Algoritmos y Estructuras de Datos II
Recuperatorio del primer parcial – 30 de noviembre de 2013

Aclaraciones

- El parcial es **a libro abierto**.
- Cada ejercicio debe entregarse **en hojas separadas**.
- Incluir en cada hoja **el número de orden asignado**, así como el número de hoja, apellido y nombre.
- Al entregar el parcial, completar el resto de las columnas en la planilla. (El enunciado **no** cuenta como una hoja.)
- Cada ejercicio se calificará con **MB**, **B**, **R** o **M**, y puede recuperarse independientemente de los demás. Podrán aprobarse los parciales de la materia con hasta un ejercicio R (regular) y uno M (mal) **entre ambos exámenes**, siempre que ninguno de dichos ejercicios sea un ejercicio 1. Para más detalles ver la sección *Cursada* del sitio Web.

Ej. 1. Especificación

Nos han contratado para especificar el sistema de manejo de contactos del nuevo dispositivo móvil "Emilio", que debe operar de la siguiente manera: Un *contacto* se define por su dirección de correo electrónico y puede tener asociados uno o más pares de nombre y apellido. Cada uno de esos pares de nombre y apellido asociados se llama un *alias* del contacto. El sistema debe ser capaz de informar cuántos alias posee cada contacto.

El sistema se controla mediante comandos de voz. Para agregar un contacto o un alias, el usuario debe decir algo de la forma: "Agregar Arnoldo Pérez, aperez@uba.ar". Dicho comando debe agregar un nuevo contacto si la dirección es desconocida, o bien un nuevo alias al contacto cuando se trate de una dirección ya conocida.

El sistema permite enviar y recibir *mensajes* por correo electrónico. Para enviar un mensaje, el usuario dice algo de la forma: "Enviar mensaje a Mariela Gómez", y a continuación procede a dictar el cuerpo del mensaje. Notar que el nombre y apellido provistos por el usuario pueden ser cualquier alias conocido.

Al recibir un mensaje entrante, el sistema recibe del hardware –además del cuerpo del mensaje– la dirección de correo electrónico del remitente, y un campo "De:" que contiene un nombre y apellido (es decir, un alias). Observar que tanto la dirección como el alias del remitente podrían ser conocidos o desconocidos. En caso de ser desconocidos, son automáticamente agregados al sistema.

Interesa, en todo momento, poder averiguar cuántos mensajes se han recibido de cada dirección conocida, y determinar a qué dirección (o direcciones) de correo se han enviado más mensajes.

a) Especifique el TAD TienesUnEmilio. (Pronúnciese con acento ibérico.)

b) ¿Cómo cambiaría su TAD si nos interesara poder leer, completos, todos los mensajes del contacto del cual más mensajes se han recibido? ¿Qué generadores, obs. básicos y/u otras operaciones habría que agregar? ¿Por qué? Describa brevemente —no es obligatorio axiomatizar, pero procure ser claro/a— y justifique.

Ej. 2. Inducción estructural

La función *caminos* computa todos los caminos (desde la raíz hasta alguna hoja) de un árbol binario:

$$
\begin{array}{lll}
\text{caminos} & : & \text{ab}(\alpha) \longrightarrow \text{secu}(\text{secu}(\alpha)) \\
c_0) & & \text{caminos}(\text{nil}) \equiv <> \\
c_1) & & \text{caminos}(\text{bin}(i,e,d)) \equiv \textbf{if } \text{nil?}(i) \wedge \text{nil?}(d) \textbf{ then } (e \bullet <>) \bullet <> \\
 & & \textbf{else } \text{agATodos}(e, \text{caminos}(i)) \,\&\, \text{agATodos}(e, \text{caminos}(d)) \textbf{ fi}
\end{array}
$$

$$
\begin{array}{lll}
\text{agATodos} & : & \alpha \times \text{secu}(\text{secu}(\alpha)) \longrightarrow \text{secu}(\text{secu}(\alpha)) \\
a_1) & & \text{agATodos}(x,s) \equiv \textbf{if } \text{vacía?}(s) \textbf{ then } <> \\
 & & \textbf{else } (x \bullet \text{prim}(s)) \bullet \text{agATodos}(x, \text{fin}(s)) \textbf{ fi}
\end{array}
$$

$$
\begin{array}{lll}
\text{long} & : & \text{secu}(\alpha) \longrightarrow \text{nat} \\
l_0) & & \text{long}(<>) \equiv 0 \\
l_1) & & \text{long}(a \bullet s) \equiv 1 + \text{long}(s)
\end{array}
$$

$$
\begin{array}{lll}
\text{h} & : & \text{ab}(\alpha) \longrightarrow \text{nat} \\
h_0) & & \text{h}(\text{nil}) \equiv 0 \\
h_1) & & \text{h}(\text{bin}(i,e,d)) \equiv 1 + \text{máx}(\text{h}(i), \text{h}(d))
\end{array}
$$

$$
\begin{array}{lll}
\text{máx} & : & \alpha \times \alpha \longrightarrow \alpha \\
m_1) & & \text{máx}(x,y) \equiv \textbf{if } x >_\alpha y \textbf{ then } \text{x} \textbf{ else } \text{y} \textbf{ fi}
\end{array}
$$

Demuestre por inducción estructural que $\big(\forall\, a\text{: ab}(\alpha)\big)\,\big(\text{long}(\text{caminos}(a)) \leq 2^{\text{h}(a)}\big)$.

a) Escriba el predicado unario. Exhiba el esquema completo de inducción a utilizar.
Marque **claramente** CB(s), PI(s), HI(s), TI(s) y el alcance de cada cuantificador.

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

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

Nota: En caso de necesitar uno o más lemas auxiliares, enúncielo(s) claramente. No es necesario demostrarlo(s).

Ej. 3. Diseño

Una fábrica de sandwiches de miga debe llevar cuenta de los diversos ingredientes que necesita, los sucesivos pedidos que se le realizan y cuáles de esos pedidos ya han sido entregados:

TAD INGREDIENTE **ES** NAT ; **TAD** NROPEDIDO **ES** NAT ; **TAD** CANTIDAD **ES** NAT
TAD SÁNGUCHE **ES** STRING ; **TAD** PEDIDO **ES** DICC(SÁNGUCHE, CANTIDAD)

TAD SANGUCHERÍA

observadores básicos

sánguches	: sanguchería	$\longrightarrow$ conj(sánguche)		
ingredientes	: sanguchería s × sánguche g	$\longrightarrow$ conj(ingrediente)		$\{g \in \text{sánguches}(s)\}$
últimoPedido	: sanguchería	$\longrightarrow$ nropedido		
verPedido	: sanguchería s × nropedido n	$\longrightarrow$ pedido		$\{n \leq \text{últimoPedido}(s)\}$
pedidosPendientes	: sanguchería s	$\longrightarrow$ conj(nropedido)		

otras operaciones

pedidosEntregados	: sanguchería s	$\longrightarrow$ conj(nropedido)	

generadores

inaugurar	:	$\longrightarrow$ sanguchería	
definirSánguche	: sanguchería s × sánguche g × conj(ingrediente)	$\longrightarrow$ sanguchería	$\{g \notin \text{sánguches}(s)\}$
tomarPedido	: sanguchería s × nropedido n × pedido p	$\longrightarrow$ sanguchería	$\{n = \text{últimoPedido}(s)+1 \wedge \text{claves}(p) \subseteq \text{sánguches}(s)\}$
entregarPedido	: sanguchería s × nropedido n	$\longrightarrow$ sanguchería	$\{n \leq \text{últimoPedido}(s) \wedge n \notin \text{pedidosEntregados}(s)\}$

axiomas ...

Fin TAD

Para representar el género imperativo Sanguchería se decidió utilizar la siguiente estructura:

Sanguchería **se representa con** estr, donde

estr es tupla ⟨ *sánguches*: dicc(sánguche, conj(ingrediente)),
ingredientes: conj(ingrediente),
ult_pedido: nropedido,
pedidos_entregados: dicc(nropedido, dicc(sánguche, cant)),
pedidos_pendientes: dicc(nropedido, dicc(sánguche, cant)),
pendientes_por_ingrediente: dicc(ingrediente, conj(nropedido)) ⟩

donde

- *sánguches* indica la composición (sus ingredientes) de cada sánguche,
- *ingredientes* contiene los ingredientes de **todos** los sánguches definidos,
- *ult_pedido* almacena el número del pedido más reciente (o, en su defecto, cero),
- *pedidos_entregados* y *pedidos_pendientes* asocian números de pedido (entregados y pendientes, respect.) con los detalles del pedido en cuestión (qué sánguches se pidieron y en qué cantidades),
- y *pendientes_por_ingrediente* indica, para cada ingrediente d, los números de los pedidos pendientes que involucren algún sánguche que necesite d.

a) Escriba el invariante de representación en castellano.

b,c) Escriba formalmente *b)* el invariante de representación y *c)* la función de abstracción.