

Algoritmos y Estructuras de Datos II

Segundo Cuatrimestre de 2016

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

Trabajo Practico 1

Especificacion

Grupo 1

Integrante	LU	Correo electrónico
Integrante 1	Nro/YY	mail@dc.uba.ar
Integrante 2	Nro/YY	mail@dc.uba.ar
Integrante 3	Nro/YY	mail@dc.uba.ar
Integrante 4	Nro/YY	mail@dc.uba.ar

Reservado para la cátedra

Instancia	Docente	Nota
Primera entrega		
Segunda entrega		

Índice

1. Especificacion

Esta es una especificacion del Trabajo Practico 1 del 2^{do} cuatrimestre del 2016 presentada por la catedra para la realizacion del Trabajo Practico 2. Ver enunciado:

<http://www.dc.uba.ar/materias/aed2/2016/2c/descargas/tps/tp1/view>

2. Renombres de TADs

TAD TIPO es STRING

TAD POKEMONES es DICCIONARIO(NAT, TIPO)

TAD POKEMON es TUPLA(NAT, TIPO)

3. TAD Juego

TAD POKEMONGO

géneros juego

observadores básicos

mapa	: Juego	→ Mapa
jugadores	: Juego	→ conj(Jugador)
posicionJugador	: Jugador $j \times$ Juego pGo	→ Coordenada $\{j \in \text{jugadores}(pGo)\}$
pokemones	: Juego	→ Pokemones
posicionPokemon	: Pokemon $p \times$ Juego j	→ Coordenada $\{\text{def?}(\Pi_1(p), \text{pokemones}(j))\}$

generadores

nuevoJuego	: Mapa	→ Juego
agJugador	: Jugador $j \times$ Coordenada $c \times$ Juego pGo	→ Juego $\{\text{esPosicionValidaJugador}(c, pGo) \wedge_L \text{id}(j) \notin \text{jugadores}(pGo)\}$
agPokemon	: Nat $n \times$ Tipo $t \times$ Coordenada $c \times$ Juego pGo	→ Juego $\{n \notin \text{pokemones}(j) \wedge_L \text{esPosicionValidaPokemon}(c, pGo)\}$
informarPosicion	: Jugador $j \times$ Coordenada $c \times$ Juego pGo	→ Juego $\{\text{puedeMoverse}(j, pGo) \wedge_L \text{esPosicionValidaEnJuego}(c, pGo)\}$
sumarEspera	: conj(jugador) $cj \times$ Coordenada $c \times$ Juego pGo	→ Juego

otras operaciones

atraparPokemon	: Pokemon $p \times$ Jugador $j \times$ Juego pGo	→ Juego
esPosicionValidaJugador	: Coordenada $c \times$ Juego j	→ bool
esPosicionValidaPokemon	: Coordenada $c \times$ Juego j	→ bool
esPosicionValidaEnJuego	: Coordenada $c \times$ Juego j	→ bool
puedeMoverse	: Jugador $j \times$ Juego pGo	→ bool
actualizarContadores	: conj(Jugador) $cj \times$ Coordenada $c \times$ Juego pGo	→ conj(Jugador)
borrarPokemones	: Pokemones $p \times$ conj(Nat) ids	→ Pokemones
unoAtrapa	: conj(Jugador) $js \times$ Juego pGo	→ conj(Jugador) $\{js \subseteq \text{jugadores}(pGo)\}$
posicionJugadores	: Coordenada $c \times$ Juego pGo	→ conj(Jugador) $\{c \in \text{posiciones}(\text{mapa}(pGo))\}$

axiomas $\forall m: \text{Mapa} \forall j, j1, j2: \text{Jugador} \forall c: \text{Coordenada} \forall pGo: \text{Juego} \forall p: \text{Pokemon} \forall n: \text{Nat} \forall t: \text{Tipo}$

$\text{mapa}(\text{nuevoJuego}(m)) \equiv m$
 $\text{jugadores}(\text{nuevoJuego}(m)) \equiv \emptyset$
 $\text{pokemones}(\text{nuevoJuego}(m)) \equiv \emptyset$
 $\text{mapa}(\text{agJugador}(j, c, pGo)) \equiv \text{mapa}(pGo)$
 $\text{jugadores}(\text{agJugador}(j, c, pGo)) \equiv \text{Ag}(j, \text{jugadores}(pGo))$
 $\text{posicionJugador}(j1, \text{agJugador}(j2, c, pGo)) \equiv \text{if } j1 = j2 \text{ then } c \text{ else } \text{posicionJugador}(j1, pGo) \text{ fi}$
 $\text{pokemones}(\text{agJugador}(j, c, pGo)) \equiv \text{pokemones}(pGo)$
 $\text{posicionPokemon}(p, \text{agJugador}(j, c, pGo)) \equiv \text{posicionPokemon}(p, pGo)$
 $\text{mapa}(\text{agPokemon}(n, t, c, pGo)) \equiv \text{mapa}(pGo)$
 $\text{jugadores}(\text{agPokemon}(n, t, c, pGo)) \equiv \text{jugadores}(pGo)$
 $\text{posicionJugador}(j, \text{agPokemon}(n, t, c, pGo)) \equiv \text{posicionJugador}(j, pGo)$
 $\text{pokemones}(\text{agPokemon}(n, t, c, pGo)) \equiv \text{definir}(n, t, \text{pokemones}(pGo))$
 $\text{posicionPokemon}(p, \text{agPokemon}(n, t, c, pGo)) \equiv \text{if } \Pi_1(p) = n \text{ then } c \text{ else } \text{posicionPokemon}(p, pGo) \text{ fi}$
 $\text{mapa}(\text{informarPosicion}(j, c, pGo)) \equiv \text{mapa}(pGo)$

```

jugadores(informarPosicion(j, c, pGo)) ≡ if nula?(c) then
    jugadores(pGo)
else
    actualizarContadores(if posicionInvalida(posicionJugador(j,
pGo), c, pGo) then
        if cantidadSanciones(j) = 4 then
            jugadores(pGo) - j
        else
            Ag(sancionar(j), jugadores(pGo) - j)
        fi
    else
        jugadores(pGo)
    fi, c, pGo)
fi
posicionJugador(j1, informarPosicion(j2, c, pGo)) ≡ if j1 ≠ j2 then
    posicionJugador(j1, pGo)
else
    if nula?(c) then
        fueraDelMapa
    else
        if nula?(posicionJugador(j2, pGo)) then
            c
        else
            if posicionInvalida(c) then
                if cantidadDeSanciones(j1) = 4 then
                    fueraDelMapa
                else
                    c
                fi
            else
                c
            fi
        fi
    fi
fi
pokemones(informarPosicion(j, c, pGo)) ≡ if nula?(c) then
    pokemones(pGo)
else
    if (posicionInvalida(posicionJugador(j, pGo), c, pGo) ∧ canti-
dadSanciones(j) = 4) then
        borrarPokemones(pokemones(pGo), claves(pokedex(j)))
    else
        pokemones(pGo)
    fi
fi
borrarPokemones(p, ids ≡ if vacio?(ids) then
    p
else
    borrarPokemones(borrar(dameUno(ids), p), sinUno(ids))
fi

```

```

actualizarContadores(cj, c, pGo)  $\equiv$  if  $\emptyset?(cj)$  then
     $\emptyset$ 
else
    if posicionJugador(dameUno(cj),pGo) == c then
        Ag(actualizarEspera(true,dameUno(cj)),      actualizarContado-
            res(sinUno(cj),c,pGo))
    else
        if esperaParaAtrapar(dameUno(cj)) == 9 then
            unoAtrapa(posicion.Jugadores(posicionJugador(dameUno(cj),pGo),pGo)
             $\cup$       actualizarContadores(cj      -      posicion.Jugado-
            res(posicionJugador(dameUno(cj),pGo),c,pGo)
        else
            Ag(actualizarEspera(false,dameUno(cj)),      actualizarContado-
            res(sinUno(cj),c,pGo))
        fi
    fi
fi

```

Fin TAD

4. TAD Mapa

TAD MAPA

géneros mapa

igualdad observacional

$$(\forall m, m' : \text{Mapa}) \left(m =_{\text{obs}} m' \iff \left(\begin{array}{l} (\text{posiciones}(m) =_{\text{obs}} \text{posiciones}(m')) \wedge \\ (\forall c1, c2 : \text{Coordenada}) \\ (\text{existeCamino}(c1, c2, m) \leftrightarrow \text{existeCamino}(c1, c2, m')) \end{array} \right) \right)$$

exporta Mapa, generadores, observadores

usa BOOL, COORDENADA, CONJ()

observadores básicos

posiciones : Mapa $m \longrightarrow \text{Conj}(\text{Coordenada})$

existeCamino : Coordenada $c1 \times \text{Coordenada } c2 \times \text{Mapa } m \longrightarrow \text{bool}$

generadores

crear : $\longrightarrow \text{Mapa}$

agCoordenada : Coordenada $c \times \text{Conj}(\text{Coordenada}) \text{ } cs \times \text{Mapa } m \longrightarrow \text{Mapa}$
 $\{cs \subseteq \text{posiciones}(m) \wedge c \notin \text{posiciones}(m)\}$

otras operaciones

existenCamino : Coordenada $c1 \times \text{Conj}(\text{Coordenada}) \text{ } cs \times \text{Mapa } m \longrightarrow \text{bool}$
 $\{\text{Ag}(c1, cs) \subseteq \text{posiciones}(m)\}$

axiomas $\forall c, c1, c2 : \text{Coordenada } \forall cs : \text{conj}(\text{Coordenada}) \forall m : \text{Mapa}$

posiciones(crear()) $\equiv \emptyset$

posiciones(agCoordenada(c, cs, m)) $\equiv \text{Ag}(c, \text{posiciones}(m))$

existeCamino(c1, c2, crear()) $\equiv \text{false}$

existeCamino(c1, c2, agCoordenada(c, cs, m)) \equiv **if** $c1 \notin \text{posiciones}(m)$ **then**
 if $c2 \notin \text{posiciones}(m)$ **then**
 false
 else
 if $c1 == c2$ **then**
 if $c2 \in cs$ **then**
 true
 else
 existenCamino(c2, cs, m)
 fi
 else
 false
 fi
 fi
else
 if $c2 \in \text{posiciones}(m)$ **then**
 if $c2 == c$ **then**
 if $c1 \in cs$ **then**
 true
 else
 existenCamino(c1, cs, m)
 fi
 else
 false
 fi
 else
 existeCamino(c1, c2, m)
 fi
fi

```

existenCaminos(c, cs, m)  $\equiv$  if vacio?(cs) then
    false
else
    existeCamino(c, dameUno(cs), m)  $\vee$  existenCaminos(c, sinUno(cs), m)
fi

```

Fin TAD

5. TAD Jugador

TAD JUGADOR

igualdad observacional

$$(\forall j, j' : \text{Jugador}) \left(j =_{\text{obs}} j' \iff \left(\begin{array}{l} (\text{id}(j) =_{\text{obs}} \text{id}(j')) \wedge \\ (\text{cantidadSanciones}(j) =_{\text{obs}} \text{cantidadSanciones}(j')) \wedge \\ (\text{pokedex}(j) =_{\text{obs}} \text{pokedex}(j')) \wedge \\ (\text{esperaParaAtrapar}(j) =_{\text{obs}} \text{esperaParaAtrapar}(j')) \end{array} \right) \right)$$

géneros jugador

exporta Jugador, generadores, observadores

usa BOOL, NAT, POKEMONES

observadores básicos

id	: Jugador	→ Nat
cantidadSanciones	: Jugador	→ Nat
pokedex	: Jugador	→ Conj(Nat)
esperaParaAtrapar	: Jugador	→ Nat

generadores

crearJugador	: Nat	→ Jugador	
atraparPokemon	: Nat $p \times$ Jugador j	→ Jugador	$\{p \notin \text{pokedex}(j)\}$
sancionar	: Jugador	→ Jugador	
actualizarEspera	: Bool \times Jugador	→ Jugador	

axiomas $\forall id, p: \text{Nat} \forall j: \text{Jugador} \forall reset: \text{Bool}$

$\text{id}(\text{crearJugador}(id)) \equiv \text{id}$
 $\text{cantidadSanciones}(\text{crearJugador}(id)) \equiv 0$
 $\text{pokedex}(\text{crearJugador}(id)) \equiv \emptyset$
 $\text{esperaParaAtrapar}(\text{crearJugador}(id)) \equiv 0$
 $\text{id}(\text{atraparPokemon}(p, j)) \equiv \text{id}(j)$
 $\text{cantidadSanciones}(\text{atraparPokemon}(p, j)) \equiv \text{cantidadSanciones}(j)$
 $\text{pokedex}(\text{atraparPokemon}(p, j)) \equiv \text{Ag}(p, \text{pokedex}(j))$
 $\text{esperaParaAtrapar}(\text{atraparPokemon}(p, j)) \equiv 0$
 $\text{id}(\text{sancionar}(j)) \equiv \text{id}(j)$
 $\text{cantidadSanciones}(\text{sancionar}(j)) \equiv \text{cantidadSanciones}(j) + 1$
 $\text{pokedex}(\text{sancionar}(j)) \equiv \text{pokedex}(j)$
 $\text{esperaParaAtrapar}(\text{sancionar}(j)) \equiv \text{esperaParaAtrapar}(j)$
 $\text{id}(\text{actualizarEspera}(\text{reset}, j)) \equiv \text{id}(j)$
 $\text{cantidadSanciones}(\text{actualizarEspera}(\text{reset}, j)) \equiv \text{cantidadSanciones}(j)$
 $\text{pokedex}(\text{actualizarEspera}(\text{reset}, j)) \equiv \text{pokedex}(j)$
 $\text{esperaParaAtrapar}(\text{actualizarEspera}(\text{reset}, j)) \equiv \text{if reset then } 0 \text{ else } \text{esperaParaAtrapar}(j) + 1 \text{ fi}$

Fin TAD

6. TAD Coordenada

TAD COORDENADA

géneros coordenada

observadores básicos

x : Coordenada c \longrightarrow Nat
 y : Coordenada c \longrightarrow Nat
 $nula?$: Coordenada c \longrightarrow Bool

$\{\neg nula?(c)\}$
 $\{\neg nula?(c)\}$

generadores

$fueraDelMapa$: \longrightarrow Coordenada
 $nueva$: Nat $x \times$ Nat y \longrightarrow Coordenada

axiomas $\forall x1, y1: \text{Nat}$

$nula?(fueraDelMapa()) \equiv \text{true}$
 $nula?(nueva(x1, y1)) \equiv \text{false}$
 $x(nueva(x1, y1)) \equiv x1$
 $y(nueva(x1, y1)) \equiv y1$

Fin TAD