TP de Especificación

Sudoku

1 de Abril de 2017 Algoritmos y Estructuras de Datos I

Grupo 10

Integrante	LU	Correo electrónico
Gomez Salaverri, Francisco	550/15	`francisco@gomezsalaverri.com`
Matias Colque, Nadia Noemí	188/17	`nmatias@dc.uba.ar`
Girón, Jorge David	637/16	`jorgedavid2905@gmail.com`

1. Problemas

1. `proc sudoku_esTableroValido` (**in t**: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, **out result**: Bool) {

   Pre {True}
   Post {$tableroValido(t) = $ **result**}
   `pred tableroValido` (t: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$) {
      $esFilaValida(t) \wedge$ esColumnaValida(t)
      }
   `pred esFilaValida` (t: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$) {
      $(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})enRango(t, i) \wedge_L$
      enRango(t[i],j) $\wedge_L$ length(t[i]) $= 9 \longrightarrow_L 0 \le t[i][j] \le 9$
      }
   `pred esColumnaValida` (t: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$) {
      $(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})length(t) = 9 \wedge$ enRango(t,i) $\wedge_L$
      enRango(t[i],j) $\longrightarrow_L 0 \le t[i][j] \le 9$
      }
   }

2. `proc sudoku_esCeldaVacia` (**in t**: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, **in f**: $\mathbb{Z}$, **in c**: $\mathbb{Z}$, **out result**: Bool) {

   Pre {$tableroValido(t) \wedge$ esFilaYColumnaValida(f,c)}
   Post {**result** $= celdaVacia(f, c)$}
   `pred celdaVacia` (t: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, i: $\mathbb{Z}$, j: $\mathbb{Z}$) {$s[i][j] = 0$}
   }

3. `proc sudoku_nroDeCeldasVacias` (**in t**: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, **out result** : $\mathbb{Z}$) {

   Pre {$tableroValido(t)$}
   Post {**result** $= nroCeldasVacias(t)$}
   `fun nroCeldasVacias` (s: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$) : $\mathbb{Z} =$
      $(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})enRango(s, i) \wedge_L$ enRango(s[i],j) $\longrightarrow_L$
      $\sum$ if $celdaVacia(s, i, j)$ then 1 else 0 fi **;**
   }

4. `proc sudoku_primeraCeldaVaciaFila` (**in t**: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, **out result** : $\mathbb{Z}$) {

   Pre {$tableroValido(t)$}
   Post {if $celdasVacias(t) = 0$ then -1 else $(\exists i : \mathbb{Z})(\exists j : \mathbb{Z})$**result** $= i \wedge$ enRango(t,i) $\wedge_L$ enRango(t[i],j) $\wedge_L$
      celdaVacia(t,i,j) $\wedge$ menorFilaVacia (t,i) $\wedge$ menorColumnaDeLaFilaVacia(t,i,j)
   `pred menorFilaVacia` (**t:** $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, **i:** $\mathbb{Z}$) {
      $(\forall f : \mathbb{Z})(\forall g : \mathbb{Z})enRango(t, f) \wedge_L$ enRango(t[f],g)
      $\longrightarrow_L$ celdaVacia(t,f,g) $\wedge$ f $\ge i$)}
   `pred menorColumnaDeLaFilaVacia` (**t:** $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, **i:** $\mathbb{Z}$, **j:** $\mathbb{Z}$) {
      $(\forall g : \mathbb{Z})enRango(t[i], g)$
      $\longrightarrow_L$ celdaVacia(t,i,g) $\wedge$ g $\ge j$)}
    fi }
   }

5. `proc sudoku_primeraCeldaVaciaColumna` (**in t**: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, **out result** : $\mathbb{Z}$) {

   Pre {$tableroValido(t)$}
   Post {if $celdasVacias(t) = 0$ then -1 else $(\exists i : \mathbb{Z})(\exists j : \mathbb{Z})$**result** $= j \wedge$ enRango(t,i) $\wedge_L$ enRango(t[i],j) $\wedge_L$
      celdaVacia(t,i,j) $\wedge$ menorFilaVacia (t,i) $\wedge$ menorColumnaDeLaFilaVacia(t,i,j) fi }
   }

6. `proc sudoku_valorEnCelda` (**in t**: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$, **in f**: $\mathbb{Z}$, **in c**: $\mathbb{Z}$, **out result**: Bool) {

   Pre {$tableroValido(t) \wedge$
      esFilaYColumnaValida(f,c) $\wedge$
      celdaVacia(t[f][c]) $=$ false
      }
   Post {**result** $= t[f][c]$}
   }

7. `proc sudoku_llenarCelda` (**inout t**: $seq\langle seq\langle \mathbb{Z} \rangle\rangle$ **in f**: $\mathbb{Z}$, **in c**: $\mathbb{Z}$, **in value**: $\mathbb{Z}$) {

   Pre {$tableroValido(t) \wedge$
      esFilaYColumnaValida(f,c) $\wedge$

$1 \leq value \leq 9 \wedge$
t = t$_0 \wedge$
$t_0[f][c] = 0\}$
Post $\{t = SetAt(t_0[f], c, value)\}$
}

8.   proc sudoku_vaciarCelda (**inout t:** $seq\langle seq\langle \mathbb{Z}\rangle\rangle$, **in f:** $\mathbb{Z}$, **in c:** $\mathbb{Z}$, **out result:** Bool) {

Pre $\{tableroValido(t)\wedge$
esFilaYColumnaValida(f,c) $\wedge$
t = t$_0\}$
Post $\{$**result** $= (t[f][c] \neq 0)\wedge$
t $=$ SetAt(t$_0[f], c, 0)$

}
}

9.   proc sudoku_esTableroParcialmenteResuelto (**in t:** $seq\langle seq\langle \mathbb{Z}\rangle\rangle$, **out result:** Bool) {

Pre {True}
Post $\{$**result** $= TableroParcialmenteResuelto(t)\}$
pred noHayRepetidos (s: $seq\langle \mathbb{Z}\rangle$) {
   $(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})((enRango(s,i)\wedge$ enRango(s, j) $\wedge$ j $\neq$ i$))\longrightarrow_L$ ((s[i] $= 0 \wedge$ s[j]=0) $\vee_L$ s[i] $\neq$ s[j])
}
pred TableroConElementosDelCeroalNueve (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
   $(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})((enRango(t,i)\wedge$ enRango(t[i], j))$\wedge_L$ $(0 \leq t[i][j] \leq 9)$
}
pred FiladeTableroParcialmenteResuelto (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
   $TableroConElementosDelCeroalNueve(t)\wedge$
   $(\forall i : \mathbb{Z})(enRango(t,i)\longrightarrow_L$ noHayRepetidos(t[i])
}
pred ColumnadeTableroParcialmenteResuelto (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
   $TableroConElementosDelCeroalNueve(t)\wedge$
   $(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})(\forall h : \mathbb{Z})((enRango(t,i)\wedge$ enRango(t, j) $\wedge$ enRango(t[i], h)$\wedge$ i $\neq$ j$)\longrightarrow_L$ (t[i][h] $\neq$ t[j][h])
}
pred regiondeTableroParcialmenteResuelto (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
   $(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})((enRango(t,i)\wedge$ i mod 3 $= 0)$ $\wedge_L$ (enRango(t[i], j) $\wedge$ j mod 3 $= 0)$
   $\longrightarrow_L$ (s = Concat(subseq(s[i], j, j+3), subseq(s[i+1], j, j+3), subseq(s[i+2], j, j+3))$\wedge$
   noHayRepetidos(s))
}
pred TableroParcialmenteResuelto (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
   $TableroValido(t)\wedge$
   filadeTableroParcialmenteResuelto(t) $\wedge$
   columnadeTableroParcialmenteResuelto(t) $\wedge$
   regiondeTableroParcialmenteResuelto(t)
}
}

10.   proc sudoku_esTableroTotalmenteResuelto (**in t:** $seq\langle seq\langle \mathbb{Z}\rangle\rangle$, **out result:** Bool) {

Pre $\{tableroValido(t)\}$
Post $\{$**result** $= tableroTotalmenteResuelto(t)\}$
pred tableroTotalmenteResuelto (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
   $nroCeldasVacias(t) = 0\wedge$
   tableroParcialmenteResuelto(t) }
}

11.   proc sudoku_esSubTablero (**in** $t_0, t_1 : seq\langle seq\langle \mathbb{Z}\rangle\rangle$, **out result** : Bool){

Pre $\{tableroValido(t_0), tableroValido(t_1)\}$
Post $\{result = esSub(t_0, t_1)\}$
pred esSub (**t**$_0 : seq\langle seq\langle \mathbb{Z}\rangle\rangle, t_1 : seq\langle seq\langle \mathbb{Z}\rangle\rangle$){
   $(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})length(t_0) = length(t_1)\wedge_L$
   enRengo(t$_0, i)\wedge_L$ **length(t**$_0[i]) = length(t_1[i])\wedge$
   enRengo(t$_0[i], j)\longrightarrow_L$ **t**$_0[i][f] \neq 0\wedge$ **t**$_0[i][f] = t_1[i][f]$
}

}

12. proc sudoku_tieneSolucion (**in t**: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$, **out tienesolucion**: Bool) {

   Pre $\{tableroValido(t)\}$
   Post $\{\textbf{tienesolucion} = (solucion(t) \neq t)\}$

fun solucion (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) : $seq\langle seq\langle \mathbb{Z}\rangle\rangle$ =
   $(\exists x : seq\langle seq\langle \mathbb{Z}\rangle\rangle)$if $esSub(t,x)\wedge_L$ tableroTotalmenteResuleto(x) then x else t fi }

13. proc sudoku_resolver (**inout t**: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$, **out tienesolucion**: Bool) {

   Pre $\{t_0 = t\}$
   Post $\{\textbf{tienesolucion} = (esTableroValido(t)\wedge (solucion(t_0) \neq t_0))\wedge t = solucion(t_0)\}$

   ;
}

14. proc sudoku_copiarTablero (**in src**: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$, **out target**: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {

   Pre $\{$True$\}$
   Post $\{src = target\}$
}

2. Predicados y Auxiliares generales

   pred Nombre (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {True}
   pred PredLargo (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
$(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})$True
}
   fun Aux (i: $\mathbb{Z}$) : Bool = True;
   pred esFilaYColumnaValida (i: $\mathbb{Z}$, j: $\mathbb{Z}$) $\{0 \leq i,j \leq 8\}$
   pred enRango (t: $seq\langle t\rangle$, i:$\mathbb{Z}$) {
$0 \leq i < length(t)$
}

3. Decisiones tomadas