



TP de Especificación

Análisis Habitacional Argentino

27 de Octubre 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]))  $\wedge$ 
    hayCorrespondenciaEntreJoinyTabla(ti, junta, ord(INDCODUSU), 1)  $\wedge$ 
    hayCorrespondenciaEntreJoinyTabla(th, junta, ord(HOGCODUSU), 0)}
}

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 { $esListaOrdenadaCreciente(distancias)$ 
     $\wedge |distancias| > 0$ 
     $\wedge noTieneValoresNulosNiNegativos(distancias)$ }
  Post { $|res| = |distancias|$ 
     $\wedge res[0] = cantidadDeHogaresEntreDosDistancias(th, centro, 0, distancias[0])$ 
     $\wedge (\forall i : \mathbb{Z})(1 \leq i < |res| \rightarrow_L res[i] = cantidadDeHogaresEntreDosDistancias(th, centro, distancias[i - 1], distancias[i])$ 
    }
}

proc quitarIndividuos (inout th:  $eph_h$ , inout ti:  $eph_i$ , in busqueda:  $seq\langle (ItemIndividuo, dato) \rangle$ , out result:  $(eph_h, eph_i)$ ) {
  Pre { $laEncuestaEsValida(th, ti) \wedge th_0 = th \wedge ti_0 = ti \wedge busquedaValida(busqueda)$ }
  Post { $result_0 \subseteq th_0 \wedge result_1 \subseteq ti_0 \wedge interseccionVacía(result_0, th) \wedge interseccionVacía(result, ti) \wedge$ 
     $(\forall i : \mathbb{Z})(0 \leq i < |result_1| \rightarrow_L cumpleBusqueda(result[i], busqueda) \wedge$ 
     $(\forall i : \mathbb{Z})(0 \leq i < |ti| \rightarrow_L \neg cumpleBusqueda(ti[i], busqueda) \wedge$ 
     $(\forall i : \mathbb{Z})(0 \leq i < |result_1| \rightarrow_L contieneHogarDeIndividuo(result[i], result_0) \wedge$ 
     $\neg contieneHogarDeIndividuo(result[i], th)$ 
     $(\forall i : \mathbb{Z})(0 \leq i < |result_1| \rightarrow_L contieneHogarDeIndividuo(ti[i], th) \wedge \neg contieneHogarDeIndividuo(ti[i], result_0)$ 
    }
}

```

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:  $seq\langle T \rangle$ ) {
   $(\exists i : \mathbb{Z})(0 \leq i < |s| \wedge_L elem = s[i])$ 
}

pred estaContenido (t:  $seq\langle seq\langle dato \rangle \rangle$ , t0:  $seq\langle seq\langle dato \rangle \rangle$ ) {
   $(\forall i : \mathbb{Z})(0 \leq i < filas(t) \rightarrow_L ((\exists j : \mathbb{Z})(0 \leq j < filas(t_0) \wedge_L (t[i] = t_0[j])))$ 
}

```

2.2. Ejercicio 1

```

pred laEncuestaEsValida (th:  $eph_h$ , ti:  $eph_i$ ) {
   $esMatriz(th) \wedge esMatriz(ti) \wedge esTablaNoVacía(th) \wedge esTablaNoVacía(ti) \wedge$ 
   $cantColumnasHogares(th) \wedge cantColumnasIndividuos(ti) \wedge$ 
   $aCadaHogarLeCorrespondeUnIndividuo(th, ti) \wedge aCadaIndividuoLeCorrespondeUnHogar(th, ti) \wedge$ 
   $noHayRepetidos(th, ord(HOGCODUSU)) \wedge noHayRepetidos(ti, ord(INDCODUSU)) \wedge$ 
   $latitudLongitudValidas(th) \wedge anioTrimestreTabla(th, ord(HOGANO), ord(HOGTRIMESTRE)) \wedge$ 
   $anioTrimestreTabla(ti, ord(INDANO), ord(INDTRIMESTRE)) \wedge cantMiembrosHogarMenorOIgual20(th, ti) \wedge$ 
   $atributoIV2MayorOIgualII2(th) \wedge atributosHogarRangoEsperado(th) \wedge atributosIndividuoRangoEsperado(ti)$ 
}

pred esMatriz (m:  $seq\langle seq\langle dato \rangle \rangle$ ) {
   $(\forall i : \mathbb{Z})(0 \leq i < filas(m) \rightarrow_L |m(i)| > 0 \wedge (\forall j : \mathbb{Z})(0 \leq j < filas(m) \rightarrow_L |m(i)| = |m(j)|))$ 
}

aux filas (m:  $seq\langle seq\langle dato \rangle \rangle$ ) :  $\mathbb{Z} = |m|$ ;

aux columnas (m:  $seq\langle seq\langle dato \rangle \rangle$ ) :  $\mathbb{Z} = \text{if } filas(m) > 0 \text{ then } |m(0)| \text{ else } 0 \text{ fi}$ ;

pred esTablaNoVacía (m:  $seq\langle seq\langle dato \rangle \rangle$ ) {
   $|m| \neq 0$ 
}

```

```

pred cantColumnasHogares (m: ephh) {
  def(ItemHogar(columnas(m) - 1)) ∧ ¬def(ItemHogar(columnas(m)))
}

pred cantColumnasIndividuos (z: ephi) {
  def(ItemIndividuo(columnas(z) - 1)) ∧ ¬def(ItemIndividuo(columnas(z)))
}

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) ∧ (0 ≤ m[i][ord(IV2)])) ∧
  (0 ≤ m[i][ord(II2)])))
}

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} =
 $\sum_{i=0}^{|th|-1} \text{cantidadPersonasQueHacenTeleworkingEnHogarEnCiudadGrande}(th[i], ti)$;

2.6. Ejercicio 5

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

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

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

2.7. Ejercicio 6

pred es2-upla (tupla: hogar \times individuo) {
 $(tupla)_0[\text{ord}(HOGCODUSU)] = (tupla)_1[\text{ord}(INDCODUSU)] \wedge \neg def((tupla)_2)$
}

pred hayCorrespondenciaEntreJoinyTabla (m: $seq\langle seq\langle dato \rangle \rangle$, z: $joinHI$, k, n: \mathbb{Z}) {
 $(\forall i : \mathbb{Z})(0 \leq i < filas(m) \longrightarrow_L (\exists j : \mathbb{Z})(0 \leq j < filas(z) \wedge_L m[i][k] = (z[j])_n[k]) \wedge$
 $(\forall j : \mathbb{Z})(0 \leq j < filas(z) \longrightarrow_L (\exists j : \mathbb{Z})(0 \leq j < filas(m) \wedge_L m[i][k] = (z[j])_n[k]))$
}

2.8. Ejercicio 7

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

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

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

pred mismoOrdenDeCodusu (th: eph_h , ti: eph_i) {
 $(\forall i : \mathbb{Z})(0 \leq i < filas(ti) - 1 \longrightarrow_L \neg((ti[i][\text{ord}(INDCODUSU)] = ti[i+1][\text{ord}(INDCODUSU)]) \leftrightarrow$
 $(th[\text{posicionHogarCorrespondiente}(th, ti[i][\text{ord}(INDCODUSU)])] =$
 $th[\text{posicionHogarCorrespondiente}(th, ti[i+1][\text{ord}(INDCODUSU)] - 1)))$
}

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

pred mismoTamanoDeTabla (t: $seq\langle seq\langle dato \rangle \rangle$, $t_0 : seq\langle seq\langle dato \rangle \rangle$) {
 $filas(t) = filas(t_0) \wedge columnas(t) = columnas(t_0)$
}

2.9. Ejercicio 8

```

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

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

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 . Ambos predicados fueron agregados en la sección de predicados, bajo el título de ”Predicados Generales”(sección 2.1).

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 limite inferior, pero no el superior, dejando afuera las casas que puedan estar a distancia $\text{distancias}(\text{---distancias---} - 1)$ del centro.