



**DEPARTAMENTO
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico Número 2

Algoritmos y Estructuras de Datos I

Grupo: 07

Integrante	LU	Correo electrónico
Demartino, Francisco	348/14	demartino.francisco@gmail.com
Frachtenberg Goldsmit, Kevin	247/14	kevinfra94@gmail.com
Gomez, Horacio	756/13	horaciogomez.1993@gmail.com
Pondal, Iván	078/14	ivan.pondal@gmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

1. Especificación

Ejercicio 1. problema posicionesMasOscuras($i : Imagen$) = $res : [(\mathbb{Z}, \mathbb{Z})]\{$
 asegura : $mismos(res, [(x, y) \mid y \leftarrow [0..alto(i)], x \leftarrow [0..ancho(i)],$
 $sumaCanalesPixel(color(i, x, y)) == menorSumaCanales(i)]);$
 $\}$

Ejercicio 2. problema top10($g : Galeria$) = $result : [Imagen]\{$
 asegura : $(|imagenes(g)| < 10) \rightarrow$
 $result == listaImagenesOrdenadasVotos(g, imagenes(g));$
 asegura : $(|imagenes(g)| \geq 10) \rightarrow$
 $result == listaImagenesOrdenadasVotos(g, imagenes(g))[0..10];$
 $\}$

Ejercicio 3. problema laMasChiquitaConPuntoBlanco($g : Galeria$) = $result : Imagen\{$
 requiere : $|listaImagenesConPixelBlanco(imagenes(g))| > 0;$
 asegura : $result == listaImagenesOrdenadasTamaño(listaImagenesConPixelBlanco(imagenes(g)))[0];$
 $\}$

Ejercicio 4. problema agregarImagen($g : Galeria, i : Imagen$) $\{$
 modifica $g;$
 asegura lasDeAntesEstanConSusVotos: $(\forall j \leftarrow imagenes(pre(g)))$
 $j \in imagenes(g) \wedge votos(g, j) == votos(pre(g), j);$
 asegura lasQueEstanSalvoLaQueAgregoVienenDeAntes: $(\forall j \leftarrow imagenes(g), j \neq i)$
 $j \in imagenes(pre(g));$
 asegura siAgregoNuevaEntraConCeroVotos: $i \notin imagenes(pre(g)) \rightarrow$
 $(i \in imagenes(g) \wedge votos(g, i) == 0);$
 $\}$

Ejercicio 5. problema votar($g : Galeria, i : Imagen$) $\{$
 requiere noSeVotaCualquierCosa: $i \in imagenes(pre(g));$
 modifica $g;$
 asegura noCambianLasImagenes: $mismos(imagenes(g), imagenes(pre(g)));$
 asegura noSeTocanLosVotosDeLosOtros: $(\forall m \leftarrow imagenes(g), m \neq i) votos(g, m) ==$
 $votos(pre(g), m);$
 asegura elQueSeVotaSumaUno: $votos(g, i) == votos(pre(g), i) + 1;$
 $\}$

Ejercicio 6. problema eliminarMasVotada($g : Galeria$) $\{$
 requiere noVacía: $|imagenes(pre(g))| > 0;$
 modifica $g;$
 asegura seVaUnaSola: $|imagenes(g)| == |imagenes(pre(g))| - 1;$
 asegura siSeVaEsGrosa: $(\forall i \leftarrow imagenes(pre(g)), i \notin imagenes(g))$
 $votos(g, i) == maximo(todosLosVotos(g));$
 asegura lasQueEstanVienenDeAntesConSusVotos: $(\forall i \leftarrow imagenes(g))$
 $i \in imagenes(pre(g)) \wedge votos(g, i) == votos(pre(g), i);$
 $\}$

1.1. Auxiliares

- **aux cuenta**($x : T, a : [T]$) : $\mathbb{Z} = |[1 \mid y \leftarrow a, y == x]|$;
- **aux mismos**($a, b : [T]$) : $Bool = |a| == |b| \wedge (\forall c \leftarrow a) cuenta(c, a) == cuenta(c, b)$;
- **aux minimo**($l : [\mathbb{Z}]$) : $\mathbb{Z} = [x \mid x \leftarrow l, (\forall y \leftarrow l) x \leq y][0]$;
- **aux maximo**($l : [\mathbb{Z}]$) : $\mathbb{Z} = [x \mid x \leftarrow l, (\forall y \leftarrow l) x \geq y][0]$;
- **aux sumaCanalesPixel**($p : Pixel$) : $\mathbb{Z} = red(p) + green(p) + blue(p)$;
- **aux listaSumaCanalesPixeles**($i : Imagen$) : $[\mathbb{Z}] = [sumaCanalesPixel(color(i, x, y)) \mid y \leftarrow [0..alto(i)], x \leftarrow [0..ancho(i)]]$;
- **aux menorSumaCanales**($i : Imagen$) : $\mathbb{Z} = minimo(listaSumaCanalesPixeles(i))$;
- **aux todosLosVotos**($g : Galeria$) : $[\mathbb{Z}] = [votos(g, i) \mid i \leftarrow imagenes(g)]$;
- **aux cantPixeles**($i : Imagen$) : $\mathbb{Z} = ancho(i) * alto(i)$;
- **aux esPixelBlanco**($px : Pixel$) : $Bool = red(px) == green(px) == blue(px) == 255$;
- **aux tienePixelBlanco**($i : Imagen$) : $Bool = alguno([esPixelBlanco(color(i, x, y)) \mid y \leftarrow [0..alto(i)], x \leftarrow [0..ancho(i)]]]$;
- **aux listaImagenesConPixelBlanco**($imgs : [Imagen]$) : $[Imagen] = [im \mid im \leftarrow imgs, tienePixelBlanco(im)]$;
- **aux cuentaMenosPixeles**($imgs : [Imagen], img : Imagen$) : $\mathbb{Z} = |[1 \mid im \leftarrow imgs, cantPixeles(im) < cantPixeles(img)]|$;
- **aux cuentaMasVotos**($g : Galeria, imgs : [Imagen], img : Imagen$) : $\mathbb{Z} = |[1 \mid im \leftarrow imgs, votos(g, im) > votos(g, img)]|$;
- **aux listaImagenesOrdenadasVotos**($g : Galeria, imgs : [Imagen]$) : $[Imagen] = [im \mid i \leftarrow [0..|imgs|], im \leftarrow imgs, cuentaMasVotos(g, imgs, im) == i]$;
- **aux listaImagenesOrdenadasTamaño**($imgs : [Imagen]$) : $[Imagen] = [im \mid i \leftarrow [0..|imgs|], im \leftarrow imgs, cuentaMenosPixeles(imgs, im) == i]$;

2. abs e invRep de Galería