DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

# Trabajo Práctico Especificación

**Buscaminas**

23 de septiembre de 2022 — Algoritmos y Estructuras de Datos I


| Integrante | LU | Correo electrónico |
|---|---|---|
| Santiago García | 627/21 | atsantiagogarcia@gmail.com |
| Felipe Saidon | 1436/21 | felipesaidon@gmail.com |
| Matías Daniel Díaz Sarmiento | 704/19 | mdds.2017@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

# Índice

# 1. Definiciones de tipos

Para nuestra especificación utilizaremos los siguientes renombres de tipos:
type *pos*= $\mathbb{Z}\times\mathbb{Z}$

Identificador de una posición en el tablero (fila, columna). Ambos indices comienzan en cero. La fila avanza de arriba hacia abajo y la columna de izquierda a derecha (ver figura 2)
type *tablero*= seq⟨seq⟨Bool⟩⟩

Matriz con valores booleanos indicando las posiciones de las minas. Cada posici´on contiene el valor **true** si hay una mina y el valor **false** si no la hay. Cada elemento de la secuencia representa una fila, o sea que si t es de tipo tablero, t[0] es la primera fila, t[1] la segunda, etc. Las filas se cuentan de arriba hacia abajo. A su vez los elementos de cada fila se cuentan de izquierda a derecha, por lo que en t[0][0] se encuentra la posición (0, 0) y, en general, en t[i][j] se encuentra la posición (i, j) (ver figura 2).
type *jugadas*= seq⟨pos ×$\mathbb{Z}$>

Secuencia de casillas jugadas. Incluye solo las posiciones de las casillas descubiertas e indica, para una determinada posición, el número de minas adyacentes.
type *banderitas* = seq⟨pos⟩

Secuencia con las posiciones en las que el jugador puso una bandera porque considera que hay una mina (ayudamemoria). El orden de los elementos de la secuencia no es importante.

# 2. Resolución de ejercicios

## 2.1. Ejercicio 1

`aux minasAdyacentes` (t: tablero, p: pos) : $\mathbb{Z} = \sum_{i=p_0-1}^{p_0+1} ( \sum_{j=p_1-1}^{p_1+1} (\textit{if hayMinaAdyacente(i,j,t) then 1 else 0 fi}))$;

```
pred hayMinaAdyacente (i:Z,j: Z, t: tablero) {
    0 ≤ i < |t| ∧ 0 ≤ j < |t[i]| ∧_L t[i][j] = true
}
```

## 2.2. Ejercicio 2

```
pred juegoValido (t: tablero, j:jugadas) {
    tableroValido(t)∧|j| ≤ |t|∧(∀i : Z)(0 ≤ i < |j| →_L (0 ≤ (j[i]_0)_0 < |t|∧0 ≤ (j[i]_0)_1 < |t|)∧_L j[i]_1 = minasAdyacentes(t, j[i]_0)∧
    #apariciones(j[i], j) = 1)))
}
pred tableroValido (t:tablero) {
    (∀i : Z)(0 ≤ i < |t| →_L |t[i]| = |t|)
}
```

`aux #apariciones` (e:T,s:$seq\langle T\rangle$) : $\mathbb{Z} = \sum_{i=0}^{|s|-1} (\textit{if s[i]=e then 1 else 0 fi})$;

## 2.3. Ejercicio 3

```
proc plantarBanderita (in t: tablero, in j: jugadas, in p: pos, inout b: banderitas) {
      Pre {b = B_0 ∧ juegoValido(t, j) ∧ noHayBanderitasDescubiertasNiRepetidas(B_0, j) ∧
      sePuedePonerBanderitaEnPosicion(p, B_0, j, t)}
      Post {|b| = |B_0| + 1 ∧ noHayBanderitasDescubiertasNiRepetidas(b, j) ∧ posicionMarcada(p, b)}
}
pred posicionValidaParaJugar (p:pos,t:tablero, j:jugada) {
    posicionValida(p, t) ∧ posicionNoJugada(p, j)
}
pred posicionValida (p:pos, t:tablero) {
    0 ≤ p_0 < |t| ∧ 0 ≤ p_1 < |t|
}
pred posicionJugada (p:pos, j:jugada) {
    (∃i : Z)(0 ≤ i < |j| ∧_L j[i]_0 = p)
}
pred posicionNoJugada (p:pos, j:jugada) {
    ¬(∃i : Z)(0 ≤ i < |j| ∧_L j[i]_0 = p)
```

}
pred sePuedePonerBanderitaEnPosicion (p: pos, $B_0 : banderitas, j : jugadas, t : tablero$){
$posicionNoMarcada(p, B_0) \wedge_L posicionValidaParaJugar(p, t, j)$
}
pred noHayBanderitasDescubiertasNiRepetidas ($B_0 : banderitas, j : jugadas$){
$(\forall i : \mathbb{Z})(0 \leq i < |B_0| \rightarrow_L posicionNoJugada(B_0[i], j) \wedge sinRepetidos(B_0))$
}
pred posicionNoMarcada (p:pos, $B_0 : banderitas$){
$p \notin B_0$
}
pred posicionMarcada (p:pos, $B_0 : banderitas$){
$p \in B_0$
}
pred sinRepetidos (s:$seq\langle T \rangle$) {
$(\forall i : \mathbb{Z})(0 \leq i < |s| \longrightarrow_L \#apariciones(s[i], s) = 1)$
}

## 2.4. Ejercicio 4

proc perdio (in t:tablero, in j:jugadas, out res:Bool) {
Pre $\{juegoValido(t, j)\}$
Post $\{res = true \Leftrightarrow tocoUnaMina(j, t)\}$
}
pred tocoUnaMina (j:jugadas,t:tablero) {
$(\exists i : \mathbb{Z})(0 \leq i < |j| \wedge_L t[j[i]_{00}][j[i]_{01}] = \text{true})$
}

## 2.5. Ejercicio 5

proc gano (in t:tablero, in j:jugadas, out res:Bool) {
Pre $\{juegoValido(t, j)\}$
Post $\{res = true \Leftrightarrow esquivoTodasLasMinas(j, t)\}$
}
pred esquivoTodasLasMinas (j:jugadas,t:tablero) {
$\neg(\exists i, k : \mathbb{Z})((0 \leq i, k < |t| \wedge t[i][k] = \text{false}) \wedge_L (i, k) \notin j) \wedge (\forall i, k : \mathbb{Z})((0 \leq i, k < |t| \wedge t[i][k] = \text{true}) \longrightarrow_L posicionNoJugada((i, k), j))$
}

## 2.6. Ejercicio 6

proc jugar (in t: tablero, in b: banderitas, in p: pos, inout j: jugadas) {
Pre $\{j = J_0 \wedge juegoValido(j, J_0) \wedge \neg tocoUnaMina(J_0, t) \wedge posicionValidaParaJugar(p, t, J_0) \wedge posicionNoMarcada(p, b) \wedge$
$noHayBanderitasDescubiertasNiRepetidas(b, J_0) \wedge hayAlMenosUnCasilleroSinMinaSinJugar(J_0, t)\}$
Post $\{|j| = |J_0| + 1 \wedge posicionJugada(p, j) \wedge juegoValido(t, j) \wedge posicionNoMarcada(p, b) \wedge$
$seMantienenPosicionesJugadas(j, J_0) \wedge noHayBanderitasDescubiertasNiRepetidas(b, j)\}$
}
pred seMantienenPosicionesJugadas (j,$J_0 : jugadas$){
$(\forall e : (pos \times \mathbb{Z}))(e \in J_0 \longrightarrow_L e \in j)$
}
pred hayAlMenosUnCasilleroSinMinaSinJugar (j:jugadas,t:tablero) {
$(\exists p : pos)(posicionValida(p, t) \wedge posicionNoJugada(p, j) \wedge t[p_0][p_1] = \text{false})$
}

## 2.7. Ejercicio 7

pred caminoLibre (t: tablero, $p_0 : pos, p_1 : pos$){
$minasAdyacentes(t, p_1) \geq 1 \wedge minasAdyacentes(t, p_0) = 0 \wedge (\forall$ i, j: $\mathbb{Z})($ ( $((p_0)_0 < i < (p_1)_0 \vee (p_1)_0 < i < (p_1)_0) \wedge ((p_0)_1 <$
$j < (p_1)_1 \vee (p_1)_1 < j < (p_0)_1)) \longrightarrow_L minasAdyacentes(t, (i, j)) = 0)$

}

## 2.8. Ejercicio 8

proc JugarPlus (in t: tablero, in b: banderitas, in p: pos, inout j: jugadas) {
Pre $\{j = J_0 \wedge juegoValido(t, J_0) \wedge \neg tocoUnaMina(J_0, t) \wedge noHayBanderitasDescubiertasNiRepetidas(b, j) \wedge posicionValidaParaJugar(p, t, J_0) \wedge posicionNoMarcada(p, b) \wedge hayAlMenosUnCasilleroSinMinaSinJugar(J_0, t)\}$
Post $\{|j| > |J_0| \wedge posicionJugada(p, j) \wedge posicionNoMarcada(p, b) \wedge noHayBanderitasDescubiertasNiRepetidas(b, j) \wedge seDescubrenCasillerosSinMinasAdyacentes(t, p, j) \wedge juegoValido(t, j) \wedge seMantienenPosicionesJugadas(j, J_0)\}$
}

pred seDescubrenCasillerosSinMinasAdyacentes (t:tablero, p: pos, j: jugadas) {
$(\exists p' : pos)(caminoLibre(t, p, p') \wedge_L (\forall i, k : \mathbb{Z})(((p_0 < i < p'_0 \vee p'_0 < i < p_0) \wedge (p_1 < k < p'_1 \vee p'_1 < k < p_1) \wedge minasAdyacentes(t, (i, k)) = 0 \rightarrow_L posicionJugada((i, k), j))))$
}

## 2.9. Ejercicio 9

proc sugerirAutomatico121 (in t: tablero, in b: banderitas, in j: jugadas, out p: pos) {
Pre $\{juegoValido(t, j) \wedge \neg tocoUnaMina(j, t) \wedge noHayBanderitasDescubiertasNiRepetidas(b, j) \wedge hay121(t, j)\}$
Post $\{posicionValida(p, t) \wedge posicionValidaParaJugar(p, t, j) \wedge posicionNoMarcada(p, b) \wedge (devuelvePosicionCorrectaHorizontal(p, t) \vee devuelvePosicionCorrectaVertical(p, t))\}$
}

pred hay121 (t:tablero, j: jugadas) {
$(\exists i, k : \mathbb{Z})((posicionValida((i, k), t) \wedge_L posicionJugada((i, k), j) \wedge minasAdyacentes(t, (i, k)) = 2) \wedge_L (hay121Vertical((i, k), t, j) \vee_L hay121Horizontal((i, k), t, j)))$
}

pred hay121horizontal (p':pos,t:tablero,j:jugadas) {
$1 \leq p'_1 < |t| - 1 \wedge_L minasAdyacentes(t, (p'_0, p'_1 + 1)) = minasAdyacentes(t, (p'_0, p'_1 - 1)) = 1$
}

pred hay121vertical (p':pos,t:tablero,j:jugadas) {
$1 \leq p'_0 < |t| - 1 \wedge_L minasAdyacentes(t, (p'_0 + 1, p'_1)) = minasAdyacentes(t, (p'_0 - 1, p'_1)) = 1$
}

pred devuelvePosicionCorrectaHorizontal (p:pos,t:tablero) {
$(hay121horizontal((p_0 - 1, p_1), t, j) \longrightarrow_L minasAdyacentes(t, (p_0 - 1, p_1)) = 2) \vee (hay121horizontal((p_0 + 1, p_1), t, j) \longrightarrow_L minasAdyacentes(t, (p_0 + 1, p_1)) = 2)$
}

pred devuelvePosicionCorrectaVertical (p:pos,t:tablero) {
$(hay121vertical((p_0, p_1 - 1), t, j) \longrightarrow_L minasAdyacentes(t, (p_0, p_1 - 1)) = 2) \vee (hay121vertical((p_0, p_1 + 1), t, j) \longrightarrow_L minasAdyacentes(t, (p_0, p_1 + 1))) = 2$
}