Man original).
- Si bien esta especificación permite partidas inganables o imperdibles (e.g. el punto de llegada esté rodeado de paredes, o que haya un fantasma al lado del punto de salida), consideramos que restringir estas posibilidades sería sobre-especificar, puesto que el enunciado no las menciona.
- Interpretamos que el punto de salida y el punto de llegada no pueden estar en el mismo espacio y que estos no se pueden solapar con fantasmas o paredes, puesto que el enunciado dice que se asignan dos casilleros especiales para ellos.
- En la axiomatización de `inmunidadAlMover`, la función `máx` se asume como la del TAD Natural, pero el parámetro `movimientosConInmunidad(p) - 1` puede ser -1 cuando `movimientosConInmunidad(p) = 0`. Asumimos que la función `máximo` se extiende con los enteros.
- El enunciado dice que el puntaje se debe poder calcular al final de la partida. En nuestra especificación el puntaje se puede calcular en cualquier punto de la partida, y en particular cuando la partida finalizó y el jugador ganó.