Algoritmos y Estructuras de Datos II
Primer parcial – 25 de abril de 2015

Aclaraciones

- El parcial es **a libro abierto**.
- Cada ejercicio debe entregarse **en hojas separadas**.
- Incluir en cada hoja el número de orden asignado, número de hoja, apellido y nombre.
- Al entregar el parcial, completar el resto de las columnas en la planilla.
- Cada ejercicio se calificará con **MB**, **B**, **R** o **M**, y podrá recuperarse independientemente de los demás. La cursada podrá aprobarse con hasta 1 (un) ejercicio **R** (regular) en cada parcial (post-recuperatorios), siempre y cuando ninguno de ellos sea un ejercicio 1. Para más detalles, ver "Información sobre la cursada" en el sitio Web.

Ej. 1. Especificación

Nos contratan para desarrollar el sistema de control de naves-taxi de cierta galaxia. Las licencias para taxis pueden tramitarse en cualquier momento dando un examen, que varía según el planeta. Cuando una nave aprueba el examen de cierto planeta, queda habilitada para trabajar en el mismo.

Cada vez que una nave finaliza una jornada de trabajo en un planeta, la computadora de a bordo transmite al sistema la cantidad de horas trabajadas y el planeta en el que trabajó. Una nave puede trabajar en cualquier planeta para el que tenga licencia. Por razones de seguridad no hay viajes interplanetarios con pasajeros.

En cada planeta hay ciertas maniobras que están prohibidas: pasar en rojo, girar en U, romper la barrera del sonido frente a una escuela, etcétera. Cuando una nave comete una infracción, la computadora de a bordo le transmite al sistema la infracción cometida y el lugar del hecho. Si una nave comete 3 veces la misma infracción en el mismo planeta, inmediatamente pierde —de manera irreversible— su licencia para volar en ese planeta.

Para fines tributarios interesa poder averiguar, dado un planeta, qué naves son las que llevan mayor cantidad de horas trabajadas en él.

Modelar este sistema utilizando tipos abstractos de datos. Documentar y/o justificar las decisiones tomadas que considere particularmente importantes.

Ej. 2. Inducción estructural

Considerar el siguiente tipo abstracto de datos y las operaciones asociadas:

TAD ÁrbolTernario(α)

observadores básicos

nil?	: at(α)	$\longrightarrow$	bool	
raíz	: at(α) t	$\longrightarrow$	α	$\{\neg$ nil?$(t)\}$
izq	: at(α) t	$\longrightarrow$	at(α)	$\{\neg$ nil?$(t)\}$
med	: at(α) t	$\longrightarrow$	at(α)	$\{\neg$ nil?$(t)\}$
der	: at(α) t	$\longrightarrow$	at(α)	$\{\neg$ nil?$(t)\}$

generadores

nil	:	$\longrightarrow$	at(α)
tern	: $\alpha \times$ at(α) $\times$ at(α) $\times$ at(α)	$\longrightarrow$	at(α)

...

Fin TAD

preorder : at(α) $\longrightarrow$ secu(α)

p_0) preorder(nil) $\equiv$ <>

p_1) preorder(tern(x, i, m, d)) $\equiv$ $x \bullet$ (preorder(i) & preorder(m) & preorder(d))

tamaño : at(α) $\longrightarrow$ nat

t_0) tamaño(nil) $\equiv$ 0

t_1) tamaño(tern(x, i, m, d)) $\equiv$ 1 + tamaño(i) + tamaño(m) + tamaño(d)

long : secu(α) $\longrightarrow$ nat

l_0) long(<>) $\equiv$ 0

l_1) long($a \bullet s$) $\equiv$ 1 + long(s)

$\bullet$ & $\bullet$: secu(α) $\times$ secu(α) $\longrightarrow$ secu(α)

&$_0$) <> & t $\equiv$ t

&$_1$) ($a \bullet s$) & t $\equiv$ $a \bullet (s$ & $t)$

Interesa demostrar por inducción estructural la siguiente propiedad:

$$\big(\forall a : \text{at}(\alpha)\big)\big(\ \text{long}(\text{preorder}(a)) \ \equiv\ \text{tamaño}(a)\ \big)$$

a) Escribir el predicado unario. Luego escribir, completo, **el esquema de inducción** a utilizar.
En el esquema, marcar **claramente** CB(s), PI(s), HI(s), TI(s) y el alcance de cada cuantificador.

b) Plantear el/los caso(s) base y resolverlo(s), justificando cada paso de la demostración.

c) Plantear el/los paso(s) inductivo(s) y resolverlo(s), justificando cada paso de la demostración.

En caso de utilizar lemas, plantearlos claramente y **demostrarlos**.

Ej. 3. Diseño

La siguiente especificación modela un castillo donde conviven piratas y ninjas. Con frecuencia arriban al castillo nuevos piratas y ninjas, que nunca mueren ni se van. Por supuesto, cada tanto surgen peleas, que por tradición ancestral son siempre entre un pirata y un ninja. Los piratas y los ninjas se identifican con naturales unívocos: no hay dos piratas, ni dos ninjas, ni un pirata y un ninja que se identifiquen con el mismo número.

TAD CASTILLO

observadores básicos

piratas	: castillo	$\longrightarrow$	conj(nat)	
ninjas	: castillo	$\longrightarrow$	conj(nat)	
cantPeleas	: castillo c $\times$ nat p $\times$ nat n	$\longrightarrow$	nat	$\{p \in \text{piratas}(c) \land n \in \text{ninjas}(c)\}$

generadores

crear	:	$\longrightarrow$	castillo	
llegaPirata	: castillo c $\times$ nat p	$\longrightarrow$	castillo	$\{p \notin (\text{piratas}(c) \cup \text{ninjas}(c))\}$
llegaNinja	: castillo c $\times$ nat n	$\longrightarrow$	castillo	$\{n \notin (\text{piratas}(c) \cup \text{ninjas}(c))\}$
pelean	: castillo c $\times$ nat p $\times$ nat n	$\longrightarrow$	castillo	$\{p \in \text{piratas}(c) \land n \in \text{ninjas}(c)\}$

axiomas

piratas(crear)	$\equiv \emptyset$	ninjas(crear)	$\equiv \emptyset$
piratas(llegaPirata(c,p))	$\equiv$ Ag(p, piratas(c))	ninjas(llegaPirata(c,p))	$\equiv$ ninjas(c)
piratas(llegaNinja(c,n))	$\equiv$ piratas(c)	ninjas(llegaNinja(c,n))	$\equiv$ Ag(n, ninjas(c))
piratas(pelean(c,p,n))	$\equiv$ piratas(c)	ninjas(pelean(c,p,n))	$\equiv$ ninjas(c)

cantPeleas(llegaPirata(c,p'), p,n) $\equiv$ **if** $p = p'$ **then** 0 **else** cantPeleas(c,p,n) **fi**
cantPeleas(llegaNinja(c,n'), p,n) $\equiv$ **if** $n = n'$ **then** 0 **else** cantPeleas(c,p,n) **fi**
cantPeleas(pelean(c,p',n'), p,n) $\equiv$ **if** $p = p' \land n = n'$ **then** 1 **else** 0 **fi** + cantPeleas(c,p,n)

Fin TAD

Para representar el TAD CASTILLO se decidió utilizar la siguiente estructura:

castillo **se representa con** estr, donde

estr es tupla $\langle$ *piratas*: conj(nat),
ninjas: conj(nat),
rivalesQueTuvo: dicc(nat, conj(nat)),
historialPeleas: secu(tupla $\langle p$: nat, n : nat$\rangle$) $\rangle$

donde *piratas* y *ninjas* representan los conjuntos de identificadores de piratas y ninjas, respectivamente, *rivalesQueTuvo* asocia a cada peleador (tanto piratas como ninjas, ya que todos los identificadores son distintos) con el conjunto de todos los rivales contra los que peleó al menos una vez, e *historialPeleas* tiene la secuencia de parejas $\langle$pirata, ninja$\rangle$ que se entreveraron en una pelea, en el orden en que éstas sucedieron.

a) Escribir en castellano el invariante de representación.

b) Escribir formalmente el invariante de representación.

c) Escribir formalmente la función de abstracción.