

Práctica integradora (1er parcial)

- Si bien el presente no es un examen, puede ser beneficioso intentar realizarlo dentro de un marco de tiempo equivalente al que tendrá en el mismo, en un ambiente sin interrupciones y de forma individual en papel.
- También recomendamos aspirar a escribir de forma prolija, o pasar el contenido de hojas borradores a hojas finales para poder estimar correctamente durante el proceso los tiempos que esta tarea demandaría durante el examen.
- Los procedimientos y funciones primitivos que son dados en el examen deberán ser implementados en caso de que se desee probar la solución en máquina, por lo que recomendamos que aspire a realizar esto solamente una vez terminado de realizar los ejercicios en papel.
- Aconsejamos leer el enunciado completo antes de empezar a resolver los ejercicios, ya que ayuda a comprender mejor el dominio.
- Recuerde que el parcial será a libro abierto, que puede consultar cualquier material (en papel) que haya sido escrito antes de comenzar el examen y usar sin definir todas las funciones y procedimientos vistos durante la cursada. Tenga esto en cuenta al desarrollar su solución.
- Pensar bien la estrategia a seguir, expresando la misma mediante procedimientos y funciones. Recordar escribir primero los contratos.

Convención de Motoqueros

Convención de Motoqueros es una simulación del comportamiento de motoqueros a lo largo de una convención. En las convenciones, a los motoqueros les gusta interactuar con otros motoqueros y asociarse a clubes.

Convención de Motoqueros se puede modelar en Gobstones. Cada motoquero puede representarse con una celda del tablero. Un motoquero es vecino de otro si sus correspondientes celdas son lindantes en cualquiera de las 4 direcciones ortogonales. La imagen de la izquierda de la Figura 1 muestra una convención posible.

Durante una convención, se arman clubes. Es decir, grupos de motoqueros activos que están conectados entre sí. Dos motoqueros se dicen conectados cuando hay una secuencia de motoqueros activos vecinos que los unen. Podemos además asignarles números a cada club, para diferenciarlos unos de otros y determinar cosas como qué grupo tiene más

miembros, o cuál tiene menos. La imagen de la derecha de la Figura 1 muestra diversos grupos en la convención anterior, habiendo asignado un número a cada grupo.

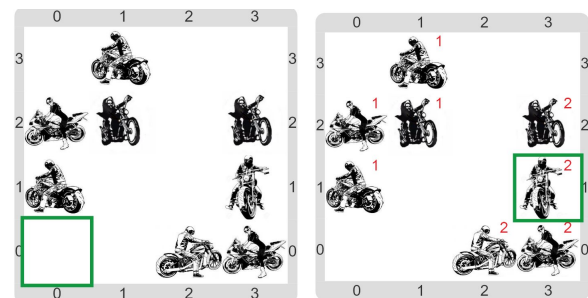


Figura 1: A la izquierda, una convención de motoqueros, con 13 motoqueros activos en total. A la derecha, la misma convención con los clubes marcados. Hay 2 clubes distintos en la convención, siendo ambos clubes de 4 miembros.

Se cuenta con los procedimientos y funciones primitivas que se enuncian a continuación:

- **elMotoqueroPerteneceAUnClub()**
PROPÓSITO: Indica si el motoquero actual pertenece a algún club.
PRECONDICIÓN: El motoquero actual se encuentra activo.
RESULTADO: Un valor de verdad.
- **clubAlQuePerteneceUnMotoquero()**
PROPÓSITO: Describe el número del club al cual pertenece el motoquero actual.
PRECONDICIÓN: El motoquero actual se encuentra activo y pertenece a un club.
RESULTADO: Un número de club válido.
- **IncluirEnElClubNúmero_(número)**
PROPÓSITO: Incluye al motoquero actual como perteneciente al club con el número dado.
PRECONDICIÓN: El motoquero actual se encuentra activo y no pertenece todavía a ningún club.
- **elMotoqueroEstáActivo()**
PROPÓSITO: Indica si el motoquero actual está activo o no.
PRECONDICIÓN: Ninguna.
- **RESULTADO:** Un valor de verdad.

EJERCICIOS

Ejercicio 1) Realizar la función `cantidadDeClubesEnLaConvención()` que determine la cantidad total de clubes de motoqueros que hay en la convención.

Para ello se propone una estrategia *top-down* para contar los clubes: asignar un número de club distinto y consecutivo a cada club, hasta no tener más clubes. El último número de club asignado debería permitir determinar la cantidad de clubes. Para esto se propone realizar las siguientes funciones y procedimientos (aunque pueden requerirse o ser útil algunos adicionales):

Ejercicio a) La función `hayMotoquerosActivosSinClub()` que indica si en la convención hay algún motoquero activo que todavía no tenga un club.

Ejercicio b) El procedimiento `IrAMotoqueroActivoSinClub()` que asumiendo que hay un motoquero activo sin club, deja el cabezal en el mismo.

Ejercicio c) El procedimiento `AsignarClub_AMotoquerosConectados(númeroDeClub)` que marca al motoquero actual y a todos los motoqueros conectados a él como pertenecientes al club `númeroDeClub`. Dichos motoqueros no deben tener un club previamente asignado. Este problema es a su vez complejo, y se propone la siguiente estrategia para solucionarlo.

- Paso 1, marcar al motoquero actual como perteneciente al club, y también a todos sus vecinos activos que no tengan club asignado.
- Paso 2, buscar en la convención algún motoquero activo sin club que tenga como vecino un motoquero del club `númeroDeClub` e ir hasta el mismo.
- Repetir sucesivamente los pasos 1 y 2 hasta que no haya más motoqueros activos en la convención que sean vecinos de otros con el club `númeroDeClub`.

Ejercicio d) El procedimiento `AsignarClub_AVecinos(númeroDeClub)` que marca todos los motoqueros vecinos al motoquero actual que estén activos y sin club como pertenecientes al club `númeroDeClub`.

Ejercicio e) La función `hayMotoqueroActivoYSinClubConVecinoEn_(númeroDeClub)` que indica si en la convención hay algún motoquero activo sin club todavía que tenga de vecino otro motoquero activo que tenga por club `númeroDeClub`.

Ejercicio f) `IrHastaElMotoqueroActivoYSinClubConVecinoEn_(númeroDeClub)` que posiciona el cabezal sobre un motoquero activo que no tenga club todavía y que tenga por vecino un motoquero en el club `númeroDeClub`.

Ejercicio 2) Escribir la función `clubConMásMotoqueros()` que determina cuál es el club que tiene más motoqueros activos en la convención. Se propone como estrategia marcar primero los clubes, y luego contabilizar la cantidad de miembros de cada club, para luego saber cuál tiene más. De esto surgen los siguientes procedimientos y funciones (nuevamente, realizar adicionales puede resultar útil):

Ejercicio a) El procedimiento `AsignarClubATodosLosMotoquerosActivos()` que asigna un club a cada motoquero activo de la convención.

Ejercicio b) El procedimiento `IrAAlgúnMotoqueroDelClub_(númeroDeClub)` que posiciona el cabezal en algún motoquero activo miembro del club `númeroDeClub`.

Ejercicio c) La función `cantidadDeMiembrosDelClubActual()` que determina la cantidad de miembros que posee el club del cual el motoquero actual es miembro. El cabezal debe encontrarse sobre un motoquero que tenga un club asignado.

AYUDA: estructurarlo como un recorrido sobre todos los motoqueros.