

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.

Se desea modelar el juego de mesa que se describe a continuación usando TADs. El mismo se juega sobre un tablero cuadrado de 64×64 posiciones. En este juego pueden participar siempre 64 jugadores. El juego comienza cuando los jugadores se posicionan en el tablero: esto es, eligen una celda del tablero y una dirección (norte, sur, este, oeste). No se admiten dos jugadores posicionados en la misma celda. El puntaje inicial de cada jugador es 0.

En cada turno del juego algún jugador avanza o cambia de dirección. Si avanza, se desplaza a la celda siguiente de acuerdo con la dirección que tiene el jugador. Si se encuentra posicionado en un extremo del tablero y tiene una dirección que lo lleva fuera del tablero, al avanzar se posiciona en el otro extremo del tablero (por ejemplo, si se encuentra en la celda más al norte de la columna i y tiene dirección norte, al avanzar se desplaza a la celda más al sur de la columna i). Si al desplazarse, un jugador llega a una celda que está ocupada por otro jugador, este último es eliminado y quién se movió gana todos los puntos que había acumulado el jugador eliminado más un bonus de 1 punto. Si cambia de dirección, el jugador no se desplaza.

En cada turno, se elige no determinísticamente el jugador que tiene derecho a moverse o cambiar de direcci3n. El juego termina cuando un jugador alcanza un puntaje que es superior a la mitad del numero de jugadores que iniciaron el juego.

Puede asumir que se encuentran definidos los tipos POSICION, DIRECCION y TABLERO, con TABLERO definido de la siguiente manera:

g�neros	tablero
---------	---------

generadores

nuevo : \longrightarrow tablero

otras operaciones

$$\text{siguiente} : \text{posicion } p \times \text{direccion } d \times \text{tablero } t \longrightarrow \text{posicion}$$

posicionValida(p,t)

$$\text{posicionValida} : \text{posicion} \times \text{tablero} \longrightarrow \text{bool}$$

Fin TAD

El generador *nuevo* permite crear un tablero de 64×64 posiciones, la operación *posicionValida* permite saber si una determinada posición es una celda válida del tablero, y la operación *siguiente*(p, d, t) permite obtener la celda siguiente a p en la dirección d (esta función maneja adecuadamente los casos en que la celda p se encuentre en un extremo del tablero).

Considerar el siguiente TAD que modela el funcionamiento de un banco que posee tres ventanillas de atención a clientes y una cola única de espera.

TAD BANCO**géneros** banco**observadores básicos**enEspera : banco \longrightarrow natatendiendo : banco \longrightarrow nat**generadores**crear : \longrightarrow bancollega : banco \longrightarrow bancoparte : banco \longrightarrow banco

atendiendo(b)>0

axiomasenEspera(crear) \equiv 0enEspera(llega(b)) \equiv enEspera(b) + (if atendiendo(b) < 3 then 0 else 1 fi)enEspera(parte(b)) \equiv enEspera(b) - (if atendiendo(b) > 0 then 1 else 0 fi)atendiendo(crear) \equiv 0atendiendo(llega(b)) \equiv if atendiendo(b) < 3 then atendiendo(b) + 1 else 3 fiatendiendo(parte(b)) \equiv atendiendo(b) - (if enEspera(b) > 0 then 0 else 1 fi)**Fin TAD**

El observador *enEspera* permite obtener el número de clientes que se encuentra en la cola de espera, mientras que *atendiendo* permite observar el número de cajas ocupadas.

La especificación anterior debería reflejar la siguiente política del banco: “Ningún cliente debe esperar si hay ventanillas desocupada”. Por este motivo se desea contar con una prueba formal de la siguiente proposición:

$$(\forall b : \text{banco}) P(b), \text{ donde } P(b) \equiv (\text{enEspera}(b) > 0 \Rightarrow \text{atendiendo}(b) = 3)$$

Demuéstrelo por inducción estructural.

1. Dar el esquema de inducción.
2. Plantear el caso base y resolverlo, justificando cada paso de la demostración.
3. Plantear el/los paso/pasos inductivos, marcando claramente hipótesis, tesis inductiva y alcance de los cuantificadores. Resolver, justificando cada paso de la demostración.

Nota: Si se necesitaran otros lemas auxiliares, enúncielos, plantee su esquema de demostración y demuéstrelos. Puede utilizar sin demostrar el siguiente lema.

Lema 1 $(\forall b : \text{banco})(P(b) \Rightarrow P(\text{llega}(b)))$

Ej. 3. Invariante de Representación y Función de Abstracción

Considerar el siguiente TAD que modela el comportamiento de una oficina del estado que procesa trámites. Cada trámite está identificado por un ID y se le asigna una categoría al momento de ingresar. Las categorías de la oficina no son fijas, y pueden agregarse nuevas categorías en cualquier momento. En cualquier momento se puede dar prioridad a una categoría. Todos los trámites pendientes que pertenecen a una categoría prioritaria se consideran prioritarios (Notar que en esta oficina, como buena dependencia estatal, un trámite nunca concluye):

TAD OFICINA**géneros** oficina**observadores básicos**categorias : oficina \rightarrow conj(categoria)pendientes : oficina \rightarrow secu(id)prioritarias : oficina \rightarrow conj(categoria)catTram : id $i \times$ oficina $o \rightarrow$ categoria $i \in pendientes(o)$ **generadores**nuevo : \rightarrow oficinanuevaCat : categoria $c \times$ oficina $o \rightarrow$ oficinanuevoTram : id $i \times$ categoria $c \times$ oficina $o \rightarrow$ oficina $c \notin categorias(o)$ priorizar : categoria $c \times$ oficina $o \rightarrow$ oficina $i \notin pendientes(o) \wedge c \in categorias(o)$ **otras operaciones**pendPrioritarios : oficina \rightarrow secu(id)filtrarPorCategorias : secu(id) $s \times$ conj(categoria) \times oficina $o \rightarrow$ secu(id)
($\forall i : \text{nat} 0 \leq i < \text{long}(s) \Rightarrow_L s[i] \in pendientes(o)$)**axiomas**categorias(nuevo) \equiv vaciocategorias(nuevaCat(c, o)) \equiv Ag($c, categorias(o)$)categorias(nuevoTram(i, c, o)) \equiv categorias(o)categorias(priorizar(c, o)) \equiv categorias(o)pendientes(nuevo) $\equiv \langle \rangle$ pendientes(nuevaCat(c, o)) \equiv pendientes(o)pendientes(nuevoTram(i, c, o)) $\equiv i \bullet pendientes(o)$ pendientes(priorizar(c, o)) \equiv pendientes(o)prioritarias(nuevo) \equiv vacioprioritarias(nuevaCat(c, o)) \equiv prioritarias(o)prioritarias(nuevoTram(i, c, o)) \equiv prioritarias(o)prioritarias(priorizar(c, o)) \equiv Ag($c, prioritarias(o)$)catTram($i, nuevaCat(c, o)$) \equiv catTram(i, o)catTram($i, nuevoTram(i', c, o)$) \equiv **if** $i = i'$ **then** c **else** catTram(i, o) **fi**catTram($i, priorizar(c, o)$) \equiv catTram(i, o)pendPrioritarios(o) \equiv filtrarPorCategorias(pendientes(o), prioritarias(o), o)filtrarPorCategorias(s, c, o) \equiv **if** vacia?(s) **then** $\langle \rangle$
else (**if** catTram(prim(s), o) $\in c$ **then** prim(s) $\bullet \langle \rangle$ **else** $\langle \rangle$ **fi**)
& filtrarPorCategorias(fin(s), c, o) **fi****Fin TAD**

Se decidió utilizar la siguiente estructura como representación:

OFICINA se representa con estr, donde

estr es tupla \langle catPrioritarias: conj(categoria),
 tramites: dicc(id, categoria),
 tramites \times Cat: dicc(categoria, conj(id)),
 pendPrioritarios: secu(id)
 pendientes: secu(id) \rangle

catPrioritarias representa el conjunto de todas las categorías a las que se ha dado prioridad, tramites asocia a cada trámite su categoría mientras que tramites \times Cat describe todos los trámites asociados a cada categoría. pendPrioritarios contiene la secuencia de trámites pendientes que tienen una categoría prioritaria mientras que pendientes contiene todos los trámites pendientes (incluso a los prioritarios).

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