DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

TP de Especificación

26 de agosto de 2020 — Algoritmos y Estructuras de Datos I

Grupo: Nos cambiaron el nombre 2;

Integrante	LU	Correo electrónico
Castro Russo, Matias Nahuel	203/19	castronahuel14@gmail.com
Torsello, Juan Manuel	248/19	juantorsello@gmail.com
Capelo, Gianluca	83/19	gianluca.capelo@gmail.com
Yazlle, Máximo	310/19	myazlle99@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. Tipos

type audio: $seq\langle\mathbb{Z}\rangle$
type tupla: $\mathbb{Z}$ x $\mathbb{Z}$

2. Problemas

Ejercicio 1:

`proc formatoValido` (in s : $seq\langle\mathbb{Z}\rangle$, in c : $\mathbb{Z}$, in p : $\mathbb{Z}$, out result : Bool) {
`Pre` $\{|s| > 0, p > 0, c > 0\}$
`Post` $\{tieneProfundidadP(s,p) \wedge_L$ respetaFormatoCCanales(s,c)$\}$
}

Ejercicio 2:

`proc replicar` (in a : audio, in c : $\mathbb{Z}$, in p : $\mathbb{Z}$, out result : audio) {
`Pre` $\{esFormatoValido(a,1,p)\}$
`Post` $\{esFormatoValido(result,c,p) \wedge |result| =$ c*$|a| \wedge$ esRéplica(a,result,c) $\}$
}

Ejercicio 3:

`proc revertirAudio` (in a : audio, in c : $\mathbb{Z}$, in p : $\mathbb{Z}$, out invertido : audio) {
`Pre` $\{esFormatoValido(a,c,p)\}$
`Post` $\{|s| = |invertido| \wedge_L$ esFormatoValido(invertido,c,p) $\wedge_L$ inversosPorCanal(a,invertido,c)$\}$
}

Ejercicio 4:

`proc magnitudAbsolutaMáxima` (in a : audio, in c : $\mathbb{Z}$, in p : $\mathbb{Z}$, out máximos : $seq\langle\mathbb{Z}\rangle$, out posicionesMaximo : $seq\langle\mathbb{Z}\rangle$) {
`Pre` $\{esFormatoValido(a,c,p)\}$
`Post` $\{|posicionesMaximo| = |maximos| \wedge |maximos| = c \wedge$ lasPosicionesyElementosCorresponden(a,maximos,posicionesMaximo,c) $\wedge$ realmenteSonMaximos(a,maximos,c) $\}$
}

Ejercicio 5:

`proc redirigir` (in a : audio, in c : $\mathbb{Z}$, in p : $\mathbb{Z}$, out invertido : audio) {
`Pre` $\{(c = 1 \vee c = 2) \wedge$ esFormatoVálido(a,2,p) $\}$
`Post` $\{|a| = |invertido| \wedge_L$ esFormatoVálido(invertido,2,p) $\wedge_L (\forall i : \mathbb{Z}\ )((0 \leq i < \frac{|a|}{2} \wedge_L$ $comparaciónConSecuenciaLimpia(subseq(a, i*2, 2*(i+1)),$ $subseq(invertido, i*2, 2*(i+1)), c)\}$
}

Ejercicio 6:

proc bajarCalidad (inout a : $seq\langle audio\rangle$, in p : $\mathbb{Z}$, in p2 : $\mathbb{Z}$) {
Pre $\{a = A_0 \wedge |A_0| > 0 \wedge 1 \leq p2 < p \wedge$ todosTienenFormatoValidos(A_0, p)$\}$
Post $\{|a| = |A_0| \wedge$ todosTienenFormatoValidos(a,p2)
$\wedge$ todosTieneLongitudCorrespondiente(a,A_0)
$\wedge$ todosSonLosAudiosDeMenorCalidadEsperados($A_0, a, p, p2$)$\}$
}

Ejercicio 7:

proc audioSoftYHard (in sa : $seq\langle audio\rangle$, in p : $\mathbb{Z}$, in long : $\mathbb{Z}$, in umbral : $\mathbb{Z}$, out soft : $seq\langle audio\rangle$, out hard : $seq\langle audio\rangle$) {
Pre $\{long > 0 \wedge$ todosTienenFormatoValidos(sa,p) $\}$
Post $\{|sa| = |soft| + |hard| \wedge_L$ todosTienenFormatoValido(soft,p) $\wedge$
todosTienenFormatoValido(hard,p) $\wedge_L$ softYHard(sa,long,umbral,soft,hard)$\}$

}

Ejercicio 8:

proc remplazarSubAudio (in p : $\mathbb{Z}$, inout a : $seq\langle audio\rangle$, in a_1 : $seq\langle audio\rangle$, in a_2 : $seq\langle audio\rangle$) {
Pre $\{$a = $A_0 \wedge$ todosTienenFormatoValidos($\langle A_0, a_1, a_2\rangle$),p)
$\wedge$ apareceALoSumoUnaVezEn(a_1, A_0)$\}$
Post $\{(|a| = |A_0| - |a_1| + |a_2| \wedge$ aEsConcatTripleConA$_2$*DondeEstabaA*$_1$(a,A_0, a_1, a_2)) $\vee$
(a = $A_0 \wedge$ noAparece(a_1, A_0)$\}$
}

Ejercicio 9:

proc máximosTemporales (in a : audio, in tiempos : $seq\langle\mathbb{Z}\rangle$, out máximos $seq\langle\mathbb{R}\rangle$, out intervalos: $seq\langle\mathbb{Z}x\mathbb{Z}\rangle$) {
Pre $\{|a| > 0, |tiempos| > 0\}$
Post $\{$*máximoDeIntervalosOrdenado*($intervalos, a, máximos$) $\wedge$ intervalosVálidos(intervalos,tiempos)$\}$
}

Ejercicio 10:

proc limpiarAudio (inout a : audio, out atípicos: $seq\langle\mathbb{Z}\rangle$) {
Pre $\{$a = $A_0 \wedge |A_0| \geq 20\}$
Post $\{|A_0| = |a| \wedge$ MismoAudioSinOutliers(a,A_0) $\wedge$ AtípicosContieneLosÍndicesdelosOutliers(a,A_0)$\}$
}

2.1. Predicados y Auxiliares generales

pred tieneProfundidadP (s: $seq\langle\mathbb{Z}\rangle$, p : $\mathbb{Z}$) {
$(\forall i : \mathbb{Z}\ )(\ i \in s \longrightarrow$ (i $\geq -2^{p-1}$) $\wedge$ (i $\leq 2^{p-1} - 1$) $)\}$

`pred respetaFormatoCCanales` (s: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$|s|\ mod\ c = 0$}

`pred esFormatoValido` (s: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$, p : $\mathbb{Z}$) {
$|s| > 0 \wedge p > 0 \wedge$ c $> 0 \wedge$ tieneProfundidadP(s,p) $\wedge_L$ respetaFormatoCCanales(s,c) }

2)
`pred esRéplica` (a: audio, res: audio, c: $\mathbb{Z}$) {
$(\forall i : \mathbb{Z})((0 \leq i < |a| \longrightarrow_L$ todosIguales(a[i], subseq(res,i*c,c*(i+1));}

`pred todosIguales` (b: $\mathbb{Z}$, s: audio) {
$(\forall i : \mathbb{Z})((0 \leq i < |s| \longrightarrow_L$ s[i] = b;}

fin 2)

3)
`pred inversosPorCanal` (S : $seq\langle\mathbb{Z}\rangle$, T: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$(\forall canal : \mathbb{Z})(\forall i : \mathbb{Z})(0 < canal \leq$ c $\wedge\ 0 \leq i < \frac{|S|}{c} \longrightarrow_L$
S [i-esimoPorCanal(c, canal, i)] = T [i-esimoInverso(a, c, canal, i)]) }

aux i-esimoPorCanal (c : $\mathbb{Z}$, canal : $\mathbb{Z}$, i : $\mathbb{Z}$) : $\mathbb{Z}= (canal - 1) + (c * i)$

aux i-esimoInvertido (a: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$, canal : $\mathbb{Z}$, i : $\mathbb{Z}$) : $\mathbb{Z}= |a| - 1 - c + canal - (c * i)$

fin3)

4)
`pred lasPosicionsyElementosCorresponden` (a: $seq\langle audio\rangle$, maximos: $seq\langle\mathbb{Z}\rangle$,posicionesMaximo: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$lasPosicionesEstanEnRango(a, posicionesMaximas)$
$\wedge$ lasPosicionesCorrespondenAlElemento(a,maximos,posicionesMaximo,c)
$\wedge$ lasPosicionesCorrespondenAlCanalCorrecto(a,posicionesMaximo,c)) }

`pred lasPosicionesEstanEnRango` (a: $seq\langle audio\rangle$, posiciones: $seq\langle\mathbb{Z}\rangle$) {
$(\forall i : \mathbb{Z}\ )(\ 0 \leq i < |posiciones| \longrightarrow_L 0 \leq posiciones[i] < |a|)$ }

`pred lasPosicionesCorrespondenAlElemento` (a: $seq\langle audio\rangle$, maximos: $seq\langle\mathbb{Z}\rangle$, posicionesMaximo: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$(\forall i : \mathbb{Z}\ )(\ 0 \leq i < |posicionesMaximo| \longrightarrow_L$ a[posicionesMaximo[i]] = maximos[i]) }

`pred lasPosicionesCorrespondenAlCanalCorrecto` (a: $seq\langle audio\rangle$, posicionesMaximo: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$(\forall canal : \mathbb{Z}\ )(\ 0 < canal \leq c \longrightarrow$ posicionCorrespondeAlCanal(canal,a,posicionesMaximo,c))}

`pred posicionCorrespondeAlCanal` (canal: $\mathbb{Z}$, a: $seq\langle audio\rangle$, posicionesMaximo: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$(\exists i : \mathbb{Z}\ )(\ 0 \leq i < \frac{|a|}{c} \wedge_L$ posicionesMaximo[canal-1]= canal-1 + c*i)}

`pred realmenteSonMaximos` (a: $seq\langle audio\rangle$, maximos: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$(\forall canal : \mathbb{Z}\ )(\ 0 < canal \leq c \longrightarrow$ esMaximoEnCanal(canal,a,maximos,c))}

`pred esMaximoEnCanal` (canal : $\mathbb{Z}$,a: $seq\langle audio\rangle$, maximos: $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$(\forall j : \mathbb{Z}\ )(\ 0 \leq j < \frac{|a|}{c} \longrightarrow_L$ abs(maximos[canal-1]) $\geq abs(\ a[canal - 1 + c * j]\ ))$}

fin4)

5)
`pred comparaciónConSecuenciaLimpia` (s : $seq\langle\mathbb{Z}\rangle$, t : $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$|s| = 2 \wedge_L\ (c = 1 \wedge s[0] = s[1] \longrightarrow setAt(s, 1, 0) = t) \wedge_L$
$(c = 2 \wedge s[0] = s[1] \longrightarrow setAt(s, 0, 0) = t) \wedge_L\ (s[0] \neq s[1] \longrightarrow s = t)\ \}$
`pred comparaciónConSecuenciaLimpia` (s : $seq\langle\mathbb{Z}\rangle$, t : $seq\langle\mathbb{Z}\rangle$, c : $\mathbb{Z}$) {
$|s| = 2 \wedge_L\ (c = 1 \longrightarrow setAt(s, 1, s[1] - s[0]) = t) \wedge_L$
$(c = 2 \longrightarrow setAt(s, 0, s[0] - s[1]) = t)\ \}$

fin 5)
6)
`pred todosTienenFormatoValidos` (a: $seq\langle audio\rangle$, p : $\mathbb{Z}$) {
$(\forall i : audio\ )(\ i \in a \longrightarrow$ esformatoValido(i,1,p)) }

`pred todosTieneLongitudCorrespondiente` (a: $seq\langle audio\rangle$, A_0: $seq\langle audio\rangle$) {
$(\forall i : \mathbb{Z})(0 \leq i < |a| \longrightarrow_L |a[i]| = |A_0[i]|)\ \}$

`pred todosSonLosAudiosDeMenorCalidadEsperados` (originales: $seq\langle audio\rangle$, bajados: $seq\langle audio\rangle$, in p : $\mathbb{Z}$, in p2 : $\mathbb{Z}$) {
$(\ \forall i : \mathbb{Z})(0 \leq i < |originales| \longrightarrow_L$ esElAudioDeMenorCalidadEsperado(originales[i],bajados[i],p,p2)
)}

`pred esElAudioDeMenorCalidadEsperado` (original: audio, bajado: audio, in p : $\mathbb{Z}$, in p2 : $\mathbb{Z}$) {
$(\ \forall i : \mathbb{Z})(0 \leq i < |originales| \wedge_L original[i] \geq 0 \longrightarrow_L bajado[i] = original[i]$ div $2^{p-p2})$
$\wedge$
$(\forall i : \mathbb{Z})(0 \leq i < |originales| \wedge_L$ original[i]$< 0 \longrightarrow_L bajado[i] = (-1) * (abs(original[i])$ div $2^{p-p2})$
) }

fin 6)

7)

pred softYHard (sa : $seq\langle audio\rangle$, long : $\mathbb{Z}$, umbral :$\mathbb{Z}$, soft : $seq\langle audio\rangle$, hard : $seq\langle audio\rangle$) {
$(\forall i : \mathbb{Z})(\ 0 \leq i < |sa| \longrightarrow_L ((\text{ esHard(long, umbral, sa[i]}) \longrightarrow_L \text{sa[i]} \in hard) \wedge$
$(\ \neg esHard(long, umbral, sa[i]) \longrightarrow_L \text{sa[i]} \in soft))\}$

pred esHard (long : $\mathbb{Z}$, umbral : $\mathbb{Z}$, s : $seq\langle\mathbb{Z}\rangle$) {
$(\exists j : \mathbb{Z})(\ 0 \leq j < |s| - (long + 1) \wedge_L$ sumatoriaDeLargoLong+1(s, long, umbral, j) = (long + 1)
}

aux sumatoriaDeLargoLong+1 (s : $seq\langle\mathbb{Z}\rangle$, long : $\mathbb{Z}$, umbral : $\mathbb{Z}$, j : $\mathbb{Z}$) : $\mathbb{Z}$=

$\sum_{k=j}^{j+long}$ (if abs(s[k]) > abs(umbral) then 1 else 0 fi)

fin 7)

8)
pred apareceALoSumoUnaVezEn ($a_1 : audio$, $A_0 : audio$) {
$noAparece(a_1, A_0) \ \vee$ apareceUnaVez(a_1, A_0)}

pred apareceUnaVez ($a_1 : audio$, $A_0 : audio$) {
$(\exists x : tupla)(0 \leq x_0 \leq |A_0|) \ \wedge_L \ 0 \leq x_1 \leq |A_0| \ \wedge_L \ subseq(A_0, x_0, x_1) = a_1$
$\wedge$
$\neg(\exists y : tupla)(0 \leq x_0 \leq |A_0| \ \wedge 0 \leq y_1 \leq |A_0|) \ \wedge x_0 \neq y_0 \ \wedge x_1 \neq y_1 \ \wedge x_0 \neq y_1$
$\wedge_L subseq(A_0, y_0, y_1) = a_1$)) }

pred aparece ($a_1 : audio$, $A_0 : audio$) {
$(\exists x : tupla)(0 \leq x_0 \leq |A_0|) \wedge_L 0 \leq x_1 \leq |A_0| \wedge_L subseq(A_0, x_0, x_1) = a_1$ }

pred noAparece ($a_1 : audio$, $A_0 : audio$) {
$\neg aparece(a_1, A_0)$}

aux concatTriple(a : audio, b : audio, c : audio) : audio = $concat$(a, $concat$(b,c));

pred esConcatTripleConA$_2$ (a : audio, $A_0 : audio$, $a_1 : audio$,$a_2 : audio$,) {
$(\exists x : \mathbb{Z})(0 \leq x \leq |A_0| \ \wedge_L$ ($A_0 = concatTriple(subseq(A_0, 0, x), a1, subseq(A_0, x + |a_1|, |A_0|))$
$\wedge (a = concatTriple(subseq(A_0, 0, x), a2, subseq(A_0, x + |a_1|, |A_0|))))\}$

pred aparece ($a_1 : audio$, $A_0 : audio$) {
$(\exists x : tupla)(0 \leq x_0 \leq |A_0|) \ \wedge_L \ 0 \leq x_1 \leq |A_0| \ \wedge_L \ subseq(A_0, x_0, x_1) = a_1$ }
fin 8)

9)
pred máximoDeIntervalosOrdenado (intervalos:$seq\langle\mathbb{Z}x\mathbb{Z}\rangle$, a : audio, máximos: $seq\langle\mathbb{R}\rangle$)
{

$(\forall i : \mathbb{Z}\ )((0 \leq i \leq |intervalos| \wedge |intervalos| = |máximos|) \longrightarrow_L$ `(máximos[i] ≥ a[intervalos[i]`$_0$`]` $\wedge$ `máximos[i] ≥ a[intervalos[i]`$_1$`])` $\wedge$ `(máximos[i] = a[intervalos[i]`$_0$`]` $\vee$ `máximos[i] = a[intervalos[i]`$_1$`])`$)$

`pred intervalosVálidos ( intervalos:`$seq\langle \mathbb{Z}x\mathbb{Z} \rangle$`, tiempos:` $seq\langle \mathbb{Z} \rangle$`) {`
$(\forall i : \mathbb{Z}\ )(\exists j : \mathbb{Z}\ )(((0 \leq j < |tiempos| \wedge 0 \leq i < |intervalos| \wedge_L intervalos[i]_0 \geq 0) \longrightarrow_L$
`intervalos[i]`$_1 = intervalos[i]_0 + (tiempos[j] - 1)))$ `}`

fin 9)

10)
`pred MismoAudioSinOutliers` (a: audio, $A_0 : Audio$){
$OutliersCambien(a, A_0) \wedge$ noOutliersIgual(a,A_0); }

`pred OutliersCambien` (a: audio, $A_0 : Audio$){
$(\forall i : \mathbb{Z})((0 \leq i < |a| \wedge_L$ esOutlier($A_0[i], A_0$)) $\longrightarrow_L$ estaCambiado(a[i],A_0, i); }

`pred noOutliersIgual` (a: audio, $A_0 : Audio$){
$(\forall i : \mathbb{Z})((0 \leq i < |a| \wedge_L \neg$ esOutlier($A_0[i], A_0$)) $\longrightarrow_L$ a[i] = $A_0[i]$; }

`pred esOutlier` (x: $\mathbb{Z}$, $A_0 : Audio$){
($porcentajeTieneDecimales(|A_0|) \wedge$ (MayorqueCuantos(x,A_0) > $porcentajeParteEntera(|A_0|)+$
1) $\vee$ ($\neg$ porcentajeTieneDecimales($|A|_0$) $\wedge$ (MayorqueCuantos(x,A_0) > $porcentajeParteEntera(|A_0|)$); }

aux MayorqueCuantos(b : $\mathbb{Z}$, s: $seq\langle \mathbb{Z} \rangle$) : $\mathbb{Z}$=
$\sum_{k=0}^{|s|-1}$ if $abs(b) > abs(s[i])$ then 1 else 0 fi;

$predporcentajeTieneDecimales\ (b : \mathbb{Z})$){
$95 * b\ mod\ 100 \neq 0$;}

aux porcentajeParteEntera(b : $\mathbb{Z}$) : $\mathbb{Z}$= 95*b div 100;

`pred estaCambiado` (outlier: $\mathbb{Z}$, $A_0 : Audio, índiceOutlier : \mathbb{Z}$){
$notieneNormalesIzquierda(outlier, A_0, índiceOutlier) \vee$ notieneNormalesDerecha(outlier, $A_0, índiceOutlier$)
$\vee_L$ esIgualalpromedioDeSusCostados(outlier, $A_0, índiceOutlier$); }

`pred notieneNormalesIzquierda` (outlier: $\mathbb{Z}$, $A_0 : Audio, índiceOutlier : \mathbb{Z}$){
$notieneNormales(subseq(A_0, 0, índiceOutlier)) \wedge$ esIgualalPrimerNormalDerecha(outlier,subseq($A_0, indiceOutlier+$
$1, |A_0|), A_0$); }

`pred notieneNormalesDerecha` (outlier: $\mathbb{Z}$, $A_0 : Audio, índiceOutlier : \mathbb{Z}$){
$notieneNormales(subseq(A_0, indiceOutlier+1, |A_0|)) \wedge$ esIgualalPrimerNormalIzquierda(outlier,subseq
$(A_0, 0, indiceOutlier), A_0$); }

`pred esIgualalPrimerNormalDerecha` (outlier: $\mathbb{Z}$, s: Audio,A_0 : *audio*){
$(\exists i : \mathbb{Z})(0 \leq i < |s| \wedge_L$ outlier=s[i] $\wedge_L$ notieneNormales(subseq(s,0, i)) $\wedge \neg$ esOutlier(s[i],A_0); }

`pred esIgualalPrimerNormalIzquierda` (outlier: $\mathbb{Z}$, s: Audio, A_0 : *audio*){
$(\exists i : \mathbb{Z})(0 \leq i < |s| \wedge_L$ outlier=s[i] $\wedge_L$ notieneNormales(subseq(s,i +1,$|s|$))$\wedge \neg$ esOutlier(s[i],A_0); }

`pred esIgualalpromedioDeSusCostados` (outlier: $\mathbb{Z}$,A_0 : *audio*, *índiceOutlier* : $\mathbb{Z}$){$(\exists x : tupla)(0 \leq x_0 < |A_0| \wedge 0 \leq x_1 < |A_0| \wedge (x_1 > índiceOutlier > x_0) \wedge_L (outlier = (A_0[x_0] + A_0[x_1])$ *div* 2) $\wedge \neg$ esOutlier($A_0[x_0], A_0$) $\wedge \neg$ esOutlier($A_0[x_1], A_0$)$\wedge$ notieneNormales(subseq($A_0, x_0 + 1, indiceOutlier$), A_0)$\wedge$ notieneNormales(subseq($A_0, indiceOutlier, x_1$), A_0); }

`pred notieneNormales` (s: Audio,A_0 : *audio*){
$(\forall i : \mathbb{Z})(0 \leq i < |s| \longrightarrow_L$ esOutlier(s[i],A_0); }

`pred AtípicosContieneLosÍndicesdelosOutliers` (atípicos: $seq\langle\mathbb{Z}\rangle$, A_0 : *audio*){
$cantOutliers(A_0) = |atípicos| \wedge$ TodosDistintos(atípicos) $\wedge$ lasPosicionesEstanEnRango($A_0, atípicos$)
$\wedge$ seanTodosOutliers($A_0, atípicos$); }

aux cantOutliers(A_0 : *audio*) : $\mathbb{Z} = \sum_{k=0}^{|A_0-1|}$ if $esOutlier(A_0[i], A_0)$ then 1 else 0 fi;

`pred existendosIguales` (atípicos : $seq\langle\mathbb{Z}\rangle$) {
$(\exists x : tupla)(0 \leq x_0 < |A_0| \wedge 0 \leq x_1 < |A_0| \wedge (x_1 \neq x_0) \wedge_L$ atípicos[x_0] = $atípicos[x_1]$}

`pred TodosDistintos` (atípicos : audio) {$\neg$ existendosIguales(atípicos)}

`pred seanTodosOutliers` (A_0 : $Audio, atípicos : seq\langle\mathbb{Z}\rangle$){$(\forall i : \mathbb{Z})((0 \leq i < |atípicos| \longrightarrow_L$ esOutlier($A_0[atípicos[i]], A_0$); }

fin 10)

3. Decisiones tomadas

En el ejercicio 4, tomamos como criterio que las dos listas devueltas poseen el mismo orden,por ende el índice mas uno corresponden al canal del que hablan. También utilizamos que ser magnitud maxima absoluta, quiere decir que su valor absoluto es mayor o igual al resto de elementos. O sea, decimos que a[i] es magnitud maxima absoluta de un canal c en un audio a, con formato valido, si para toda posicion j,i que este en el rango de posiciones del audio y canal c, a[j] $\leq a[i]$. $i, j, c \in \mathbb{Z}$.

Para resolver el ejercicio 6, en el pred esElAudioDeMenorCalidadEsperado, usamos div que esta en el preludio, para que nos devuelva la division entera y asumimos y tuvimos en cuenta que al dividir un numero negativo siempre redondea alejandose del cero, ejemplo: (-10) div 3 = -4

En el ejercicio 7, tomamos que el módulo de las magnitudes tenía que ser mayor al módulo del umbral . Por ejemplo si el audio fuese (6,5,-8,7,-6) y el umbral fuese |3| (y el long 4) entonces sería un audio hard. Ya que el modulo de todos estos valores son mayores que 3.

En el ejercicio 10 , para ver que una magnitud sea outlier decimos que va a ser mayor estricto a todos los elementos que se encontrarían detrás del percentil inclusive si el audio estuviese ordenado y él formase parte. Para encontrar el índice, en esa lista, de ese percentil utilizamos la fórmula de Wikipedia, si el resultado tiene decimales el índicePercentil = parte entera +1, en ese caso ya formaría parte de la secuencia y mi outlier debería ser mayor a la cantidad de elementos anteriores desde audioOrdenado[parte entera +1], si no tiene decimales percentil=((audioOrdenado[parte entera]+ audioOrdenado[parte entera + 1]) div 2) entonces el percentil estaría ubicado en el medio de esos dos por ende debería ser mayor al primero , a lo sumo igual pero en tal caso no afectaría la resolución porque mi outlier debería seguir siendo mayor a este, tampoco lo haria el caso en el que es igual al elemento siguiente porque solamente sucede cuando las dos magnitudes son iguales. Y también como hicimos en ejercicios anteriores tomamos magnitud como valor absoluto, por eso cuando hablamos de la audioOrdenado estaría formado por los valores absolutos del audio.

Observación General: En algunos casos despues de escribir un auxiliar, nos sucede que el predicado siguiente cambia el formato y aparece en cursiva prednombreDelPredicado. Es un pequeño detalle que en algunos auxiliares pudimos resolver, y en otros no.