DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

TP de Especificación

Análisis Habitacional Argentino

26 de Agosto de 2019 — Algoritmos y Estructuras de Datos I

Grupo 25

Integrante	LU	Correo electrónico
Yulita, Federico	351/17	fyulita@dc.uba.ar
Chanes, Mauricio	226/19	mchanes@dc.uba.ar

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

proc esImagenValida (in A : imagen , out result : Bool) {
Pre {true}
Post $\{esValida(A) \land esBinaria(A) \leftrightarrow result = true\}$
}

pred esValida (A: imagen) $\{(\forall i : \mathbb{Z})(0 \leqslant i < filas(A)) \longrightarrow_L (|A[i]| > 0 \land (\forall j : \mathbb{Z})(0 \leqslant j < filas(A) \longrightarrow_L |A[i]| = |A[j]|))\}$

pred esBinaria (A: imagen) $\{(\forall s : seq\langle\mathbb{Z}\rangle)(s \in A \longrightarrow (\forall d : \mathbb{Z})(d \in s \longrightarrow esDato(d))\}$

proc sonPixelesConectados (in A : imagen, in $p_0 : pixel$, in $p_1 : pixel$, in $k : \mathbb{Z}$, out $result$: Bool){
Pre $\{esImagenValida(A) \land pixelEnImagen(p_0, A) \land pixelEnImagen(p_1, A) \land (k = 4 \lor k = 8)\}$
Post $\{result = \text{true} \leftrightarrow (\exists s : sqPixel)(secuenciaValida(s, A) \land secuenciaConectada(s, k, A) \land p_0 \in s \land p_1 \in s)\}$
}

pred secuenciaConectada (s : sqPixel, k : $\mathbb{Z}$, A : imagen) $\{sqPixelesActivados(s, A) \land (\forall i : \mathbb{Z})(1 \leqslant i < |s| - 1 \longrightarrow_L sonVecinos(s[i], s[i + 1], k))\}$

proc esFormaConvexa (in A : imagen, in k : $\mathbb{Z}$, out result : Bool) {
Pre $\{esImagenValida(A) \land (k = 4 \lor k = 8) \land regionImagen(A, k)\}$
Post $\{result = \text{true} \leftrightarrow (\forall p, q : pixel)((pixelActivado(p, A) \land pixelActivado(q, A)) \longrightarrow (\exists r : sqPixel)(esRectaDe(r, p, q, A) \land sqPixelesActivados(r, A)))\}$
}

pred esRectaDe (r : sqPixel, p : pixel, q : pixel, A : imagen) $\{secuenciaValida(r, A) \land (p \in r) \land (q \in r) \land (\forall z : pixel)(z \in r \longrightarrow_L (\exists x : \mathbb{R})((z[1] \leqslant x < z[1] + 1) \land (z[0] \leqslant pendiente(p, q) * x + ordenada(p, q) < z[0] + 1)))\}$

aux pendiente (p : pixel, q : pixel) : $\mathbb{R} = (p[0] - q[0])/(p[1] - q[1])$;

aux ordenada (p : pixel, q : pixel) : $\mathbb{R} = q[1]$;

proc devolverPromedioAreas (in A: imagen, in k: $\mathbb{Z}$, out prom: $\mathbb{R}$) {
Pre $\{esImagenValida(A) \land (k = 4 \lor k = 8)\}$
Post $\{(noHayRegion(A) \longrightarrow prom = 0) \lor (\neg noHayRegion(A) \longrightarrow prom = sumaPixelesActivados(A)/sumaDeRegiones(A, deImagenASecuencia(A), k)\}$
}

pred noHayRegion (A: Imagen, k: $\mathbb{Z}$) $\{\neg(\exists s : sqPixel)(esRegion(s, A, k))\}$

aux sumaPixelesActivados (A: imagen) : $\mathbb{R} = \sum_{i=0}^{|A|-1} \sum_{j=0}^{|A[i]|-1} A[i][j]$;

aux deImagenASecuencia (A: imagen) : $\text{sq}\langle Pixel\rangle = (\forall j : \mathbb{Z})(\forall i : \mathbb{Z})((0 \leqslant i < |A|)(0 \leqslant j < |A[i]|) \longrightarrow_L$ if $A[i][j] = 1$ then $\langle i, j\rangle$ else $\langle\rangle$ fi ;

aux sumaDeRegiones (A: imagen, S: $\text{sq}\langle Pixel\rangle, k : \mathbb{Z}$) : $\mathbb{Z} = (\forall p : Pixel)(p \in S) \longrightarrow_L \sum_{i=0}^{|S|-1}$ if $\neg sonVecinos(S[i], p, k)$ then 1 else 0 fi ;

proc calcularContorno (in A : imagen, in k : $\mathbb{Z}$, out edge : sqPixel) {
Pre $\{regionesValidas(A, k) \land regionImagen(A, k) \land (k = 4 \lor k = 8)\}$
Post $\{esBorde(edge, A, k) \land \neg(\exists l : sqPixel)((|l| > |edge|) \land esBorde(l, A, k))\}$
}

pred esBorde (s : sqPixel, A : imagen, k : $\mathbb{Z}$) $\{sqPixelesActivados(s, A) \land (\forall p : pixel)(p \in s \rightarrow (\exists q : pixel)(sonVecinos(p, q, k) \land \neg pixelActivado(q, A)))\}$

pred regionesValidas (A : imagen, k : $\mathbb{Z}$) $\{(\forall s : sqPixel)(esRegion(s, A, k) \longrightarrow |s| \geqslant 2)\}$

proc cerrarForma (in A : imagen, in b : imagen, out c : seq $\langle sqPixel \rangle$){
Pre {$esImagenValida(A) \land (|A| > 0) \land esElemEstruc(b)$}
Post {$(\forall i : \mathbb{Z})(0 \leqslant i < |c| - 1 \longrightarrow_L closing(c[i], c[i+1], b))$}
}

pred closing (si : sqPixel, so : sqPixel, b : imagen) {$(\exists sa : sqPixel)(dilatacion(si, sa, b) \land erosion(sa, so, b))$}

pred dilatacion (si : sqPixel, so : sqPixel, b : imagen) {$elemEstrucCentrado(so[|so| - 1], b) \land (\exists p : pixel)((p \in si) \land (pixelActivado(p, b)))$}

pred erosion (si : sqPixel, so : sqPixel, b : imagen) {$elemEstrucCentrado(so[|so| - 1], b) \land (\forall p : pixel)(pixelActivado(p, b) \longrightarrow p \in si)$}

proc obtenerRegionConectada (A: imagen, in semilla: pixel, out seq: seq$\langle Imagen \rangle$){
Pre {$esImagenValida(A) \ \land \ pixelActivado(semilla, A)$}
Post {$(\forall p : pixel)((\exists B : imagen)(esImagenValida(B) \land \neg esActiva(B) \ \land \ |B| = |A|))$
$(\neg sonVecinos(semilla, p, 8) \longrightarrow seq = \langle A, \ dePixelAImagen(B, semilla)) \lor ((\exists s : sq\langle Pixel \rangle)$
$(\forall i : \mathbb{Z})(0 \leqslant i < |s| - 1 \land_L sonVecinos(semilla, s[i], 8)) \longrightarrow$
$seq = \langle A, desqPixelAImagen(B, \langle semilla \rangle + + s)$}
}

aux dePixelAImagen (A: imagen, p: pixel) : Imagen $= setAt(A[p[0]], A[p[0]][p[1]], 1$;

aux desqPixelAImagen (A: imagen, s: sq $\langle Pixel \rangle$) : $imagen = (\forall i : \mathbb{Z})(0 \leqslant i < |s| - 1) \longrightarrow_L dePixelAImagen(A, s[i])$;

proc esMaximoDiscoEsqueleto (in A : imagen, in S : imagen, in D_z : $imagen$, out res : Bool){
Pre {$esImagenValida(A) \land esElemEstruc(D_z) \land esImagenValida(S) \land esqueletiza(A, S, D_z)$}
Post {$\neg(\exists D : imagen)(esqueletiza(A, S, D) \land esElemEstruc(D) \land (|D| > |D_z|))$}
}

pred esqueletiza (A : imagen, S : imagen, D_z : $imagen$){$esSubImagen(S, A) \land \neg(\exists p : pixel)(pixelEnImagen(p, S) \land elemEstrucCentrado(p, D_z) \land (\exists q : pixel)(pixelEnImagen(q, D_z) \land pixelEnImagen(q, A) \land \neg pixelEnImagen(q, S)))$}

pred esSubImagen (A : imagen, B : imagen) {$(\forall p : pixel)(pixelEnImagen(p, A) \longrightarrow pixelEnImagen(p, B))$}

pred interseccion (A : imagen, S: imagen, b : imagen) {$(\exists p : pixel)((\forall q : pixel)(pixelActivado(q, S) \longrightarrow pixelActivado(q, A) \land pixelActivado(q, elemEstrucCentrado(p, b))))$}

2. Predicados y Auxiliares generales

aux filas (m : A) : $\mathbb{Z} = |m|$;

aux columnas (m : A) : $\mathbb{Z}$ = if $filas(A) > 0$ then $|m[0]|$ else 0 fi ;

pred esDato (d : $\mathbb{Z}$) {$(d = 0) \lor (d = 1)$}

pred esPixel (p : seq$\langle \mathbb{Z} \rangle$){$|p| = 2 \land (p[0] \geqslant 0) \land (p[1] \geqslant 0)$}

pred pixelEnImagen (p : pixel, A : imagen) {$esPixel(p) \land_L (0 \leqslant p[0] < filas(A)) \land (0 \leqslant p[1] < columnas(A))$}

pred secuenciaEnImagen (s : sqPixel, A : imagen) {$(\forall p : pixel)(p \in s \longrightarrow pixelEnImagen(p, A))$}

pred secuenciaValida (s : sqPixel, A : imagen) {$esImagenValida(A) \land secuenciaEnImagen(s, A)$}

pred sqPixelesActivados (s : sqPixel, A : imagen) {$secuenciaEnImagen(s, A) \land (\forall p : pixel)$ $(p \in s \longrightarrow pixelActivado(p, A))$}

pred pixelActivado (p : pixel, A : imagen) {$pixelEnImagen(p, A) \land A[p[0]][p[1]] = 1$}

`pred sonVecinos` (p : pixel, q : pixel, k : $\mathbb{Z}$) $\{(esPixel(p) \wedge esPixel(q)) \wedge_L$
$(((k = 4) \wedge (|p[0] - q[0]| + |p[1] - q[1]| \leqslant 1)) \vee ((k = 8) \wedge (\max(|p[0] - q[0]|, |p[1] - q[1]|) \leqslant 1)))\}$

`aux max` (n : $\mathbb{Z}$, m : $\mathbb{Z}$) : $\mathbb{Z}$ $=$ if $n > m$ then n else m fi ;

`pred esRegion` (s: sqPixel, A: Imagen, k: $\mathbb{Z}$) $\{(\forall p, q : pixel)((p \in s \wedge q \in s) \longrightarrow sonPixelesConectados(A, p, q, k)) \wedge$
$\neg(\exists l : sqPixel)(esSubsecuencia(s, l) \wedge (\forall p, q : pixel)((p \in l \wedge q \in l) \longrightarrow sonPixelesConectados(A, p, q, k)))\}$

`pred regionImagen` (A : imagen, k : $\mathbb{Z}$) $\{(\exists! s : sqPixel)(esRegion(s, A, k))\}$

`pred esSubsecuencia` (s : sqPixel, l : sqPixel) $\{(|s| < |l|) \wedge (\forall p : pixel)(p \in s \longrightarrow p \in l)\}$

`pred esElemEstruc` (b : imagen) $\{esCuadradaImpar(b) \wedge esActiva(b)\}$

`pred esCuadradaImpar` (b : imagen) $\{(\forall s : seq\langle dato\rangle)(s \in b \longrightarrow (|b| = |s|)) \wedge (|b| \mod 2 = 1) \wedge (|b| > 0)\}$

`pred esActiva` (b : imagen) $\{(\forall p : pixel)(pixelEnImagen(p, b) \longrightarrow pixelActivado(p, b))\}$

`pred elemEstrucCentrado` (p : pixel, b : imagen) $\{(p[0] = (|b| - 1)/2) \wedge (p[1] = (|b| - 1)/2)\}$

3. Decisiones tomadas

- Para el predicado *secuenciaConectada* se tomó implícitamente que la secuencia va a considerarse conectada sólo si además de tener un conjunto de pixeles que están conectados en la secuencia están en orden. Esto lo hicimos ya que usamos la definición de trayectoria de secuencia del pdf.
- Tomamos que los únicos valores válidos de adyecencia k son $k = 4$ y $k = 8$ ya que fueron los únicos descriptos en el pdf.
- Al procedimiento 3 (*esFormaConvexa*) le agregamos el input k ya que dependiendo de qué forma de adyecencia querramos las rectas se definen de forma distinto y, por lo tanto, las regiones convexas se definen distinto.
- Para el predicado *esSubsecuencia* consideramos que una secuencia no es una subsecuencia de sí misma por motivos prácticos.
- Para los predicados dilatación, erosión y closing se tomó que los pixeles por los que se va trasladando el elemento estructurante están en orden.