

Algoritmos y Estructuras de Datos II

Primer parcial — 3/5/2008

Aclaraciones

- El parcial **NO** es a libro abierto.
- Numere las hojas entregadas. Complete en la primera hoja la cantidad total de hojas entregadas.
- Incluya el número de orden asignado (léalo cuando circule la planilla), apellido y nombre en cada hoja.
- Al entregar el parcial complete los datos faltantes en la planilla.
- Cada ejercicio se calificará con B, R ó M. Una B no significa que el ejercicio está “perfecto”, sino que cumple con los requisitos necesarios para aprobar. En los parciales promocionados se asignará una nota numérica más precisa a cada ejercicio.
- Para aprobar el parcial debe obtenerse B en el ejercicio 1 y en alguno de los ejercicios 2 y 3. Un parcial se considera promocionado si está aprobado y su puntaje es 70 o superior.

Ej. 1. Especificación

Debido a la gran congestión de los servicios urbanos de colectivos, la UBA decidió operar un servicio de micros express para transportar a sus alumnos y docentes desde una única parada, ubicada en Plaza Italia. Los usuarios deben registrarse previamente para poder utilizarlo este servicio.

La operatoria es la siguiente: Cuando el micro llega a la parada, detiene el motor y se queda esperando pasajeros. Éstos van subiendo al micro a medida que llegan. En el momento en que la cantidad de pasajeros es exactamente 30, el micro parte (no le está permitido viajar ni con menos ni con más pasajeros).

Las personas que llegan cuando el colectivo ya partió, forman una cola y se quedan esperando el próximo servicio. Cuando éste llega, se suben inmediatamente, siempre y cuando haya espacio disponible. Aquellos que no alcancen a subir, deberán quedarse en la parada esperando al siguiente micro.

Se pide:

- a) Modelar el funcionamiento de la parada de micros y la registración de usuarios mediante TADs.
- b) Responda la siguiente pregunta: ¿Qué comportamientos son automáticos y cuáles son manuales? ¿Cómo se ve esto reflejado en el TAD propuesto?

Ej. 2. Inducción estructural

La siguiente función devuelve, dado un árbol binario, todos los caminos del árbol, desde la raíz hasta alguna hoja.

```

caminos : ab( $\alpha$ )  $\longrightarrow$  secu(secu( $\alpha$ ))
caminos(a)  $\equiv$  if nil?(a) then <> else
               if nil?(izq(a))  $\wedge$  nil?(der(a)) then (a • <>) • <> else
               agATodos(raiz(a), caminos(izq(a))) & agATodos(raiz(a), caminos(der(a))) fi fi

```

donde

```

agATodos :  $\alpha \times \text{secu}(\text{secu}(\alpha)) \longrightarrow \text{secu}(\text{secu}(\alpha))$ 
agATodos(a,s)  $\equiv$  if vacía?(s) then <>
               else (a • prim(s)) • agATodos(a, fin(s)) fi

```

Se quiere probar por inducción estructural la siguiente propiedad:

$$(\forall a : \text{ab}(\alpha)) (\text{long}(\text{caminos}(a)) \leq 2^{h(a)})$$

donde

$h : \text{ab}(\alpha) \longrightarrow \text{nat}$
 $h(a) \equiv \text{if nil?}(a) \text{ then } 0 \text{ else } 1 + \max(h(\text{izq}(a)), h(\text{der}(a))) \text{ fi}$

Deberá escribir claramente el esquema de inducción, indicando: predicado a demostrar, paso inductivo, caso base, hipótesis y tesis inductiva. Marque claramente el alcance de cada uno de los cuantificadores que utilice. En la demostración, puede valerse de las propiedades de los números naturales. Si necesita utilizar algún lema, enuncielo correctamente (no es necesario que lo demuestre).

Ej. 3. Diseño

En la cátedra de Algo2 necesitamos implementar en forma urgente un sistema para administrar las notas de los alumnos. Como primer paso, decidimos especificar y diseñar el TAD TP. Para cada TP, los alumnos se dividen en grupos. Aunque indicamos la cantidad de personas que tiene que tener el grupo, siempre aparecen alumnos que forman grupos de menos integrantes y son igualmente aceptados. En cambio, nunca se aceptan grupos con un número de integrantes mayor. La nota siempre es para todo el grupo; nunca ponemos notas individuales para los integrantes (aún en el caso en que sospechemos que algunos trabajaron más que otros).

Se modeló el problema mediante el siguiente TAD:

TAD TP

generadores

nuevo : $\text{nat } tamMax \longrightarrow \text{tp}$
 agregarGrupo : $\text{tp } t \times \text{conj}(\text{alumno}) \ c \longrightarrow \text{tp}$
 $\#c < \text{tamañoMáximo}(t) \wedge (\forall n : \text{nat})(n < \text{cantGrupos}(t) \Rightarrow_L c \cap \text{grupo}(t, n) = \emptyset)$
 agregarNota : $\text{tp } t \times \text{nat } n \times \text{nota} \longrightarrow \text{tp}$ $n < \text{cantGrupos}(t) \wedge_L \neg \text{tieneNota}(t, n)$

observadores básicos

tamañoMáximo : $\text{tp} \longrightarrow \text{nat}$
 cantGrupos : $\text{tp} \longrightarrow \text{nat}$
 grupo : $\text{tp} \times \text{nat} \longrightarrow \text{conj}(\text{alumno})$ $n < \text{cantGrupos}(t)$
 tieneNota : $\text{tp } t \times \text{nat } n \longrightarrow \text{bool}$ $n < \text{cantGrupos}(t)$
 nota : $\text{tp } t \times \text{nat } n \longrightarrow \text{nota}$ $n < \text{cantGrupos}(t) \wedge_L \text{tieneNota}(t, n)$
 ...

Fin TAD

TAD alumno es string

TAD nota es nat

En la etapa de diseño, se decidió utilizar la siguiente estructura para representar un TP:

TP se representa con *estr*, donde

estr es tupla
 $\langle \text{grupos: dicc}(\text{nat}, \text{conj}(\text{alumno})),$
 $\text{notas: diccionario}(\text{alumno}, \text{nota}),$
 $\text{tamañoMáximo: nat} \rangle$

donde las claves de *grupos* representan el número de grupo y el valor asociado el grupo de alumnos correspondiente, *notas* contiene, para cada alumno, la nota que éste obtuvo, y *tamañoMáximo* indica la cantidad máxima de integrantes de cada grupo.

Se pide:

- Escribir formalmente y en castellano el invariante de representación.
- Escribir la función de abstracción.