



TP de Especificación

Análisis Habitacional Argentino

8 de Septiembre de 2021

Algoritmos y Estructuras de Datos I

Grupo 7

Integrante	LU	Correo electrónico
Acuña, Martín	596/21	acunamartin1426@gmail.com
Castro, Lucía	278/21	lucia.ines.castro.98@gmail.com
Clas, Giulia	11/15	clas.giulia.s@gmail.com
Seidler, Daniel	973/12	danieljseidler@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

```

proc encuestaVálida (in th:  $eph_h$ , in ti:  $eph_i$ , out result: Bool) {
  Pre {true}
  Post {result = true  $\leftrightarrow$  laEncuestaEsValida(th, ti)}
}

proc histHabitacional (in th:  $eph_h$ , in ti:  $eph_i$ , in region:  $\mathbb{Z}$ , out res:  $seq\langle\mathbb{Z}\rangle$ ) {
  Pre {laEncuestaEsValida(th, ti)  $\wedge$  laRegionEsValida(region)}
  Post {( $\forall i : \mathbb{Z})(0 \leq i < |res| \rightarrow_L$ 
    res[i] = casasConNHabitaciones(th, region, i + 1))  $\wedge$  esMaxDeHabitaciones(|res|, th, region)}
}

proc laCasaEstaQuedandoChica (in th:  $eph_h$ , in ti:  $eph_i$ , out res:  $seq\langle\mathbb{R}\rangle$ ) {
  Pre {laEncuestaEsValida(th, ti)}
  Post {( $\forall i : \mathbb{Z})(0 \leq i < |res| \rightarrow_L$ 
    res[i] =  $\frac{casasConHacinamientoCriticoEnLaRegion(th, ti, i)}{hogaresEnLaRegion(th, i)}$ )}
}

proc creceElTeleworkingEnCiudadesGrandes (in t1h:  $eph_h$ , in t1i:  $eph_i$ , in t2h:  $eph_h$ , in t2i:  $eph_i$ , out result: Bool) {
  Pre {(laEncuestaEsValida(t1h, t1i)  $\wedge$  laEncuestaEsValida(t2h, t2i))  $\wedge$  L(t1h[0][ord(HOGANO)] <
    t2h[0][ord(HOGANO)]  $\wedge_L$  t1h[0][ord(HOGTRIMESTRE)] = t2h[0][ord(HOGTRIMESTRE)]}
  Post {result =
     $\frac{cantidadPersonasQueHacenTeleworkingEnCiudadGrande(t1h, t1i)}{cantidadPersonasEnCiudadGrande(t1h, t1i)} < \frac{cantidadPersonasQueHacenTeleworkingEnCiudadGrande(t2h, t2i)}{cantidadPersonasEnCiudadGrande(t2h, t2i)}$ 
}
}

proc costoSubsidioMejora (in th:  $eph_h$ , in ti:  $eph_i$ , in monto:  $\mathbb{Z}$ , out res:  $\mathbb{Z}$ ) {
  Pre {laEncuestaEsValida(th, ti)  $\wedge$  monto > 0}
  Post {res = monto * hogaresCandidatosASubsidio(th, ti)}
}

proc generarJoin (in th:  $eph_h$ , in ti:  $eph_i$ , inout junta:  $joinHI$ ) {
  Pre {laEncuestaEsValida(th, ti)}
  Post {( $\forall i : \mathbb{Z})(0 \leq i < filas(junta) \rightarrow_L es2 - upla(junta[i]))$ }
}

proc ordenarRegionYTipo (inout th:  $eph_h$ , inout ti:  $eph_i$ ) {
  Pre {laEncuestaEsValida(th, ti)  $\wedge$  th = th0  $\wedge$  ti = ti0}
  Post {mismoTamanioDeTabla(th, th0)  $\wedge$  mismoTamanioDeTabla(ti, ti0)  $\wedge$ 
    th  $\subseteq$  th0  $\wedge$  ti  $\subseteq$  ti0  $\wedge$  ordenadoPorRegion(th)  $\wedge$ 
    regionesOrdenadasPorHogcodusu(th)  $\wedge$  mismoOrdenDeCodusu(th, ti)  $\wedge$  casasOrdenadasPorComponente(ti)}
}

proc muestraHomogenea (in th:  $eph_h$ , in ti:  $eph_i$ , out res:  $seq\langle hogar \rangle$ ) {
  Pre {laEncuestaEsValida(th, ti)}
  Post {(estaOrdenadaPorIngresos(ti, res)
     $\wedge$  diferenciaDeIngresosConstante(ti, res)
     $\wedge$  |res|  $\geq$  3
     $\wedge$  esLaSecuenciaMasGrande(th, ti, res))
     $\vee$  |res| = 0}
}

proc corregirRegion (inout th:  $eph_h$ , inout ti:  $eph_i$ ) {
  Pre {laEncuestaEsValida(th, ti)  $\wedge$  th0 = th}
  Post {(th0 = |th|  $\wedge$  cantColumnasHogares(th0) = cantColumnasHogares(th)  $\wedge$  ( $\forall i : \mathbb{Z})(\forall col : \mathbb{Z})$ 
    (0  $\leq i < |th| \wedge col \neq ord(REGION) \rightarrow_L$  th0[i][col] = th[i][col])  $\wedge$ 
    (0  $\leq i < |th| \wedge th_0[i][ord(REGION)] = 1 \rightarrow_L$  th[i][ord(REGION)] = 5)  $\wedge$ 
    (0  $\leq i < |th| \wedge th_0[i][ord(REGION)] \neq 1 \rightarrow_L$  th0[i][ord(REGION)] = th[i][ord(REGION)]))}
}

proc histoGramadeAnillosConcentricos (in th:  $eph_h$ , in centro:  $\mathbb{Z} \times \mathbb{Z}$ , in distancias:  $seq\langle\mathbb{Z}\rangle$ , out result:  $seq\langle\mathbb{Z}\rangle$ ) {
  Pre {esListaOrdenadaCreciente(distancias)}
}

```

```

    ∧ |distancias| > 0
    ∧ noTieneValoresNulosNiNegativos(distancias)}
Post { |res| = |distancias|
    ∧ res[0] = cantidadDeHogaresEntreDosDistancias(th, centro, 0, distancias[0])
    ∧ (∀i : ℤ)(1 ≤ i < |res| →L
    res[i] = cantidadDeHogaresEntreDosDistancias(th, centro, distancias[i - 1], distancias[i])
    }
}

proc quitarIndividuos (inout th: ephh, inout ti: ephi, in busqueda: seq⟨(ItemIndividuo, dato)⟩, out result: (ephh, ephi)) {
    Pre {laEncuestaEsValida(th, ti) ∧ th0 = th ∧ ti0 = ti ∧ busquedaValida(busqueda)}
    Post {result0 ⊆ th0 ∧ result1 ⊆ ti0 ∧ interseccionVacía(result0, th) ∧ interseccionVacía(result1, ti) ∧
        (∀i : ℤ)(0 ≤ i < |result1| →L cumpleBusqueda(result[i], busqueda) ∧
        (∀i : ℤ)(0 ≤ i < |ti| →L ¬cumpleBusqueda(ti[i], busqueda) ∧
        (∀i : ℤ)(0 ≤ i < |result1| →L contieneHogarDeIndividuo(result[i], result0) ∧
        ¬contieneHogarDeIndividuo(result[i], th)
        (∀i : ℤ)(0 ≤ i < |result1| →L contieneHogarDeIndividuo(ti[i], th) ∧ ¬contieneHogarDeIndividuo(ti[i], result0)
        }
    }
}

```

2. Predicados y Auxiliares generales

Al comienzo hay algunos predicados genéricos. En adelante, se separaron los predicados y auxiliares en orden en que fueron apareciendo, pero se reutilizan algunos en ejercicios posteriores.

2.1. Predicados generales

```

pred pertenece (elem: T, s: seq⟨T⟩) {
    (∃i : ℤ)(0 ≤ i < |s| ∧L elem = s[i])
}

pred estaContenido (t : seq⟨seq⟨dato⟩⟩, t0 : seq⟨seq⟨dato⟩⟩) {
    (∀i : ℤ)(0 ≤ i < filas(t) →L ((∃j : ℤ)(0 ≤ j < filas(t0) ∧L (t[i] = t0[j])))
}

```

2.2. Ejercicio 1

```

pred laEncuestaEsValida (th: ephh, ti: ephi) {
    esMatriz(th) ∧ esMatriz(ti) ∧ esTablaNoVacía(th) ∧ esTablaNoVacía(ti) ∧
    cantColumnasHogares(th) ∧ cantColumnasIndividuos(ti) ∧
    aCadaHogarLeCorrespondeUnIndividuo(th, ti) ∧ aCadaIndividuoLeCorrespondeUnHogar(th, ti) ∧
    noHayRepetidos(th, ord(HOGCODUSU)) ∧ noHayRepetidos(ti, ord(INDCODUSU)) ∧
    latitudLongitudValidas(th) ∧ anioTrimestreTabla(th, ord(HOGANO), ord(HOGTRIMESTRE)) ∧
    anioTrimestreTabla(ti, ord(INDANO), ord(INDTRIMESTRE)) ∧ cantMiembrosHogarMenorOIgual20(th, ti) ∧
    atributoIV2MayoroIgualII2(th) ∧ atributosHogarRangoEsperado(th) ∧ atributosIndividuoRangoEsperado(ti)
}

pred esMatriz (m: seq⟨seq⟨dato⟩⟩) {
    (∀i : ℤ)(0 ≤ i < filas(m) →L |m(i)| > 0 ∧ (∀j : ℤ)(0 ≤ j < filas(m) →L |m(i)| = |m(j)|))
}

aux filas (m: seq⟨seq⟨dato⟩⟩) : ℤ = |m|;

aux columnas (m: seq⟨seq⟨dato⟩⟩) : ℤ = if filas(m) > 0 then |m(0)| else 0 fi;

pred esTablaNoVacía (m: seq⟨seq⟨dato⟩⟩) {
    |m| ≠ 0
}

```

```

pred cantColumnasHogares (m: ephh) {
  (∀i : ℤ)(0 ≤ i < columnas(m) →L ItemHogar(i) ≠ ⊥) ∧ (∀i : ℤ)(i ≥ columnas(m) →L
    ItemHogar(i) = ⊥)
}

pred cantColumnasIndividuos (m: ephi) {
  (∀i : ℤ)(0 ≤ i < columnas(m) →L ItemIndividuo(i) ≠ ⊥) ∧ (∀i : ℤ)(i ≥ columnas(m) →L
    ItemIndividuo(i) = ⊥)
}

pred aCadaHogarLeCorrespondeUnIndividuo (m: ephh, z: ephi) {
  (∀i : ℤ)(0 ≤ i < filas(m) →L (∃j : ℤ)(0 ≤ j < filas(z) ∧L z[j][ord(INDCODUSU)] = m[i][ord(HOGCODUSU)]))
}

pred aCadaIndividuoLeCorrespondeUnHogar (m: ephh, z: ephi) {
  (∀j : ℤ)(0 ≤ j < filas(z) →L (∃i : ℤ)(0 ≤ i < filas(m) ∧L z[j][ord(INDCODUSU)] = m[i][ord(HOGCODUSU)]))
}

pred noHayRepetidos (m: seq⟨seq⟨dato⟩⟩, l:ℤ) {
  (∀i : ℤ)(0 ≤ i < filas(m) →L (∀j : ℤ)((0 ≤ j < filas(m) ∧ j ≠ i) →L m[i][l] ≠ m[j][l]))
}

pred latitudLongitudValidas (m: ephh) {
  (∀i : ℤ)(0 ≤ i < filas(m) →L (-55 ≤ m[i][ord(HOGLATITUD)] ≤ -22)) ∧
  (∀j : ℤ)(0 ≤ j < filas(m) →L (-74 ≤ m[j][ord(HOGLONGITUD)] ≤ -53))
}

aux anoEncuesta (m: ephh) : ℤ = m[0][ord(HOGANO)];

aux trimestreEncuesta (m: ephh) : ℤ = m[0][ord(HOGTRIMESTRE)];

pred anioTrimestreTabla (m: seq⟨seq⟨dato⟩⟩, l,r:ℤ) {
  (∀i : ℤ)(0 ≤ i < filas(m) →L m[i][l] = anoEncuesta ∧ m[i][r] = trimestreEncuesta)
}

pred cantMiembrosHogarMenorOIgual20 (m: ephh, z: ephi) {
  (∀i : ℤ)(0 ≤ i < filas(m) →L (∑j=0filas(z)-1 if hayCorrespondenciaHogarIndividuo(i, j, m, z) then 1 else 0 fi ≤ 20))
}

pred atributoIV2MayoroIgualII2 (m: ephh) {
  (∀i : ℤ)(0 ≤ i < filas(m) →L m[i][ord(IV2)] ≥ m[I][ord(II2)])
}

pred atributosHogarRangoEsperado (m: ephh) {
  (∀i : ℤ)(0 ≤ i < filas(m) →L ((0 < m[i][ord(HOGCODUSU)])) ∧ (0 < m[i][ord(HOGANO)])) ∧
  (0 < m[i][ord(HOGTRIMESTRE)]) ∧ (1 ≤ m[i][ord(II7)] ≤ 3) ∧ (1 ≤ m[i][ord(REGION)] ≤ 10) ∧
  (0 ≤ m[i][ord(MAS_500)] ≤ 1) ∧ (1 ≤ m[i][ord(IV1)] ≤ 5) ∧ (1 ≤ m[i][ord(II3)] ≤ 3)))
}

pred atributosIndividuoRangoEsperado (z: ephi) {
  (∀i : ℤ)(0 ≤ i < filas(m) →L ((0 < z[i][ord(INDCODUSU)])) ∧ (0 < z[i][ord(INDANO)])) ∧
  (0 < z[i][ord(INDTRIMESTRE)]) ∧ (0 < z[i][ord(COMPONENTE)]) ∧ (0 ≤ z[i][ord(CH6)])) ∧
  (-1 ≤ z[i][ord(P47T)]) ∧ (1 ≤ z[i][ord(CH4)] ≤ 2) ∧ (0 ≤ z[i][ord(NIVELED)] ≤ 1) ∧
  (-1 ≤ Z[i][ord(ESTADO)] ≤ 1) ∧ (0 ≤ z[i][ord(CATOCUP)] ≤ 4) ∧ (1 ≤ z[i][ord(PP04G)] ≤ 10)))
}

```

2.3. Ejercicio 2

```

aux casasConNHabitaciones (th: ephh, region: ℤ, n: ℤ) : ℤ =
  ∑i=0|th|-1 if (th[i][ord(region)] = region) ∧ esCasa(th[i]) ∧ (th[i][ord(IV2)] = n) then 1 else 0 fi;

```

```

pred laRegionEsValida (region:  $\mathbb{Z}$ ) {
   $0 < region \leq 6$ 
}

pred esMaxDeHabitaciones (th:  $eph_h$ , region:  $\mathbb{Z}$ , n:  $\mathbb{Z}$ ) {
   $(\forall i : \mathbb{Z})((0 \leq i < filas(th) \wedge th[i][ord(REGION)] = region) \longrightarrow_L th[i][ord(IV2)] \leq n)$ 
}

```

2.4. Ejercicio 3

```

aux casasConHacinamientoCriticoEnLaRegion (th:  $eph_h$ , ti:  $eph_i$ , i:  $\mathbb{Z}$ ) :  $\mathbb{Z} =$ 
 $\sum_{j=0}^{|th|-1} \text{if } (th[j][ord(REGION)] = i) \wedge (esCasaConHacinamientoCritico(th, ti, j)) \text{ then } 1 \text{ else } 0 \text{ fi};$ 

```

```

pred esCasaConHacinamientoCritico (th:  $eph_h$ , ti:  $eph_i$ , i:  $\mathbb{Z}$ ) {
   $esCasa(th[i]) \wedge hogarEnCiudadGrande(th[i]) \wedge esHogarConHacinamientoCritico(th, ti, i)$ 
}

```

```

pred esCasa (h: hogar) {
   $h[ord(IV1)] = 1$ 
}

```

```

pred hogarEnCiudadGrande (h: hogar) {
   $h[ord(MAS\_500)] = 1$ 
}

```

```

pred esHogarConHacinamientoCritico (th:  $eph_h$ , ti:  $eph_i$ , i:  $\mathbb{Z}$ ) {
   $\frac{miembrosDelHogar(ti, th[i][ord(CODUSU)])}{cantidadDeHabitaciones(th[i])} > 3$ 
}

```

```

aux miembrosDelHogar (ti:  $eph_i$ , codusu: dato) :  $\mathbb{Z} =$ 
 $\sum_{j=0}^{|ti|-1} \text{if } ti[j][ord(CODUSU)] = codusu \text{ then } 1 \text{ else } 0 \text{ fi};$ 

```

```

aux cantidadDeHabitaciones (h: hogar) :  $\mathbb{Z} =$ 
 $h[ord(IV2)];$ 

```

```

aux hogaresEnLaRegion (th:  $eph_h$ , i:  $\mathbb{Z}$ ) :  $\mathbb{Z} =$ 
 $\sum_{j=0}^{|th|-1} \text{if } th[j][ord(REGION)] = i \text{ then } 1 \text{ else } 0 \text{ fi};$ 

```

2.5. Ejercicio 4

```

pred esCasaODepartamento (h: hogar) {
   $h[ord(IV1)] = 1 \vee h[ord(IV1)] = 2$ 
}

```

```

pred haceTeleworking (h: hogar, i: individuo) {
   $h[ord(II3)] = 1 \wedge i[ord(PP04G)] = 6$ 
}

```

```

aux cantidadPersonasEnHogarEnCiudadGrande (h: hogar, ti:  $eph_i$ ) :  $\mathbb{Z} =$ 
 $\sum_{i=0}^{|ti|-1} \text{if } hogarEnCiudadGrande(h) \wedge h[ord(HOGCODUSU)] = ti[i][ord(INDCODUSU)] \text{ then } 1 \text{ else } 0 \text{ fi} \quad ;$ 

```

```

aux cantidadPersonasQueHacenTeleworkingEnHogarEnCiudadGrande (h: hogar, ti:  $eph_i$ ) :  $\mathbb{Z} =$ 
 $\sum_{i=0}^{|ti|-1} \text{if } hogarEnCiudadGrande(h) \wedge h[ord(HOGCODUSU)] = ti[i][ord(INDCODUSU)] \wedge_L \text{haceTeleworking}(h, ti[i]) \wedge$ 
 $esCasaODepartamento(h) \text{ then } 1 \text{ else } 0 \text{ fi};$ 

```

```

aux cantidadPersonasEnCiudadGrande (th:  $eph_h$ , ti:  $eph_i$ ) :  $\mathbb{Z} =$ 
 $\sum_{i=0}^{|th|-1} \text{cantidadPersonasEnHogarEnCiudadGrande}(th[i], ti) \quad ;$ 

```

aux cantidadPersonasQueHacenTeleworkingEnCiudadGrande (th: eph_h , ti: eph_i) : \mathbb{Z} =
 $\sum_{i=0}^{|th|-1}$ cantidadPersonasQueHacenTeleworkingEnHogarEnCiudadGrande(th[i], ti) ;

2.6. Ejercicio 5

pred tenenciaPropia (h: hogar) {
 $h[ord(II7)] = 1$
}

aux habitacionesParaDormir (h: hogar) : \mathbb{Z} =
 $h[ord(II2)]$;

aux hogaresCandidatosASubsidio (in th: eph_h , in ti: eph_i) : \mathbb{Z} =
 $\sum_{i=0}^{|th|-1}$ if esCasa(th[i]) \wedge tenenciaPropia(th[i]) \wedge (habitacionesParaDormir(th[i]) < (miembrosDelHogar(ti, th[i][ord(HOGCODUSU)] - 2)) then 1 else 0 fi ;

2.7. Ejercicio 6

pred es2-upla (tupla: hogar \times individuo) {
 $(tupla)_0[ord(HOGCODUSU)] = (tupla)_1[ord(INDCODUSU)] \wedge (\forall i : \mathbb{Z})(i \geq 2 \rightarrow_L (tupla)_i = \perp)$
}

2.8. Ejercicio 7

pred ordenadoPorRegion (th: eph_h) {
 $(\forall i : \mathbb{Z})(0 \leq i < filas(th) - 1 \rightarrow_L (th[i][ord(REGION)] \leq th[i+1][ord(REGION)]))$
}

pred regionesOrdenadasPorHogcodusu (th: eph_h) {
 $(\forall i : \mathbb{Z})((0 \leq i < filas(th) - 1 \wedge th[i][ord(REGION)] = th[i+1][ord(REGION)]) \rightarrow_L (th[i][ord(HOGCODUSU)] \leq th[i+1][ord(HOGCODUSU)]))$
}

pred mismoOrdenDeCodusu (th: eph_h , ti: eph_i) {
 $(\forall i : \mathbb{Z})(0 \leq i < filas(th) - 1 \rightarrow_L (th[i][ord(HOGCODUSU)] = ti[i][ord(INDCODUSU)]))$
}

pred casasOrdenadasPorComponente (ti: eph_i) {
 $(\forall i : \mathbb{Z})((0 \leq i < filas(ti) - 1 \wedge ti[i][ord(INDCODUSU)] = ti[i+1][ord(INDCODUSU)]) \rightarrow_L (ti[i][ord(COMPONENTE)] \leq ti[i+1][ord(COMPONENTE)]))$
}

pred mismoTamanoDeTabla (t : seq(seq<dato>), t₀ : seq(seq<dato>)) {
 $filas(t) = filas(t_0) \wedge columnas(t) = columnas(t_0)$
}

2.9. Ejercicio 8

pred estaOrdenadaPorIngresos (in th: eph_h , in ti: eph_i in res: seq(hogar)) {
 $(\forall i : \mathbb{Z})(0 \leq i < |res| - 1 \rightarrow_L$
 $cantidadDeIngresos(ti, res[i][ord(HOGCODUSU)] \leq cantidadDeIngresos(ti, res[i+1][ord(HOGCODUSU)]))$
}

aux cantidadDeIngresos (in ti: eph_i , in codusu : \mathbb{Z}) : \mathbb{Z} =
 $\sum_{i=0}^{|ti|-1}$ if th[i][ord(INDCODUSU)] = codusu \wedge (ti[i][ord(P47T)] \neq -1 then ti[i][ord(P47T)] else 0 fi ;

```

pred diferenciaDeIngresosConstante (in ti:  $eph_i$  in res:  $seq\langle hogar \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |res| - 2 \rightarrow_L$ 
    cantidadDeIngresos( $ti$ ,  $res[i+1][ord(HOGCODUSU)]$ ) - cantidadDeIngresos( $ti$ ,  $res[i][ord(HOGCODUSU)]$ ) =
    cantidadDeIngresos( $ti$ ,  $res[i+2][ord(HOGCODUSU)]$ ) - cantidadDeIngresos( $ti$ ,  $res[i+1][ord(HOGCODUSU)]$ )
  )
}

pred esLaSecuenciaMasGrande (in th:  $eph_h$ , in ti:  $eph_i$ , in res:  $seq\langle hogar \rangle$ ) {
  ( $\forall s : seq\langle hogar \rangle$ )( $(s \subseteq th) \wedge estaOrdenadaPorIngresos(ti, s)$ 
     $\wedge diferenciaDeIngresosConstante(ti, s) \rightarrow |s| \leq |res|$ 
  )
}

```

2.10. Ejercicio 9

No se utilizaron predicados auxiliares

2.11. Ejercicio 10

```

aux cantidadDeHogaresEntreDosDistancias
  (in th:  $eph_h$ , in centro :  $\mathbb{Z} \times \mathbb{Z}$ , in limiteInferior :  $\mathbb{Z}$ , in limiteSuperior:  $\mathbb{Z}$ ) :  $\mathbb{Z} =$ 
   $\sum_{i=0}^{|th|-1} if (limiteInferior \leq$ 
     $distanciaEuclidiana(th[i][ord(HOGLATITUD)], th[i][ord(HOGLONGITUD)], centro_0, centro_1) <$ 
     $limiteSuperior) then 1 else 0 fi$ ;

aux distanciaEuclidiana (in C1Lat:  $\mathbb{Z}$ , in C1Long:  $\mathbb{Z}$ , in C2Lat:  $\mathbb{Z}$ , in C2Long:  $\mathbb{Z}$ ) :  $\mathbb{Z} =$ 
 $\sqrt{(C1Lat - C2Lat)^2 + (C1Long - C2Long)^2}$ ;

pred esListaOrdenadaCreciente (in distancias:  $seq\langle \mathbb{Z} \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |distancias| - 1 \rightarrow_L distancias[i] \leq distancias[i+1]$ )
}

pred noTieneValoresNulosNiNegativos (in distancias:  $seq\langle \mathbb{Z} \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |distancias| \rightarrow_L distancias[i] > 0$ )
}

```

2.12. Ejercicio 11

```

pred busquedaValida (busqueda:  $seq\langle (ItemIndividuo, dato) \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |busqueda| \rightarrow_L atributoIndividuoEnRangoEsperado(busqueda[i]_0, busqueda[i]_1)$ )
}

pred interseccionVacua (s:  $T$ , q:  $seq\langle T \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |s| \rightarrow_L ((\forall j : \mathbb{Z})(0 \leq j < |q| \rightarrow_L s[i] \neq q[j]))$ )
}

pred cumpleBusqueda (ind: individuo, busqueda  $seq\langle (ItemIndividuo, dato) \rangle$ ) {
  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |busqueda| \rightarrow_L ind[busqueda[i]_0] = busqueda[i]_1$ )
}

pred contieneHogarDeIndividuo (ind: individuo, th:  $seq\langle hogar \rangle$ ) {
  ( $\exists i : \mathbb{Z}$ )( $0 \leq i < |th| \wedge_L ind[ord(INDCODUSU)] = th[i][ord(HOGCODUSU)]$ )
}

```

3. Decisiones tomadas

3.1. Generales

Se reemplaza el uso de los predicados ‘pertenece’ y ‘estaContenido’ por sus equivalentes matematicos \in y \subseteq

3.2. Ejercicio 1

Para las coordenadas se decidió utilizar la convención de que la latitud se mide entre -90° y 90° , mientras que la longitud va desde -180° a 180°

3.3. Ejercicio 4

En este ejercicio se decidió que en el numerador, se pondría exclusivamente la cantidad de personas que viven en casa o departamento, que trabajan desde su casa y que a la vez tienen un ambiente dedicado para el trabajo, además de vivir en una gran ciudad, mientras que en el denominador, solo se consideran los individuos en hogares en una gran ciudad.

3.4. Ejercicio 10

Se agregó el chequeo de que las distancias contempladas no sean negativas, ya que no tendría sentido hablar de distancias negativas.

Al considerar las distancias, en cada iteración, se decidió incluir el límite inferior, pero no el superior, dejando afuera las casas que puedan estar a distancia $\text{distancias}(\text{---distancias---} - 1)$ del centro