DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

TP de Especificación

Sudoku

21 de Abril de 2017   Algoritmos y Estructuras de Datos I

Grupo 17

Integrante	LU	Correo electrónico
Maqueda, Ignacio	279/14	ignaciomaqueda95@gmail.com
Parral, Guillermo	280/16	guillermoeparral@gmail.com
Quintela, Gonzalo	089/16	gquintela@dc.uba.ar
Sirio, Tomás	440/16	tomassirio@gmail.com

1. Problemas

```
proc  sudoku_esTableroValido (in t: seq⟨seq⟨ℤ⟩⟩, out result: Bool)  {
        Pre {True}
        Post {result = esTableroValido(t)}
}
```

```
proc sudoku_esCeldaVacia (in t: seq⟨seq⟨ℤ⟩⟩, in f: ℤ,in c: ℤ, out result: Bool)  {
        Pre {esTableroValido(t) ∧_L 0 ≤ f, c < |t|}
        Post {result = (t[f][c] = 0)}
}
```

```
proc sudoku_nroDeCeldasVacias (in t: seq⟨seq⟨ℤ⟩⟩, out result: ℤ)  {
        Pre {esTableroValido(t)}
        Post {∑_{i=0}^{|t|-1}(∑_{j=0}^{|t|-1} if t[i][j] = 0 then 1 else 0 fi)}
}
```

```
proc sudoku_primeraCeldaVaciaFila (in t: seq⟨seq⟨ℤ⟩⟩, out result: ℤ)  {
        Pre {esTableroValido(t)}
        Post {if (∃i : ℤ)(0 ≤ i < |t| ∧_L filaTieneCeldaVacia(t[i]) ∧_L (∀j : ℤ)
           (0 ≤ j < i ⟶_L ¬filaTieneCeldaVacia(t[j])))
           then result = i
           else result = −1 fi}
}
```

```
proc sudoku_primeraCeldaVaciaColumna (in t: seq⟨seq⟨ℤ⟩⟩, out result: ℤ)  {
        Pre {esTableroValido(t)}
        Post {if (∃i : ℤ)(0 ≤ i < |t| ∧_L filaTieneCeldaVacia(t[i]) ∧_L (∀j : ℤ)(0 ≤ j < i ⟶_L ¬filaTieneCeldaVacia(t[j])))
           then result = indicePrimeraCeldaVaciaEnFila(t[i])
           else result = −1 fi}
}
```

```
proc sudoku_valorEnCelda (in t: seq⟨seq⟨ℤ⟩⟩, in f: ℤ, in c: ℤ, out result: ℤ)  {
        Pre {esTableroValido(t) ∧_L 0 ≤ f, c ≤ 8 ∧_L t[i][j] ≠ 0}
        Post {result = t[f][c]}
}
```

```
proc sudoku_llenarCelda (inout t: seq⟨seq⟨ℤ⟩⟩, in f: ℤ, in c: ℤ, in value: ℤ)  {
        Pre {esTableroValido(t) ∧_L 0 ≤ f, c ≤ 8 ∧_L t[i][j] = 0 ∧_L 1 ≤ value ≤ 9 ∧_L t = t_0}
        Post {t[f][c] = value ∧_L (∀i : ℤ)(∀j : ℤ)(0 ≤ i, j < |t| ∧_L (i ≠ f ∨ j ≠ c)) ⟶_L t[i][j] = t_0[i][j]}
}
```

```
proc sudoku_vaciarCelda (inout t: seq⟨seq⟨ℤ⟩⟩, in f: ℤ, in c: ℤ)  {
        Pre {esTableroValido(t) ∧_L 0 ≤ f, c ≤ 8 ∧_L t[i][j] ≠ 0 ∧_L t = t_0}
        Post {t[f][c] = 0 ∧_L (∀i : ℤ)(∀j : ℤ)((0 ≤ i, j < |t| ∧_L (i ≠ f ∨ j ≠ c)) ⟶_L t[i][j] = t_0[i][j])}
}
```

```
proc sudoku_esTableroParcialmenteResuelto (in t: seq⟨seq⟨ℤ⟩⟩, out result: Bool))  {
        Pre {True}
        Post {result = esTableroParcialmenteResuelto(t)}
}
```

```
proc sudoku_esTableroTotalmenteResuelto (in t: seq⟨seq⟨ℤ⟩⟩, out result: Bool)  {
```

Pre $\{true\}$
        Post $\{esParcialmenteResuelto(t) \wedge_L \forall i : \mathbb{Z})(0 \leq i < |t| \longrightarrow_L \neg filaTieneCeldaVacia(t[i])\}$
}

proc sudoku_esTablero (in $t_0, t_1 : seq\langle seq\langle \mathbb{Z}\rangle\rangle$, out $result$: Bool){
        Pre $\{true\}$
        Post $\{esSubtablero(t_0, t_1) = result\}$
}

proc sudoku_esSubTablero (in $t_0, t_1 : seq\langle seq\langle \mathbb{Z}\rangle\rangle$, out $result$: Bool){
        Pre $\{true\}$
        Post $\{esSubtablero(t_0, t_1) = result\}$
}

2. Predicados y Auxiliares generales

pred esMatriz (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
$(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})(0 \leq i, j < |t| \longrightarrow_L |t[i]| = |t[j]|)$
}

fun cantidadFilas (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) : $\mathbb{Z} = |t|$;

fun cantidadColumnas (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) : $\mathbb{Z} =$ if $cantidadFilas(t) > 0$ then $|t[0]|$ else 0 fi ;

pred esMatrizCuadrada (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) {
$esMatriz(t) \wedge (cantidadFilas(t) = cantidadColumnas(t))$
}

pred esTableroValido (t: $seq\langle seq\langle \mathbb{Z}\rangle\rangle$) $\{esMatrizCuadrada(t) \wedge_L |t| = 9 \wedge_L$
$(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})(0 \leq i, j < |t| \longrightarrow_L 0 \leq t[i][j] \leq 9)$
}

pred filaTieneCeldaVacia (f: $seq\langle \mathbb{Z}\rangle$) {
$(\exists i : \mathbb{Z})(0 \leq i < |f| \wedge_L f[i] = 0)$
}

fun indicePrimeraCeldaVaciaEnFila (s: $seq\langle \mathbb{Z}\rangle$) : $\mathbb{Z} =$ if $((\exists : i\mathbb{Z})(0 \leq i < |s| \wedge_L s[i] = 0 \wedge_L$
$(\forall j : \mathbb{Z})(0 \leq j < i \longrightarrow_L s[j] \neq 0)))$ then i else -1 fi ;

pred noHayRepetidosEnRegion (s: $seq\langle \mathbb{Z}\rangle$) $\{(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})(\forall k : \mathbb{Z})(\forall l : \mathbb{Z})$
$(0 \leq i, j, k, l < 9 \wedge_L (i div 3 = k div 3) \wedge_L (j div 3 = l div 3) \wedge_L (i \neq k \vee j \neq l) \longrightarrow_L t[i][j] \neq t[k][l]$
}

pred noHayRepetidosEnFila (s: $\mathbb{Z}$) $\{(\forall i : \mathbb{Z})(\forall j : \mathbb{Z})(0 \leq i, j < |s| \wedge_L j \neq i \longrightarrow_L s[i] \neq s[j])$
}

pred noHayRepetidosEnColumna (t: $\mathbb{Z}$) $\{(\forall j : \mathbb{Z})(0 \leq j < |t| \longrightarrow_L (\forall l : \mathbb{Z})(\forall k : \mathbb{Z})$
$(0 \leq l, k < |t| \wedge_L l \neq k \longrightarrow_L t[i][j] \neq t[k][j]))$
}

pred esTableroParcialmenteResuelto (t: $\mathbb{Z}$) $\{esTableroValido(t) \wedge_L (\forall i : \mathbb{Z})(0 \leq i < |t| \longrightarrow_L$
$(noHayRepetidosEnFila(t[i]) \wedge_L noHayRepetidosEnColumna(t) \wedge_L noHayRepetidosEnReligion(t))$
}

pred esSubTablero ($t_0, t_1 : seq\langle seq\langle \mathbb{Z}\rangle\rangle$)$\{(esTableroValido(t_0) \wedge_L esTableroValido(t_1)) \wedge_L ((\forall i : \mathbb{Z})(\forall j : \mathbb{Z})$
$(0 \leq i|t| \wedge_L t_0[i][j] \neq 0) \longrightarrow_L (t_0[i][j] = t_1[i][j]))$
}

3. Decisiones tomadas