Algoritmos y Estructuras de Datos

Primer Parcial – Viernes 27 de septiembre de 2024

Libreta	Apellido y Nombre	E1	E2	E3	E4	Nota Final
752/23	HOERTH FEDERICO	33	20	35	5	93

ELÍAS

- Es posible tener una hoja (2 carillas) con los anotaciones que se deseen, más los dos apuntes (Especificación y TADs)
- Cada ejercicio debe entregarse en **hojas separadas**
- Incluir en cada hoja el número de libreta, número de hoja, apellido y nombre
- El parcial se aprueba con 50 puntos, y se necesita un promedio de 60 entre los dos parciales para aprobar la materia

Excelente

E1. TADs y especificación de problemas [35 pts]

Se quiere modelar una biblioteca en la cuál los libros se organizan en estantes. Cada estante puede alojar una cierta cantidad de libros. Al abrir la biblioteca, se indica la cantidad de estantes que tiene y cuántos libros puede alojar cada uno (cada estante puede alojar una cantidad diferente de libros). A partir de ese momento, no pueden agregarse ni eliminarse estantes.

Los libros se identifican por un número (llamado ISBN) que es único. Cuando ingresa un libro a la biblioteca, se indica en qué estante se va a guardar, y el libro se ubica en la primera posición libre de dicho estante. Cuando se retira un libro de la biblioteca, se indica el número de estante y la posición. En dicho estante, todos los libros a la derecha del que se retiró se corren a la izquierda para tapar la posición que queda libre.

Se desea conocer, para una determinada posición de un estante en particular, qué libro se encuentra guardado en esa posición.

Ejemplo:

Estado de la biblioteca (en un cierto momento):

estante 0	111	2864	6946				
estante 1							
estante 2	12423	23453	12	5467			

Luego de agregar el libro 975 al estante 0:

estante 0	111	2864	6946	975			
estante 1							
estante 2	12423	23453	12	5467			

Luego de sacar el libro de la posición 1 del estante 2:

estante 0	111	2864	6946	975			
estante 1							
estante 2	12423	12	5467				

a) Indique las operaciones (procs) del TAD con todos sus parámetros.
b) Describa el TAD en forma completa, indicando sus observadores, los requiere y asegura de las operaciones. Puede agregar los predicados y funciones auxiliares que necesite, con su correspondiente definición

E2. Precondición más débil [20 pts]

Dado el siguiente condicional determinar la precondición más débil que permite hacer valer la poscondición (Q) propuesta.

```
if (a mod 3 = 0)
    a := abs(a) + 1
else
    a := abs(a) * 3
endif
```

$Q \equiv (\exists j : \mathbb{Z})(j \geq 0 \wedge j \times 3 = a)$

a) Escriba la WP esperada y justifique brevemente en castellano
b) Derivarla formalmente a partir de los axiomas de precondición más débil. Para obtener el puntaje máximo deberá simplificarla lo más posible.

FEDERICO HOERTH
752/23
Hoja: 1

ISBN es $\mathbb{Z}$

TAD Biblioteca {

obs estantes: seq<seq<ISBN>>

obs dimensiones: seq<$\mathbb{Z}$>

pred estanteValido(s: seq<seq<ISBN>>, e: $\mathbb{Z}$) {$0 \leq e < |s|$}

~~pred tieneLibroEnPos(s: seq<seq<ISBN>>, e: $\mathbb{Z}$, [illegible]~~

~~estanteValido(s, e) $\wedge_L$ posicionValida(s, e, p) [illegible]~~

pred posicionValido(s: seq<seq<ISBN>>, e: $\mathbb{Z}$, p: $\mathbb{Z}$) {

estanteValido(s, e) $\wedge_L$ $0 \leq p < |s[e]|$

}

~~pred tieneLugar(s: seq<seq<ISBN>>, [illegible], d: seq<$\mathbb{Z}$>~~ {

pred tieneLugar(s: seq<seq<ISBN>>, d: seq<$\mathbb{Z}$>, e: $\mathbb{Z}$) {

estanteValido(s, e) $\wedge_L$ $0 \leq e < |d|$ $\wedge_L$ $|s[e]| < d[e]$

}

pred estaLibroEnEstantes(s: seq<seq<$\mathbb{Z}$>>, id: ISBN) {

~~[illegible]~~

$(\exists i, j : \mathbb{Z})(0 \leq i < |s| \wedge_L 0 \leq j < |s[i]| \wedge_L s[i][j] = id)$

}

```
proc crearBiblioteca (dimensiones: seq⟨ℤ⟩): Biblioteca
    requiere {|estantes| > 0}
    requiere {(∀j:ℤ)(0 ≤ j < |estantes| → estantes[j] > 0)}
    asegura {|res.estantes| = |res.dimensiones| = |dimensiones|}
    asegura {res.dimensiones = dimensiones}
    asegura {(∀j:ℤ)(0 ≤ j < |estantes| → res.estantes[j] = ⟨⟩)}
```

```
proc guardarLibro(inout b: Biblioteca, d: ISBN, e: ℤ)
    requiere {B₀ = B}
    requiere {estanteValido(B₀.estantes, e)}
    requiere {¬estaLibroEnEstante(B₀.estantes, id)}
    requiere {tieneLugar(B₀.estantes, B₀.dimensiones, e)}
    asegura {b.dimensiones = B₀.dimensiones}
    asegura {|b.estantes| = |B₀.estantes|}
    asegura {(∀j:ℤ)(0 ≤ j < |B₀.estantes| ∧ j ≠ e) → B₀.estantes[j]
        = b.estantes[j])}
    asegura {b.estantes[j] = B₀.estantes[j] ++ ⟨id⟩}
```

FEDERICO HOERTH, 752/23 HOJA 2

proc retirarLibro (inout b: Biblioteca, e: $\mathbb{Z}$, p: $\mathbb{Z}$) {

requiere $\{b = B_0\}$

requiere $\{estanteValido(B_0.estantes, e)\}$

requiere $\{posicionValido(B_0.estantes, e, p)\}$

~~requiere {tieneLugar(B0.estantes~~

~~asegura {B0 ... B0.dimensiones = ... dimensiones~~

asegura $\{b.dimensiones = B_0.dimensiones\}$

asegura $\{|B_0.estantes| = |b.estantes|\}$

asegura $\{(\forall j: \mathbb{Z})((0 \leq j < |B_0.estantes| \land j \neq e) \to_L$
$b.estantes[j] = B_0.estantes[j])\}$

asegura $\{(\forall j: \mathbb{Z})(0 \leq j < p \to_L b.estantes[e][j] = B_0.estantes[e][j])\}$

asegura $\{(\forall j: \mathbb{Z})(p < j < |B_0.estantes[e]| \to_L$
$b.estantes[e][j-1] = B_0.estantes[e][j]$

)} deberías pedir además que el tamaño sea -1 que en bo.

proc libroEnPosicion (in b: Biblioteca, e: $\mathbb{Z}$, p: $\mathbb{Z}$): ISBN

requiere $\{estanteValido(B_0.estantes, e)\}$

requiere $\{posicionValido(B_0.estantes, e, p)\}$

~~requiere~~

asegura { ~~res = b.estantes~~

res = b.estantes[e][p]

}

FEDERICO HUERTA, 752/23 HOJA 3

2. Pre condición Mas débil

$S_1 \equiv (a := abs(a)+1)$ $S_2 \equiv (a := abs(a)*3)$ $B \equiv (a \bmod 3 = 0)$

$def(B) \equiv def(a \bmod 3 = 0) \equiv def(a) \wedge def(3)$
$\equiv True.$

a) Qué nos dice Q?

Que $a \geq 0$ y que $a \bmod 3 = 0$, si se cumple esto va a existir el j.

Pero vemos que si lo guarda se cumple $a = ab(a)+1$ nunca puede ser mult. de 3.

Por ende la wp de la rama del if es False.

Si no se cumple lo guarda $j = abs(a)$ cumple que $j \geq 0$ y $j*3 = abs(a)*3$, entones la wp de la ramo del else: $\neg B$.

B) Calculemos la wp:

$wp(S_1, Q) \equiv wp(a := abs(a)+1, Q) \equiv def(abs(a)+1) \wedge_L Q^{a}_{abs(a)+1}$

$\equiv (def(a) \wedge def(1)) \wedge_L Q^{a}_{abs(a+1)} \equiv True \wedge True \wedge Q^{a}_{abs(a)+1}$

$\equiv (\exists j : \mathbb{Z})(j \geq 0 \wedge j*3 = abs(a)+1)$

$B \wedge wp(S_1, Q) \equiv a \bmod 3 = 0 \wedge (\exists j : \mathbb{Z})(j \geq 0 \wedge j*3 = abs(a)+1)$

Sabemos q como $a \bmod 3 = 0 \Rightarrow \frac{a \bmod 3}{3} \in \mathbb{Z}$

Pero $\frac{1}{3} \notin \mathbb{Z}$. $\left(j = \frac{abs(a)+1}{3} = \frac{abs(a)}{3} + \frac{1}{3}\right)$

Es decir, no existe un j que cumplo con esas condiciones

$\therefore B \wedge wp(S_1, Q) \equiv False$

Calculo $wp(S_2, Q) \equiv wp(a := abs(a)*3, Q) \equiv$

$def(abs(a)*3) \wedge_L Q^{a}_{abs(a)*3} \equiv True \wedge_L Q^{a}_{abs(a)*3}$

$\equiv (\exists j : \mathbb{Z})(j \geq 0 \wedge j*3 = abs(a)*3$

~~$B \wedge wp(S_2, Q) \equiv a \bmod 3 \neq 0 \wedge (\exists j : \mathbb{Z})(j \geq 0 \wedge j = abs(a)*3)$~~

$\neg B \wedge wp(S_2, Q) \equiv a \bmod 3 \neq 0 \wedge (\exists j : \mathbb{Z})(j \geq 0 \wedge j*3 = abs(a)*3)$

Como vimos antes, un posible $j = abs(a)$

Con lo cual el cuantif es True

$\therefore \neg B \wedge wp(S_2, Q) \equiv a \bmod 3 \neq 0 \wedge True \equiv a \bmod 3 \neq 0$

Conclusión:

$wp(S, Q) \equiv def(B) \wedge ((B \wedge wp(S_1, Q)) \vee (\neg B \wedge wp(S_2, Q))$

$\equiv True \wedge (False \vee a \bmod 3 \neq 0)$

$\equiv False \vee a \bmod 3 \neq 0 \equiv a \bmod 3 \neq 0$

3. Correctitud de GCIo. FEDERICO HOERTH 752/23

HOJA 4

Sea S_c el ~~programa~~ cuerpo del ciclo.

a) Probemos el axioma

① $P_c \Rightarrow I$.

$n > 0 \wedge i = 0 \Rightarrow 0 \leq i \leq n$

$i = 0 \wedge res = 0 \Rightarrow res = \sum_{k=1}^{0} 2^k = 0$

$i = 0 \wedge j = 1 \Rightarrow j = 2^{2i} = 2^{2 \cdot 0} = 2^0 = 1$

$\therefore P_c \Rightarrow I$

Los demás puntos no se van a poder demostrar ya que obs. res en Q es impar, y res en el inv. propuesto es par.

② $\{I \wedge B\}\ S\ \{I\}$:

Calculo $wp_1 = (i := i+1, I) \equiv def(i+1) \wedge_L I^i_{i+1} \equiv True \wedge I^i_{i+1} \equiv$
$0 \leq i+1 \leq n \wedge j = 2^{2(i+1)} \wedge res = \sum_{k=1}^{i+1} 2^k$

Calculo $wp_2 = wp(j := j*2, wp_1) \equiv def(j*2) \wedge_L {wp_1}^j_{j*2} \equiv True \wedge {wp_1}^j_{j*2}$
$\equiv 0 \leq i+1 \leq n \quad j*2 = 2^{2i+2} \wedge res = \sum_{k=1}^{i+1} 2^k$

Calculo $wp(S,I) \equiv (res := res + j,\ wp_2) \equiv$

$def(res+j) \wedge_L wp_2{}^{res}_{res+j} \equiv True \wedge wp_2{}^{res}_{res+j} \equiv 0 \leq i+1 \leq n \wedge j*2 = 2^{2i+2} \wedge res+j = \sum_{k=1} 2$

$\equiv 0 \leq i+1 \leq n \wedge j = 2^{2i+1} \wedge res = \sum_{k=1}^{i-1} 2^k$

Ahora, debemos ver ~~q~~ si:

$\{I \wedge B\} \Rightarrow_L wp(S,I)$

En $I \wedge B$:

Vemos que $0 \leq i \leq n \wedge i < n \equiv 0 \leq i < n \Rightarrow 0 \leq i+1 \leq n$ ✓

pero vemos que $j = 2^{2i} \not\Rightarrow j = 2^{2i+1}$ ✓

y que $res = \sum_{k=1}^{i-1} 2^k \not\Rightarrow res = \sum_{k=1}^{i-1} 2^k$ ✓

③ $I \wedge \neg B \Rightarrow Qc$

$I \wedge \neg B \equiv 0 \leq i \leq n \wedge j = 2^{2i} \wedge res = \sum_{k=1}^{i} 2^k \wedge i \geq n$

$\equiv i = n \wedge j = 2^{2n} \wedge res = \sum_{k=1}^{n} 2^k$

$res = \sum_{k=1}^{n} 2^k \not\Rightarrow res = \sum_{k=1}^{n-1} 2^k$ ✓

$I \wedge \neg B \not\Rightarrow Qc$ ✓

HOERTH FEDERICO

752/23 Hoja 5

$$I \equiv \{0 \le i \le n \wedge j = 2^i \wedge res = \sum_{k=0}^{i-1} 2^k\}$$

c)

- $fv = n$. No es una función variante decreciente válida, ya que es constante, y se necesita una fv decreciente

- $fv = n - i$. Es válido. Es decreciente. Y cuando $i = n$ $fv(i) = 0$, nos va a servir para dem. los axiomas del Teorema de Terminación.

- $fv = n - i - 1$ Si bien es una fv decreciente tiene un problema. Cuando $i = n$ $fv(i) = -1$, y en la demo $I \wedge fv \le 0 \Rightarrow \neg B$ se va a complicar.

Vamos a tener que:

$I \wedge fv \le 0 \Rightarrow i \le n \wedge n - i - 1 \le 0$

$\Rightarrow i \le n \wedge i \ge n - 1$

$\Rightarrow n - 1 \le i \le n \equiv i = n \vee i = n - 1$

Es una disyunción, así que debo probar que:

$i = n \Rightarrow \neg B$ pero $i = n - 1 \not\Rightarrow i \ge n$

caso $i = 22$ $n = 23$

$i = n - 1 \Rightarrow \neg B$ $i = n - 1 \not\Rightarrow \neg B$.

Por lo tanto, esa fv no nos sirve.

HUERTA FEDERICO
752/23 Hoja 6

) Sea S el programa
P la precondición $\{|s| \bmod 2 = 0\}$
Q la postcondición $\{res = true \Leftrightarrow (\forall i:\mathbb{Z})(0 \le i < |s| \rightarrow_L s[i] \bmod 2 = 0)\}$

~~Por hipótesis~~
Se sabe entonces que $\{P\} S \{Q\}$ es válido, por ende
$P \Rightarrow_L wp(S,Q)$

~~Se P' tal que P ⇒ P' es taut, P' más débil que P.~~

a) Sea P', una pre más fuerte, talque
$P' \Rightarrow P$ es Verdadero y $P \Rightarrow wp(S,Q)$ entonces
$P' \Rightarrow wp(S,Q)$ sii $\{P'\} S \{Q\}$ es válido.

Conclusión: El programa sigue siendo correcto. ✓

b) Sea Q' una post más debil, tal que
↳ sigue hablando de una precondición el enunciado.
$Q \Rightarrow Q'$ es Verdadero, por monotonía se tiene que:
$wp(S,Q) \Rightarrow wp(S,Q')$
y como $P \Rightarrow wp(S,Q) \Rightarrow wp(S,Q')$ entonces
$P \Rightarrow wp(S,Q') \Leftrightarrow \{P\} S \{Q'\}$ es válido.

Conclusión: El programa sigue siendo correcto. ✗