DEPARTAMENTO
DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

TP de Especificación

Juego de la vida toroidal

3 de octubre de 2018 — Algoritmos y Estructuras de Datos I

Grupo: Java the Hutt;

Integrante	LU	Correo electrónico
Pomsztein, Vladimir	364/18	blastervla@gmail.com
Zinik, Luciano	290/17	lzinik@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
http://www.exactas.uba.ar

1. Problemas

1.1. esValido

proc esValido (in t: *toroide*, out result: Bool) {
    Pre {true}
    Post {result = true$\leftrightarrow$ $esToroideValido(t)$}
}

1.2. posicionesVivas

proc posicionesVivas (in t: *toroide*, out vivas: $seq\langle\mathbb{Z}\times\mathbb{Z}\rangle$) {
    Pre {$esToroideValido(t)$}
    Post {$(\forall i,j:\mathbb{Z})(enRango(t,i,j)\longrightarrow_L((i,j)\in vivas\leftrightarrow estaViva(t[i][j])))$}
}

1.3. densidadPoblacion

proc densidadPoblacion (in t: *toroide*, out result: $\mathbb{R}$) {
    Pre {$esToroideValido(t)$}
    Post {$result = cantidadVivas(t)/cantidadTotal(t)$}
    aux cantidadTotal (t: *toroide*) : $\mathbb{Z} = filas(t)\times columnas(t)$;
}

1.4. evolucionDePosicion

proc evolucionDePosicion (in t: *toroide*, in posicion: $\mathbb{Z}\times\mathbb{Z}$, out result: Bool) {
    Pre {$esToroideValido(t)\wedge_L enRango(t,posicion_0,posicion_1)$}
    Post {$result = valorLuegoDeEvolucion(t,posicion)$}
}

1.5. evolucionToroide

proc evolucionToroide (inout t: *toroide*) {
    Pre {$esToroideValido(t)\wedge_L t = T0$}
    Post {$esEvolucion(T0,t)$}
}

1.6. evolucionMultiple

proc evolucionMultiple (in t: *toroide*, in k: $\mathbb{Z}$, out result: *toroide*) {
    Pre {$esToroideValido(t)\wedge k\geq 0$}
    Post {$esEvolucionMultiple(t,result,k)$}
}

1.7. esPeriodico

```
proc esPeriodico (in t: toroide, inout p: ℤ, out result: Bool) {
        Pre {esToroideValido(t)}
        Post {result = true ↔ (∃k : ℤ)(esEvolucionMultiple(t, t, k)
                ∧_L esElMenorKAntesDeEvolucion(t, t, k)) ∧_L p = k}
}
```

1.8. primosLejanos

```
proc primosLejanos (in t1: toroide, in t2: toroide, out primos: Bool) {
        Pre {esToroideValido(t1) ∧ esToroideValido(t2) ∧_L mismaDimension(t1, t2)}
        Post {primos = true ↔ ((∃k : ℤ)((k > 0)
                ∧_L ((esEvolucionMultiple(t1, t2, k)) ∨_L (esEvolucionMultiple(t2, t1, k)))))}
}
```

1.9. seleccionNatural

```
proc seleccionNatural (in ts: seq⟨toroide⟩, out res: ℤ) {
        Pre {length(ts) > 0 ∧_L (todosToroidesValidos(ts) ∧_L algunToroideMuere(ts))}
        Post {(0 ≤ res < length(ts)) ∧_L tieneElMayorTiempoDeMuerte(ts[res], ts)}
        pred todosToroidesValidos (ts: seq⟨toroide⟩) {
              (∀t :toroide)((t ∈ ts) ⟶_L esToroideValido(t))
        }
        pred algunToroideMuere (ts: seq⟨toroide⟩) {
              (∃t :toroide)((t ∈ ts) ∧_L (muere(t)))
        }
        pred tieneElMayorTiempoDeMuerte (t: toroide, ts: seq⟨toroide⟩) {
              (∀tx : toroide)((tx ∈ ts)
                ⟶_L (cantidadDeTicksHastaMuerte(t) ≥ cantidadDeTicksHastaMuerte(tx)))
        }
        aux cantidadDeTicksHastaMuerte (t: toroide) : ℤ =
              if (∃tx : toroide)(mismaDimension(t, tx) ∧_L cantidadVivas(tx) = 0)
              ∧_L(∃k : ℤ)(k > 0∧_L esEvolucionMultiple(t, tx, k)∧_L esElMenorKAntesDeEvolucion(t, tx, k))
              then k else − 1 fi ;
        pred muere (t: toroide) {
              cantidadDeTicksHastaMuerte(t) ≠ −1
        }
}
```

1.10. fusionar

```
proc fusionar (in t1: toroide, in t2: toroide, out res: toroide) {
        Pre {(esToroideValido(t1) ∧ esToroideValido(t2)) ∧_L mismaDimension(t1, t2)}
        Post {mismaDimension(res, t1) ∧_L
              (∀i, j : ℤ)((enRango(res, i, j)) ⟶_L (estaViva(res, i, j)
              ↔ estaViva(t1, i, j) ∧ estaViva(t2, i, j)))}
}
```

1.11. vistaTrasladada

proc vistaTrasladada (in t1: *toroide*, in t2: *toroide*, out res: Bool) {
  Pre {$(esToroideValido(t1) \wedge esToroideValido(t2)) \wedge_L mismaDimension(t1, t2)$}
  Post {$res = \text{true} \leftrightarrow esVistaTrasladada(t1, t2)$}
}

1.12. enCrecimiento

proc enCrecimiento (in t: *toroide*, out res: Bool) {
  Pre {$esToroideValido(t)$}
  Post {$res = \text{true} \leftrightarrow (\exists te : toroide)(esEvolucion(t, te) \wedge_L crecioSuperficie(t, te))$}
  pred crecioSuperficie (t: *toroide*, te: *toroide*) {
    $(\exists s, se : \mathbb{Z})((esMenorSuperficieDeToroide(t, s) \wedge esMenorSuperficieDeToroide(te, se))$
    $\wedge_L s < se)$
  }
  pred esMenorSuperficieDeToroide (t: *toroide*, s: $\mathbb{Z}$) {
    $(\exists x : toroide)((esVistaTrasladada(t, x) \wedge_L tieneLaMenorSuperficieDelToroide(x)))$
  }
  pred tieneLaMenorSuperficieDelToroide (t: *toroide*) {
    $(\forall x : toroide)(((x \neq t) \wedge esVistaTrasladada(t, x)) \longrightarrow_L esSuperficieMayorOIgual(x, t))$
  }
  pred esSuperficieMayorOIgual (t1: *toroide*, t2: *toroide*) {
    $(\exists s1, s2 : \mathbb{Z})((esSuperficie(t1, s1) \wedge esSuperficie(t2, s2)) \wedge_L s1 \geq s2)$
  }
  pred esSuperficie (t: *toroide*, s: $\mathbb{Z}$) {
    $(\exists rect : (\mathbb{Z} \times \mathbb{Z}) \times (\mathbb{Z} \times \mathbb{Z}))(esRectanguloDeToroide(rect, t) \wedge_L mantieneCantidadVivas(rect, t)$
    $\wedge_L esElMenorRectangulo(rect, t) \wedge_L area(rect) = s)$
  }
  pred mantieneCantidadVivas (rect: $(\mathbb{Z} \times \mathbb{Z}) \times (\mathbb{Z} \times \mathbb{Z})$, t: *toroide*) {
    $/* rect = (xStart, yStart) \times (xEnd, yEnd) */$
    $contarVivasEnArea(t, rect) = cantidadVivas(t)$
  }
  pred esRectanguloDeToroide (rect: $(\mathbb{Z} \times \mathbb{Z}) \times (\mathbb{Z} \times \mathbb{Z})$, t: *toroide*) {
    $/* rect = (xStart, yStart) \times (xEnd, yEnd) */$
    $(0 \leq rect_{0_0} \leq rect_{1_0} \leq columnas(t)) \wedge (0 \leq rect_{0_1} \leq rect_{1_1} \leq filas(t))$
  }
  pred esElMenorRectangulo (r1: $(\mathbb{Z} \times \mathbb{Z}) \times (\mathbb{Z} \times \mathbb{Z})$, t: *toroide*) {
    $(\forall r2 : (\mathbb{Z} \times \mathbb{Z}) \times (\mathbb{Z} \times \mathbb{Z}))((r2 \neq r1)$
    $\longrightarrow_L (\neg mantieneCantidadVivas(r2, t) \vee_L area(r2) \geq area(r1)))$
  }
  $/* rect = (xStart, yStart) \times (xEnd, yEnd) */$
  aux contarVivasEnArea (t: *toroide*, rect: $(\mathbb{Z} \times \mathbb{Z}) \times (\mathbb{Z} \times \mathbb{Z})) : \mathbb{Z} =$
    $\sum_{i=rect_{0_1}}^{rect_{1_1}} (\sum_{j=rect_{0_0}}^{rect_{1_0}}$ if $estaViva(t[i][j])$ then 1 else 0 fi) ;
  aux area (rect: $(\mathbb{Z} \times \mathbb{Z}) \times (\mathbb{Z} \times \mathbb{Z})) : \mathbb{Z} =$
    $/* base \times altura */$
    $(rect_{1_0} - rect_{0_0}) \times (rect_{1_1} - rect_{0_1})$;
}

2. Predicados y Auxiliares generales

pred noEsVacia (t: *toroide*) {
(length(t) >0)$\wedge_L$($\forall x : seq\langle$Bool$\rangle$)(($x \in t$) $\longrightarrow_L$ (length(x) >0))
}
pred esMatriz (t: *toroide*) {
($\forall x, y : seq\langle$Bool$\rangle$)(($x, y \in t$) $\longrightarrow_L$ (length(x) = length(y)))
}
pred esToroideValido (t: *toroide*) {
($noEsVacia(t) \wedge esMatriz(t)$)
}
pred filas (t: *toroide*) {
length(t)
}
pred columnas (t: *toroide*) {
if filas(t) >0 then length(t[0]) else 0 fi
}
pred estaViva (x: Bool) {
x = true
}
pred enRango (t: *toroide*, i: $\mathbb{Z}$, j: $\mathbb{Z}$) {
($0 \leq i < filas(t)$) $\wedge_L$ ($0 \leq j < columnas(t)$)
}
aux cantidadVivas (t: *toroide*) : $\mathbb{Z}$ = $\sum_{i=0}^{filas(t)-1}$ ($\sum_{j=0}^{columnas(t)-1}$ if $estaViva(t[i][j])$ then 1 else 0 fi) ;
aux valorLuegoDeEvolucion (t: *toroide*, pos: $\mathbb{Z} \times \mathbb{Z}$) : Bool =
if $seMantieneViva(t, pos) \vee_L vivePorReproduccion(t, pos)$ then true else false fi ;
pred seMantieneViva (t: *toroide*, pos: $\mathbb{Z} \times \mathbb{Z}$) {
$estaViva(t[posicion_0][posicion_1]) \wedge_L 2 \leq vivasAdyacentes(t, posicion) \leq 3$
}
pred vivePorReproduccion (t: *toroide*, pos: $\mathbb{Z} \times \mathbb{Z}$) {
($\neg estaViva(t[posicion_0][posicion_1]) \wedge_L vivasAdyacentes(t, posicion) = 3$)
}
aux vivasAdyacentes (t: *toroide*, pos: $\mathbb{Z} \times \mathbb{Z}$) : $\mathbb{Z}$ =
($\sum_{i=-1}^{1} \sum_{j=-1}^{1}$ if $valorPosicionNormalizada(t, (pos_0 + i, pos_1 + j)) =$ true then 1 else 0 fi)
$-$ (if $estaViva(t, pos_0, pos_1)$ then 1 else 0 fi) ;
aux valorPosicionNormalizada (t: *toroide*, pos: $\mathbb{Z} \times \mathbb{Z}$) : Bool =
$t[normalizarIndice(filas(t), pos_0)][normalizarIndice(columnas(t), pos_1)]$;
aux normalizarIndice (limite: $\mathbb{Z}$, i: $\mathbb{Z}$) : $\mathbb{Z}$ = if $i < 0$ then $(i + limite)$ else
(if $i \geq limite$ then $(i - limite)$ else i fi) fi ;
pred mismaDimension (t1: *toroide*, t2: *toroide*) {
$filas(t1) = filas(t2) \wedge_L columnas(t1) = columnas(t2)$
}
pred esEvolucion (t: *toroide*, te: *toroide*) {
$mismaDimension(t, te) \wedge_L (\forall i, j : \mathbb{Z})(enRango(t, i, j) \longrightarrow_L (te[i][j] = valorLuegoDeEvolucion(t, (i, j))))$
}
pred esEvolucionMultiple (t: *toroide*, te: *toroide*, k: $\mathbb{Z}$) {
($\exists ts : seq\langle toroide\rangle$)($(length(ts) = k + 1) \wedge_L ts[0] = t \wedge_L ordenadaPorEvolucion(ts)$
$\wedge_L te = ts[k]$)
}

```
pred ordenadaPorEvolucion (ts: seq⟨toroide⟩) {
      (∀i : ℤ)((0 < i ≤ k) ⟶_L esEvolucion(ts[i − 1], ts[i]))
}
/* Indica si el toroide t llega a tener la forma de te por primera vez en el k-ésimo tick*/
pred esElMenorKAntesDeEvolucion (t: toroide, te: toroide, k: ℤ) {
      (∀i : ℤ)((0 < i < k) ⟶_L ¬esEvolucionMultiple(t, te, i))
}
pred esVistaTrasladada (t1: toroide, t2: toroide) {
      (∃i, j : ℤ)((∀x, y : ℤ)(enRango(t1, x, y)
       ⟶_L (t1[x][y] = valorPosicionNormalizada(t2, (x + i, y + j)))))
}
```

3. Decisiones tomadas

Intuimos que una posición tiene 8 adyacentes independientemente del tamaño del toroide, implicando esto que dentro de las adyacentes a una posición se pueden contar posiciones repetidas.