



TP de Especificación

BUSCAMINAS

Septiembre 2022

Algoritmos y Estructuras de Datos I

Algorítmicos

Integrante	LU	Correo electrónico
Gonzalez, Alvaro	233/22	alvarogonzalezc4@gmail.com
Lista, Melanie	516/21	melaalista@gmail.com
Rivero, Barbara	1206/22	barbara.m.rivero@gmail.com
Ruz, Luciano	589/22	luciruzveloso@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 Definición de Tipos

type *pos* = $\mathbb{Z} \times \mathbb{Z}$

Identificador de una posición en el tablero (fila, columna). Ambos índices comienzan en cero. La fila avanza de arriba hacia abajo y la columna de izquierda a derecha.

type *tablero* = $seq(seq\langle Bool \rangle)$

Indica si el estado del juego representado por *j* es válido y se corresponde con la ubicación real de las minas representada por un tablero *t*.

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* = $seq(pos)$

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

2 Problemas

2.1 Ejercicio 1

aux minasAdyacentes (t: tablero, p: pos) : \mathbb{Z} =

$$\sum_{i=p_0-1}^{p_0+1} \sum_{j=p_1-1}^{p_1+1} (\text{if } (0 \leq p_0 < |t| \wedge 0 \leq p_1 < |t| \wedge p \neq (i, j) \wedge_L t[i][j]) \text{ then } 1 \text{ else } 0 \text{ fi});$$

2.2 Ejercicio 2

pred juegoValido (t: tablero, j: jugadas) {

$$jugadasValidas(t, j) \wedge coincidenMinasAdy(t, j) \wedge cantJugadasConMinas(t, j) \leq 1 \wedge noSeRepiten(j)$$

}

pred jugadasValidas (t: tablero, j: jugadas) {

$$(\forall i : \mathbb{Z})(0 \leq i < |j| \longrightarrow 0 \leq j[i]_{00}, j[i]_{01} < |t|)$$

}

pred coincidenMinasAdy (t: tablero, j: jugadas) {

$$(\forall i : \mathbb{Z})(0 \leq i < |j| \longrightarrow_L minasAdyacentes(t, (j[i]_{00}, j[i]_{01}) = j[i]_1)$$

}

aux cantJugadasConMinas (t: tablero, j: jugadas) : \mathbb{Z} =

$$\sum_{i=0}^{|j|} (\text{if } t[j[i]_{00}][j[i]_{01}] \text{ then } 1 \text{ else } 0 \text{ fi});$$

pred noSeRepiten (j: jugadas) {

$$(\forall i : \mathbb{Z})(0 \leq i < |j| \longrightarrow_L \neg(\exists k : \mathbb{Z})(0 \leq k < |j| \wedge i \neq j) \wedge_L j[i]_0 = j[k]_0))$$

}

2.3 Ejercicio 3

proc plantarBanderita (in t: tablero, in j: jugadas, in p: pos, inout b: banderitas) {

$$\text{Pre } \{ |t| > 1 \wedge juegoValido(t, j) \wedge (0 \leq p_0 < |t| \wedge 0 \leq p_1 < |t|) \wedge noHayBanderasJugadas(j, b) \\ \wedge p \notin b \wedge (p, minasAdyacentes(t, p)) \notin j \wedge b = b_0 \}$$

$$\text{Post } \{ p \in b \wedge |b_0| = |b| - 1 \wedge (\forall i : \mathbb{Z})(0 \leq i < |b_0| \longrightarrow_L b_0[i] \in b) \}$$

}

pred noHayBanderasJugadas (j: jugadas, b: banderitas) {

$$(\forall i : \mathbb{Z})(0 \leq i < |b| \longrightarrow_L (b[i], minasAdyacentes(t, b[i])) \notin jugadas)$$

}

2.4 Ejercicio 4

proc perdió (in t: tablero, in j: jugadas, out res: Bool) {

$$\text{Pre } \{ juegoValido(t, j) \}$$

$$\text{Post } \{ res = true \iff cantJugadasConMinas(t, j) = 1 \}$$

}

2.5 Ejercicio 5

```

proc ganó (in t:tablero, in j:jugadas, out res: Bool) {
  Pre {juegoValido(t,j)}
  Post {res = true  $\iff$  esUnTableroTerminado(t,j)  $\wedge$  cantJugadasConMinas(t,j) = 0}
}

pred esUnTableroTerminado (t:tablero , j: jugadas) {

  ( $\forall i : \mathbb{Z}$ )( $\forall k : \mathbb{Z}$ )( $(0 \leq i < |t| \wedge 0 \leq k < |t|) \wedge_L t[i][k] = false$ )  $\longrightarrow_L$  ( $\exists h : \mathbb{Z}$ )( $(i, k) = (j[h]_{00}, j[h]_{01})$ )
}

```

2.6 Ejercicio 6

```

proc jugar (in t: tablero, in b: banderitas, in p: pos, inout j: jugadas) {
  Pre {tableroValido(t)  $\wedge$  juegoValido(t,j)  $\wedge$  cantJugadasConMinas(t,j) = 0  $\wedge$   $\neg$ esUnTableroTerminado(t,j)  $\wedge$ 
  noSeHaJugado(p,j)  $\wedge$  noTieneBandera(p,b)  $\wedge$  j = j0}
  Post {(p, minasAdyacentes(t,p))  $\in$  j  $\wedge$  |j| = |j0| + 1  $\wedge$  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |j_0| \longrightarrow_L j_0[i] \in j$ )}
}

pred tableroValido (in t: tablero) {
  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |t| \longrightarrow_L |t| = |t[i]| = |t[0]|$ )
}

pred noSeHaJugado (in p: pos, in j: jugadas) {
   $\neg$ ( $\exists i : \mathbb{Z}$ )( $0 \leq i < |j| \wedge p = (j[i]_{00}, j[i]_{01})$ )
}

pred noTieneBandera (in p: pos, in b: banderitas) {
   $\neg$ ( $\exists i : \mathbb{Z}$ )( $0 \leq i < |b| \wedge p = b[i]$ )
}

```

2.7 Ejercicio 7.

```

pred caminoLibre (t: tablero , p : pos , q : pos) {
  ( $\exists s : jugadas$ )( $s[0]_0 = p \wedge s[|s| - 1]_0 = q \wedge minasAdyacentes(t,q) \geq 1$ )
   $\wedge$  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |s| - 1 \longrightarrow_L (sonAdyacentes(s[i]_0, s[i + 1]_0) \wedge minasAdyacentes(t, s[i]_0) = 0)$ )
}

pred sonAdyacentes (p : pos , q : pos) {
  ( $\exists i : \mathbb{Z}$ )( $\exists j : \mathbb{Z}$ )( $(p_0 - 1 \leq i \leq p_0 + 1 \wedge (p_1 - 1 \leq j \leq p_1 + 1 \neq p) \wedge_L q = (i, j))$ )
}

```

2.8 Ejercicio 8*.

```

proc jugarPlus (in t: tablero, in b: banderitas, in p: pos, inout j: jugadas) {
  Pre {tableroValido(t) ∧ juegoValido(t, j) ∧ cantJugadasConMinas(t, j) = 0 ∧ ¬esUnTableroTerminado(t, j) ∧
    noSeHaJugado(p, j) ∧ noTieneBandera(p, b) ∧ minasAdyacentes(t, p) = 0 ∧ j = j0}
  Post {(∀u : jugadas)esCaminoLibre(u, t, p) → ((∀i : Z)(0 ≤ i < |u|) →L u[i] ∈ j))
    ∧ ¬(∃k : pos)¬caminoLibre(t, p, k) ∧ (k, minasAdyacentes(t, k)) ∈ j
    ∧ (∀h : Z)(0 ≤ h < |j0| →L j0[h] ∈ j)}
}

pred esCaminoLibre (u: jugadas, t: tablero, p: pos) {
  u[0]0 = p ∧ minasAdyacentes(t, p) = 0 ∧ minasAdyacentes(t, u[|u| - 1]0) ≥ 1
  ∧ (∀i : Z)(0 ≤ i < |u| - 1 →L sonAdyacentes(u[i]0, u[i + 1]0))
}

```

2.9 Ejercicio 9.

```

proc sugerirAutomatico121 (in t: tablero, in b: banderitas, in j: jugadas, out p: pos) {
  Pre {tableroValido(t) ∧ juegoValido(t, j) ∧ cantJugadasConMinas(t, j) = 0 ∧ ¬esUnTableroTerminado(t, j) ∧
    noSeHaJugado(p, j) ∧ hayPatron(j)}
  Post {(∃p, q, r : pos)(p, 1) ∈ j ∧ (q, 2) ∈ j ∧ (r, 1) ∈ j
    ∧ ((sonHorizontales(p, q, r) ∧ lasDeArribaDescubiertas(j, q) ∧ res = (q0 + 1, q1))
    ∨ (sonHorizontales(p, q, r) ∧ lasDeAbajoDescubiertas(j, q) ∧ res = (q0 - 1, q1))
    ∨ ((sonVerticales(p, q, r) ∧ lasDeIzquierdaDescubiertas(j, q) ∧ res = (q0, q1 + 1))
    ∨ (sonVerticales(p, q, r) ∧ lasDeDerechaDescubiertas(j, q) ∧ res = (q0, q1 - 1))))}
}

pred sonHorizontales (p: pos , q : pos , r : pos) {
  (p = (q0, q1 - 1) ∨ p = (q0, q1 + 1)) ∧ (r = (q0, q1 - 1) ∨ r = (q0, q1 + 1)) ∧ p ≠ r
}

pred sonVerticales (p: pos , q : pos , r : pos) {
  (p = (q0 - 1, q1) ∨ p = (q0 + 1, q1)) ∧ (r = (q0 - 1, q1) ∨ r = (q0 + 1, q1)) ∧ p ≠ r
}

pred sonRectos (p: pos , q : pos , r : pos) {
  sonHorizontales(p, q, r) ∨ sonVerticales(p, q, r)
}

pred hayPatron (j: jugadas) {
  (∃p, q, r : pos)(pertenecenAJ121(p, q, r) ∧ sonRectos(p, q, r) ∧ algunCostadoDescubierto(j, p, q, r))
}

pred pertenecenAJ121 (p: pos , q: pos , r: pos , j: jugadas) {
  (p, 1) ∈ j ∧ (q, 2) ∈ j ∧ (r, 1) ∈ j
}

pred algunCostadoDescubierto (j: jugadas, p: pos, q: pos, r: pos) {
  sonHorizontales(p, q, r) → (lasDeArribaDescubiertas(j, q) ∨ lasDeAbajoDescubiertas(j, q))
  ∨ sonVerticales(p, q, r) → (lasDeIzquierdaDescubiertas(j, q) ∨ lasDeDerechaDescubiertas(j, q))
}

```

```

pred lasDeArribaDescubiertas (j: jugadas , q: pos) {
     $(q_0 - 1, q_1 - 1, \text{minasAdyacentes}(q_0 - 1, q_1 - 1)) \in j$ 
     $\wedge (q_0 - 1, q_1, \text{minasAdyacentes}(q_0 - 1, q_1)) \in j$ 
     $\wedge (q_0 - 1, q_1 + 1, \text{minasAdyacentes}(q_0 - 1, q_1 + 1)) \in j$ 
}

pred lasDeAbajoDescubiertas (j: jugadas , q: pos) {
     $(q_0 + 1, q_1 - 1, \text{minasAdyacentes}(q_0 + 1, q_1 - 1)) \in j$ 
     $\wedge (q_0 + 1, q_1, \text{minasAdyacentes}(q_0 + 1, q_1)) \in j$ 
     $\wedge (q_0 + 1, q_1 + 1, \text{minasAdyacentes}(q_0 + 1, q_1 + 1)) \in j$ 
}

pred lasDeIzquierdaDescubiertas (j: jugadas , q: pos) {
     $(q_0 - 1, q_1 - 1, \text{minasAdyacentes}(q_0 - 1, q_1 - 1)) \in j$ 
     $\wedge (q_0, q_1 - 1, \text{minasAdyacentes}(q_0, q_1 - 1)) \in j$ 
     $\wedge (q_0 + 1, q_1 - 1, \text{minasAdyacentes}(q_0 + 1, q_1 - 1)) \in j$ 
}

pred lasDeIzquierdaDescubiertas (j: jugadas , q: pos) {
     $(q_0 - 1, q_1 + 1, \text{minasAdyacentes}(q_0 - 1, q_1 + 1)) \in j$ 
     $\wedge (q_0, q_1 + 1, \text{minasAdyacentes}(q_0, q_1 + 1)) \in j$ 
     $\wedge (q_0 + 1, q_1 + 1, \text{minasAdyacentes}(q_0 + 1, q_1 + 1)) \in j$ 
}

```

3 Anexo: Comentarios

* No se consideró que el tablero fuera cuadrado en los primeros 5 ejercicios. Ya desde el ejercicio 6 se vuelve condición necesaria para una correcta especificación.

* Para hacerlo más fiel al juego real, en el ejercicio 8 se descubrirán todos los caminos libres posibles desde la posición, inclusive los del borde que sí tienen minas adyacentes. Además sólo se hará cuando la posición inicial no tenga minas adyacentes.