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
Tewrence Da Cruz	619/21	tewrence.cruz@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.

1.1. Ejercicio 1

`aux minasAdyacentes` (t: tablero, p: pos) : $\mathbb{Z} = \sum_{i=pos_0-1}^{pos0+1} (\sum i = pos_1 - 1^{pos_1+1}(\textit{If hayMinaAdyacente(i,j,t) then 1 else 0 fi}))$;

`pred hayMinaAdyacente` (i,j: $\mathbb{Z}$, t: tablero, p: pos) {
$(i \neq p_0 \wedge i \geq 0) \wedge (j \neq p_1 \wedge j \geq 0) \wedge_L t[i][j] = true$
}

1.2. Ejercicio 2

`pred juegoValido` (t: tablero, p: pos) {
$|j| \leq |t| \wedge (\forall i : \mathbb{Z})(0 \leq i < |j| \rightarrow_L (0 \leq j[i]_0 0, j[i]_0 1 < |t|) \wedge_L t[j[i]_0 0][j[i]_0 1] = false \wedge_L j[i]_1 = minasAdyacentes(t, j[i]_0) \wedge$
$apariciones(j[i], j) = 1)) \wedge tableroValido(t)$
}
`pred tableroValido` (t:tablero) {
$(\forall i : \mathbb{Z})(0 \leq |t| \rightarrow_L |t[i]| = |t|)$
}

1.3. Ejercicio 3

`proc plantarBanderita` (in t: tablero, in j: jugadas, in p: pos inout b: banderitas) {
`Pre` {$b = B_0 \wedge juegoValido(t, j) \wedge noHayBanderitasDescubiertasNiRepetidas(B_0, j) \wedge sePuedePonerBanderitaEnPosicion(p, l$
`Post` {|b| = |B$_0$— ∧noHayBanderitasDescubiertasNiRepetidas(b,j)}
}

`pred posicionValidaParaJugar` (p:pos,t:tablero, j:jugada) {
$posicionValida(p, t) \wedge posicionNoJugada(p, j)$
}
`pred posicionValida` (p:pos, t:tablero) {
$0 \leq p_0, p_1 < |t|$
}
`pred posicionNoJugada` (p:pos, j:jugada) {
$\not(\exists i : \mathbb{Z})(0 \leq i < |j| / yLuego j[i]_0 = p)$
}
`pred sePuedePonerBanderitaEnPosicion` (p: pos, B$_0$: $banderitas, j : jugadas, t : tablero$){
$posicionNoMarcada(p, B_0) \wedge_L posicionValidaParaJugar(p, t, j)$
}
`pred noHayBanderitasDescubiertasNiRepetidas` (B$_0$: $banderitas, j : jugadas$){
$(\forall i : \mathbb{Z})(0 \leq i < |B_0| \rightarrow_L posicionNoJugada(B_0[i], j) \wedge sinRepetidos(B_0))$
}

1.4. 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) {
$(\exists i : \mathbb{Z})(0 \leq i < |j| \wedge_L t[j[i]_{00}][j[i]_{01}] = \text{true})$
}

1.5. 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) {
$\neg(\exists i : \mathbb{Z}) \wedge \neg(\exists k : \mathbb{Z})((0 \leq i < |t| \wedge 0 \leq k < |t| \wedge t[i][k] = \text{false}) \wedge_L posicionJugada((i, k), j)) \wedge ((\forall i : \mathbb{Z}) \wedge (\forall k : \mathbb{Z})(0 \leq i < |t| \wedge 0 \leq k < |t| \wedge t[i][k] = \text{true}) \longrightarrow_L posicionJugada((i, k), j))$
}
pred posicionJugada (p:pos,j:jugadas) {
$\neg posicionNoJugada(p, j)$
}

1.6. Ejercicio 6

proc jugar (in t: tablero, in b: banderitas, in p: pos, inout j: jugadas) {
Pre $\{j = J_0 \wedge juegoValido(j, J_0) \wedge \neg tocoUnaMina(J_0, t) \wedge posicionValidaParaJugar(p, t, J_0) \wedge posicionNoMarcada(p, b) \wedge noHayBanderitasDescubiertasNiRepetidas(b, J_0)\}$
Post $\{|j| = |J_0+1| \wedge posicionJugada(p, j) \wedge juegoValido(t, j) \wedge posicionNoMarcada(p, b) \wedge noHayBanderitasDescubiertasNiRep$
}

1.7. Ejercicio 7

pred caminoLibre (t: tablero, $p_0 : pos, p_1 : pos$){
$minasAdyacentes(t, p_1) \geq 1 \wedge minasAdyacentes(t, p_0) = 0 \wedge (\exists$ i, j: $\mathbb{Z})(((p_{00}$ ¡i ¡$p_{10} \vee p_{10} < i < p_{10}) \wedge (p_{01} < j < p_{11} \vee p_{11} < j < p_{01}) \wedge minasAdyacentes(t, (i, j)) = 0)$
}

1.8. Ejercicio 8

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

pred seDescubrenCasillerosSinMinasAdyacentes (t:tablero, p: pos, j: jugadas) {
$(\exists p - 1 : pos)(caminoLibre(t, p, p - 1) \wedge_L (\forall i, k : \mathbb{Z})(((p_0 < i < p - 1_0 \vee p - 1_0 < i < p_0) \wedge (p_1 < j < p - 1_1 \vee p - 1_1 < j < p_1) \wedge minasAdyacentes(t, (i, k)) = 0 \rightarrow_L posicionJugada((i, k), j))))$
}

1.9. Ejercicio 9

proc sugerirAutomatico121 (in t: tablero, in b: banderitas, in j: jugadas, out p: pos) {
Pre {$juegoValido(t, j) \land \neg tocoUnaMina(j, t) \land noHayBanderitasDescubiertasNiRepetidas(b, j) \land hay121(t, j)$}
Post {$t[\mathrm{p}_0][p_1] = \text{false} \land$posicionValidaParaJugar(p,t,j) $\land$posicionNoMarcada(p,b) $\land$((hay121Vertical((p_0,p_1-1),t,j) $\lor hay121Ver$
1),t,j)) $\lor$($hay121horizontal(\mathrm{p}_0 - 1, p_1)$,t,j) $\lor hay121horizontal(\mathrm{p}_0 + 1, p_1)$,t,j))}
}
pred hay121 (t:tablero, j: jugadas) {
$(\exists i, k : \mathbb{Z})((posicionValida((i, k), t) \land_L posicionJugada((i, k), j) \land minasAdyacentes(t, (i, k)) = 2) \land_L (hay121Vertical((i, k), t, j) \lor_L$
$hay121Horizontal((i, k), t, j))$
}
pred hay121Vertical ((i,k):pos, t:tablero, j: jugadas) {
$((posicionJugada((i-1, k), j) \land minasAdyacentes(t, (i-1, k)) = 1) \land (posicionJugada((i+1, k), t) \land minasAdyacentes(t, (i+$
$1, k)) = 1)) \land_L ((((posicionJugada((i - 1, k - 1), j) \land posicionJugada((i, k - 1), j) \land posicionJugada((i + 1, k - 1), j)) \land_L$
$(posicionValida((i-1, k+1), t) \land posicionValida((i+1, k+1), t) \land t[i-1][k+1] = t[i+1][k+1] = true \land posicionValida((i, k+$
$1), t) \land t[i][k + 1] = false)) \lor_L (\neg(posicionValida((i - 1, k - 1), t) \land posicionValida((i, k - 1), t) \land posicionValida((i +$
$1, k - 1), t)) \land_L (posicionValida((i - 1, k + 1), t) \land posicionValida((i + 1, k + 1), t) \land t[i - 1][k + 1] = t[i + 1][k + 1] =$
$true \land posicionValida((i, k + 1), t) \land t[i][k + 1] = false)))$
$\lor_L$
(((posicionJugada((i-1,k+1),j) $\land$posicionJugada((i,k+1),j) $\land$posicionJugada((i+1,k+1),j)) $\land_L (posicionValida((i-1, k-1), t) \land$
$posicionValida((i + 1, k - 1), t) \land t[i - 1][k - 1] = t[i + 1][k - 1] = true \land posicionValida((i, k - 1), t) \land t[i][k - 1] = false)) \lor_L$
$(\neg(posicionValida((i-1, k+1), t) \land posicionValida((i, k+1), t) \land posicionValida((i+1, k+1), t)) \land_L (posicionValida((i-1, k-$
$1), t) \land posicionValida((i+1, k-1), t) \land t[i-1][k-1] = t[i+1][k-1] = true \land posicionValida((i, k-1), t) \land t[i][k-1] = false)))$
}
pred hay121Horizontal ((i,k): pos, t:tablero, j: jugada) {
$((posicionJugada((i, k+1), j) \land minasAdyacentes(t, (i, k+1)) = 1) \land (posicionJugada((i, k-1), j) \land minasAdyacentes(t, (i, k-$
$1)) = 1) \land_L$
$((((posicionJugada((i - 1, k - 1), j) \land posicionJugada((i + 1, k), j) \land posicionJugada((i + 1, k + 1), j)) \land_L$
$(posicionValida(i-1, k-1), t) \land posicionValida((i-1, k+1), t) \land t[i-1][k-1] = t[i-1][k+1] = true \land posicionValida((i-$
$1, k), t) \land t[i - 1][k] = false)) \lor_L (\neg(posicionValida((i + 1, k - 1), t) \land posicionValida((i + 1, k), t) \land posicionJugada((i +$
$1, k + 1), j)) \land_L$
$(posicionValida((i - 1, k), t) \land t[i - 1][k] = false)) \lor_L (((posicionJugada((i - 1, k - 1), j) \land posicionJugada((i - 1, k), j) \land$
$posicionJugada((i - 1, k + 1), j)) \land_L$
$(posicionValida((i + 1, k), t) \land t[i + 1][k] = false)) \lor_L (\neg(posicionValida((i - 1, k - 1), t) \land posicionValida((i - 1, k), t) \land$
$posicionValida((i - 1, k + 1), t)) \land_L$
$(posicionValida((i + 1, k - 1), t) \land posicionValida((i + 1, k + 1), t) \land posicionValida((i + 1, k + 1), t) \land t[i + 1][k - 1] =$
$t[i + 1][k + 1] = true \land posicionValida((i + 1, k), t) \land t[i + 1][k] = false))))$
}