

Insuficiente



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

TPE - Reuniones Remotas

Grupo 6 - Alrescate.com

Integrante	LU	Correo electrónico
Nicolas Andres Kinaschuk	348/15	nicolaskinaschuk@gmail.com
Maria Vanesa Reyes Reyes	398/05	vanereyes22@gmail.com
Camilo Manuel DÁloisio	800/18	camilodaloisio@gmail.com
Sebastian E C Speranza	120/19	sebacagnoni@gmail.com

Notas generales:

- * Hay algunos usos incorrectos de "esReunionValidaAux" en las post. Remover o justificar todas sus apariciones en las post.
- * Faltan espacios y tabulaciones en las funciones. Puse notas en el ejercicio 1 pero aplica para todos. Es importante la prolijidad.
- * Recomendamos utilizar `result = true <=> predicado`, en vez de `result = predicado`
- * Para la re-entrega, arreglar las correcciones mencionadas



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>

Índice

Índice	1
Ejercicio 1	2
Ejercicio 2	3
Ejercicio 3	4
Ejercicio 4	5
Ejercicio 5	6
Ejercicio 6	7
Ejercicio 7	8
Ejercicio 8	9
Ejercicio 9	10
Ejercicio 10	11

1. Ejercicio 1

```
proc esSeñal (in s: seq(Z), in prof: Z, in freq: Z, out result: Bool) {  
  Pre {|s| > 0 ∧ prof > 0 ∧ freq > 0}  
  Post {  
    result = esSeñalAux(s, prof, freq)  
  }  
}
```

Por qué no aceptan secuencias vacías? El proc debería decir cuando una señal es válida o no, para cualquier señal.

Falta tabulación

mal tabulado

```
pred esSeñalAux (s: seq(Z), prof: Z, freq: Z) {  
  |s| ≥ 0 ∧  
  frecuenciaEnRango(freq) ∧  
  profundidadCorrecta(s) ∧  
  duraMasDeUnSegundo(s, freq) ∧  
  ningunaMuestraSuperaLaProfundidad(s, prof)  
}
```

Variable no ligada

```
pred frecuenciaEnRango (freq: Z) {freq ∈ [8, 32]}  
pred profundidadCorrecta (prof: Z) {freq ∈ [8, 16, 32]}  
pred duraMasDeUnSegundo (s: seq(Z), freq: Z) {  
   $\frac{|s|}{(freq-1000)} \geq 1$   
}  
pred ningunaMuestraSuperaLaProfundidad (s: seq(Z), p: Z) {  
  (∀ i : Z) 0 ≤ i < |s|  
  →L (-2)p-1 ≤ s[i] ≤ 2p-1 - 1  
}
```

Se indefinido cuando freq = 0

debería estar todo en la misma línea

Falta espacio entre predicados

2. Ejercicio 2

```

proc seEnojó? (in s: señal, in umbral:  $\mathbb{Z}$ , in prof:  $\mathbb{Z}$ , in freq:  $\mathbb{Z}$ , out result: Bool) {
  Pre {umbral > 0  $\wedge$  esSeñalAux(s, prof, freq)}
  Post {
    result = umbralEnRango(umbral, prof)  $\wedge$ 
    existeUnaSubsecuenciaQueSuperaUmbral(s, freq, umbral)
  }
}

```

Que pasa si el umbral está entre 0 y 2^{p-1} ? result da siempre false

```

pred umbralEnRango (umbral:  $\mathbb{Z}$ , p:  $\mathbb{Z}$ ) {umbral  $\geq 2^{p-1} - 1$ }
pred existeUnaSubsecuenciaQueSuperaUmbral (s: señal, freq:  $\mathbb{Z}$ , u:  $\mathbb{Z}$ ) {
  ( $\exists d, h : \mathbb{Z}$ )  $0 \leq d, h < |s| + 1 \wedge (h > (d + freq * 1000 * 5)) \wedge_L ($ 
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |subseq(s, d, h)| \rightarrow_L abs(subseq(s, d, h)[i]) > u$ )
}
fun abs (x:  $\mathbb{Z}$ ) :  $\mathbb{Z}$  = if  $x > 0$  then  $x$  else  $-x$  fi;

```

3. Ejercicio 3

```
proc esReuniónVálida? (in r: reunion, in prof:  $\mathbb{Z}$ , in freq:  $\mathbb{Z}$ , out result: Bool) {  
  Pre  $\{|r| > 0 \wedge prof > 0 \wedge freq > 0\}$   
  Post  $\{result = esReuniónVálidaAux(r, prof, freq)\}$   
}  
  
pred esReuniónVálidaAux (r: reunion, prof:  $\mathbb{Z}$ , freq:  $\mathbb{Z}$ ) {  
  contieneSeñalesValidas(r, prof, freq)  $\wedge$   
  lasLongitudesDeSeñalSonIguales(r)  $\wedge$   
  todosHablantesDistintos(r)  $\wedge$   
  losHablantesEstanEnRangosDe0ANMenos1(r) }  
pred contieneSeñalesValidas (r: reunion, prof:  $\mathbb{Z}$ , freq:  $\mathbb{Z}$ )  $\{(\forall i : \mathbb{Z}) 0 \leq i < |r| \rightarrow_L esSeñalAux(r[i]_0, prof, freq)\}$   
pred lasLongitudesDeSeñalSonIguales (r: reunion)  $\{(\forall i, j : \mathbb{Z}) 0 \leq i, j < |r| \wedge i \neq j \rightarrow_L (|r[i]_0| = |r[j]_0|)\}$   
pred todosHablantesDistintos (r: reunion)  $\{(\forall i, j : \mathbb{Z}) 0 \leq i, j < |r| \wedge i \neq j \rightarrow_L (r[i]_1 \neq r[j]_1)\}$   
pred losHablantesEstanEnRangosDe0ANMenos1 (r: reunion)  $\{(\forall i : \mathbb{Z}) 0 \leq i < |r| \rightarrow_L 0 \leq r[i]_1 < |r|\}$ 
```

Si la funcion me tiene que decir que es una reunion valida, porque no puedo pasar una reunion vacia? o una profundidad negativa?

4. Ejercicio 4

```

proc acelerar (inout r: reunion, in prof:  $\mathbb{Z}$ , in freq:  $\mathbb{Z}$ ) {
  Pre {esReuniónVálidaAux(r, prof, freq)  $\wedge$   $r_0 = r$ }
  Post {
    esReuniónVálidaAux(r, prof, freq)  $\wedge$ 
    ( $|r| = |r_0| \wedge_L$ 
    lasSeñalesTieneLaMitadDeMuestras(r,  $r_0$ )  $\wedge_L$ 
    losImpares(r,  $r_0$ ))}
}

```

Para que lo utilizan? Revisen si es necesario, y si lo es justificar

Que pasa si la señal resultado (solo las impares) dura menos de un segundo? La señal se vuelve invalida!

```

pred lasSeñalesTieneLaMitadDeMuestras (r: reunion,  $r_v$  : reunion){
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |r| \rightarrow_L$  if esPar( $|r_v[i]_0|$ ) then  $|r[i]_0| = \frac{|r_v[i]_0|}{2}$  else  $|r[i]_0| = \frac{|r_v[i]_0|-1}{2}$  fi
}
pred losImpares (r: reunion,  $r_v$  : reunion){( $\forall i : \mathbb{Z}$ )  $0 \leq i < |r_v| \rightarrow_L$  (
  ( $\exists j : \mathbb{Z}$ )  $0 \leq j < |r| \wedge_L (r_v[i]_1 = r[j]_1) \wedge_L$ 
  ( $\forall q : \mathbb{Z}$ )  $0 \leq q < |r_v[i]_0| \wedge (\neg esPar(q)) \rightarrow_L (r_v[i]_0[q] = r[j]_0[\frac{q-1}{2}]))$ )}

```

5. Ejercicio 5

```

proc ralentizar (inout r: reunion, in prof:  $\mathbb{Z}$ , in freq:  $\mathbb{Z}$ ) {
  Pre {esReuniónVálidaAux(r, prof, freq)  $\wedge$   $r_0 = r$ }
  Post {
    esReuniónVálidaAux(r, prof, freq)  $\wedge$ 
    ( $|r| = |r_0| \wedge_L$ 
    lasSeñalesTienenElDobleDeMuestras(r,  $r_0$ )  $\wedge_L$ 
    promedioEntrePares(r,  $r_0$ ))}
}

pred lasSeñalesTienenElDobleDeMuestras (r: reunion,  $r_v$  : reunion){
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |r_v| \longrightarrow_L (2 \cdot |r_v[i]_0|) = (|r[i]_0| + 1)$ }
pred promedioEntrePares (r: reunion,  $r_v$  : reunion){
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |r| \longrightarrow_L ($ 
  ( $\exists j : \mathbb{Z}$ )  $0 \leq j < |r_v| \wedge_L (r[i]_1 = r_v[j]_1) \wedge_L ($ 
  ( $\forall q : \mathbb{Z}$ )  $0 \leq q < |r[i]_0| \longrightarrow_L$ 
  if esPar(q) then  $r[i]_0[q] = r_v[j]_0[\frac{q}{2}]$  else  $r[i]_0[q] = \frac{r_v[j]_0[\frac{q-1}{2}] + r_v[j]_0[\frac{q+1}{2}]}{2}$  fi ) ) }

```

6. Ejercicio 6

```

proc tonosDeVozElevados (inout r: reunion, in freq:  $\mathbb{Z}$ , in prof:  $\mathbb{Z}$ , out hablantes:  $seq\langle hablante \rangle$ ) {
  Pre {esReuniónVálidaAux(r, prof, freq)}
  Post {
    siPertenecenAHablantesElPromedioDeAmplitudEsMasGrandeOIgualQueElResto(r, hablantes)  $\wedge$ 
    losHablantesPertenecenALaReunión(r, hablantes)  $\wedge$ 
    losHablantesNoSeRepiten(hablantes)}
}

pred siPertenecenAHablantesElPromedioDeAmplitudEsMasGrandeOIgualQueElResto (r: reunion, hs:  $seq\langle hablante \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |hs| \rightarrow_L$ 
  ( $r[i]_1 \in hs \wedge elPromedioDeAmplitudEsMasGrandeOIgualQueElResto(r, r[i]_0)$ )
   $\vee$ 
  ( $r[i]_1 \notin hs \wedge \neg elPromedioDeAmplitudEsMasGrandeOIgualQueElResto(r, r[i]_0)$ )
}

pred elPromedioDeAmplitudEsMasGrandeOIgualQueElResto (r: reunion, s: señal) {
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |r| \rightarrow_L (tonoDeVoz(s) \geq tonoDeVoz(r[i]_0))$ 
}

fun tonoDeVoz (s: señal) :  $\mathbb{Z} = sumaDelValorAbsolutoDeAmplitudes(s) \div |s|$ ;
fun sumaDelValorAbsolutoDeAmplitudes (s: señal) :  $\mathbb{Z} = \sum_{i=0}^{|s|} abs(s[i])$ ;
pred losHablantesPertenecenALaReunión (r: reunion, hs:  $seq\langle hablante \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |hs| \rightarrow_L ((\exists j : \mathbb{Z}) 0 \leq j < |r| \wedge_L (hs[i] = r[j]_1))$ 
}
pred losHablantesNoSeRepiten (r: reunion, hs:  $seq\langle hablante \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |hs| \rightarrow_L (\#apariciones(hs, hs[i]) = 1)$ 
}

```

Que pasa con un r tal que
 $|r| = 2$ y $|hs| > |r|$? Se indefin!

7. Ejercicio 7

```

proc ordenar (inout r: reunion, in freq:  $\mathbb{Z}$ , in prof:  $\mathbb{Z}$ ) {
  Pre {esReuniónVálidaAux(r, prof, freq)  $\wedge$   $r_0 = r$ }
  Post {
    esReuniónVálidaAux(r, prof, freq)  $\wedge$ 
    ordenadaDeMayorAMenorPorTonoDeVoz(r)  $\wedge$ 
    esUnaPermutación( $r_0$ , r)}
}

```

```

pred ordenadaDeMayorAMenorPorTonoDeVoz (r: reunion) {
  ( $\forall i : \mathbb{Z}$ )  $1 \leq i < |r| \rightarrow_L \text{tonoDeVoz}(r[i-1]_0) \geq \text{tonoDeVoz}(r[i]_0)$ 
}

```

```

pred esUnaPermutación (x: reunion, y: reunion) {
   $|x| = |y| \wedge_L$ 
  ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |x| \rightarrow_L ($ 
  ( $\exists j : \mathbb{Z}$ )  $0 \leq j < |y| \wedge_L (x[i]_1 = y[j]_1) \wedge_L (x[i]_0 = y[j]_0))$ 
}

```

→ Acá hay un “luego” que no es necesario

8. Ejercicio 8

```

proc silencios (in s: señal, in freq:  $\mathbb{Z}$ , in prof:  $\mathbb{Z}$ , out intervalos:  $seq\langle intervalo \rangle$ ) {
  Pre {esSeñalAux(s, prof, freq)  $\wedge$  (umbral > 0)}
  Post {
    noHayIntervalosRepetidos(intervalos)  $\wedge$ 
    ( $\forall i : \mathbb{Z}$ )  $0 \leq i < |intervalos| \rightarrow_L$  (
      esSilencio(s, umbral, freq, intervalos[i]))
  }
}

pred esSilencio (s: señal, umbral:  $\mathbb{Z}$ , freq:  $\mathbb{Z}$ , in: intervalo) {
  finEsMayorQueInicio(in0, in1)  $\wedge$ 
  estaDentroDeLaSeñal(s, in0, in1)  $\wedge$ 
  esAlMenosUnDecimoDeSegundo(freq, in0, in1)  $\wedge$ 
  entreIndicesNoPasaCiertUmbral(s, umbral, in0, in1)  $\wedge$ 
  losAdyacentesSuperanElUmbral(s, umbral, in0, in1)
}
pred noHayIntervalosRepetidos (ins:  $seq\langle intervalo \rangle$ ) {( $\forall i : \mathbb{Z}$ )  $0 \leq i < |ins| \rightarrow_L$  (#apariciones(ins, e) = 1)}
pred finEsMayorQueInicio (inicio:  $\mathbb{Z}$ , fin:  $\mathbb{Z}$ ) {fin > inicio}
pred estaDentroDeLaSeñal (s: señal, inicio:  $\mathbb{Z}$ , fin:  $\mathbb{Z}$ ,) {(inicio  $\geq 0$ )  $\wedge$  (fin < |s|)}
pred esAlMenosUnDecimoDeSegundo (freq:  $\mathbb{Z}$ , inicio:  $\mathbb{Z}$ , fin:  $\mathbb{Z}$ ,) {(fin - inicio + 1)  $\geq$  (frecuencia * 100)}
pred losAdyacentesSuperanElUmbral (s: señal, umbral:  $\mathbb{Z}$ , inicio:  $\mathbb{Z}$ , fin:  $\mathbb{Z}$ ) {
  ( (inicio = 0)  $\vee$  ((inicio - 1  $\geq 0$ )  $\wedge_L$  (s[inicio - 1]  $\geq$  umbral)))  $\wedge$ 
  (fin = |s| - 1)  $\vee$  ((fin + 1 < |s|)  $\wedge_L$  (s[fin + 1]  $\geq$  umbral)))
}
pred entreIndicesNoPasaCiertUmbral (s: señal, umbral:  $\mathbb{Z}$ , inicio:  $\mathbb{Z}$ , fin:  $\mathbb{Z}$ ) {
  ( $\forall i : \mathbb{Z}$ ) inicio  $\leq i < fin + 1 \rightarrow_L$  (abs(s[i])  $\leq$  umbral)}

```

Si inicio > |s|+1, se indefine!

debería ser estricto

9. Ejercicio 9

```

proc hablantesSuperpuestos (in r: reunion, in prof:  $\mathbb{Z}$ , in freq:  $\mathbb{Z}$ , in umbral:  $\mathbb{Z}$ , out result: Bool) {
  Pre {esReuniónVálidaAux(r, prof, freq)}
  Post {result =  $\neg$ noHayHablantesSuperpuestos(r, freq, umbral)}
}

pred noHayHablantesSuperpuestos (r: reunion, freq:  $\mathbb{Z}$ , umbral:  $\mathbb{Z}$ ) {
  ( $\forall i, j : \mathbb{Z}$ )  $0 \leq i, j < |r| \wedge (i \neq j) \longrightarrow_L$ 
  ( $\forall k, l : \mathbb{Z}$ )  $0 \leq k, l < |r[i]_0| \wedge k < l \longrightarrow_L$ 
   $\neg$ haySilencio(subseq(r[i]0, k, l), umbral, freq)  $\longrightarrow_L$  esSilencio(r[j]0, umbral, freq, (k, l))}
pred haySilencio (s: señal, umbral:  $\mathbb{Z}$ , freq:  $\mathbb{Z}$ ) {( $\exists i, j : \mathbb{Z}$ )  $0 \leq i, j < |s| \wedge (i < j) \wedge_L$  esSilencio(s, umbral, (i, j))}

```

10. Ejercicio 10

```

proc reconstruir (in s: señal, in prof:  $\mathbb{Z}$ , in freq:  $\mathbb{Z}$ , out señal: Bool) {
  Pre {esSeñalAux(s, prof, freq)} ← Faltaria decir que hay al menos dos valores no nulos
  Post {esSeñalAux(result) ∧
    (|s| = |result|) ∧L
    enDondeNoSeaCeroDebenCoincidir(s, result) ∧L
    enDondeEsCeroDebeSerElPromedioDeSusVecinosNoNulos(s, result)} } ← Acá hay una contradicción
}

pred enDondeEsCeroDebeSerElPromedioDeSusVecinosNoNulos (original: señal, reconstruida: señal) {
  (∀i :  $\mathbb{Z}$ ) 0 ≤ i < |original| →L
  (original[i] = 0) ∧L
  reconstruida[i] = promedioDeVecinosNoNulos(original[i], reconstruida[i]) }
fun promedioDeVecinosNoNulos (s: señal, i:  $\mathbb{Z}$ ) :  $\mathbb{Z}$  =  $\frac{(s[\text{elIndiceNoNuloMasCercano}(s,i)] + s[\text{el2doIndiceNoNuloMasCercano}(s,i)])}{2}$ ;
fun elIndiceNoNuloMasCercano (s: señal, i:  $\mathbb{Z}$ ) :  $\mathbb{Z}$  =
  if dist(i, indiceSiguienteNoNulo(s, i)) < dist(i, indiceAnteriorNoNulo(s, i)) then
    indiceSiguienteNoNulo(s, i) else
  if dist(i, indiceSiguienteNoNulo(s, i)) > dist(i, indiceAnteriorNoNulo(s, i)) then
    indiceAnteriorNoNulo(s, i) else ← no tipa
    indiceSiguienteNoNulo(s, i) fi fi;
fun dist (x:  $\mathbb{Z}$ , y:  $\mathbb{Z}$ ) :  $\mathbb{Z}$  = abs(x - y);
fun indiceAnteriorNoNulo (s: señal, i:  $\mathbb{Z}$ ) :  $\mathbb{Z}$  =  $\sum_{p=0}^{i-1}$  if esElPrimerAnteriorNoNulo(s, i, p) then p else 0 fi;
pred esElPrimerAnteriorNoNulo (s: señal, i:  $\mathbb{Z}$ , p:  $\mathbb{Z}$ ) { (∀j :  $\mathbb{Z}$ ) p ≤ j < i →L (s[j] = 0) ∧L (s[p] ≠ 0) } ← Da siempre false!
fun indiceSiguienteNoNulo (s: señal, i:  $\mathbb{Z}$ ) :  $\mathbb{Z}$  =  $\sum_{p=i+1}^{|s|-1}$  if esElPrimerSiguienteNoNulo(s, i, p) then p else 0 fi;
pred esElPrimerSiguienteNoNulo (s: señal, i:  $\mathbb{Z}$ , p:  $\mathbb{Z}$ ) { (∀j :  $\mathbb{Z}$ ) i ≤ j < p →L (s[j] = 0) ∧L (s[p] ≠ 0) }
fun el2doIndiceNoNuloMasCercano (s: señal, i:  $\mathbb{Z}$ ) :  $\mathbb{Z}$  =
  elIndiceNoNuloMasCercano(setAt(s, elIndiceNoNuloMasCercano(s, i), 0)) ;
pred enDondeNoSeaCeroDebenCoincidir (original: señal, reconstruida: señal) {
  (∀i :  $\mathbb{Z}$ ) 0 ≤ i < |original| →L
  (original[i] ≠ 0) ∧L
  (original[i] = reconstruida[i]) }

```

Recomiendo re-hacerlo completo este ejercicio, o realizar los cambios que marqué más los que sean necesarios para arreglarlo.