

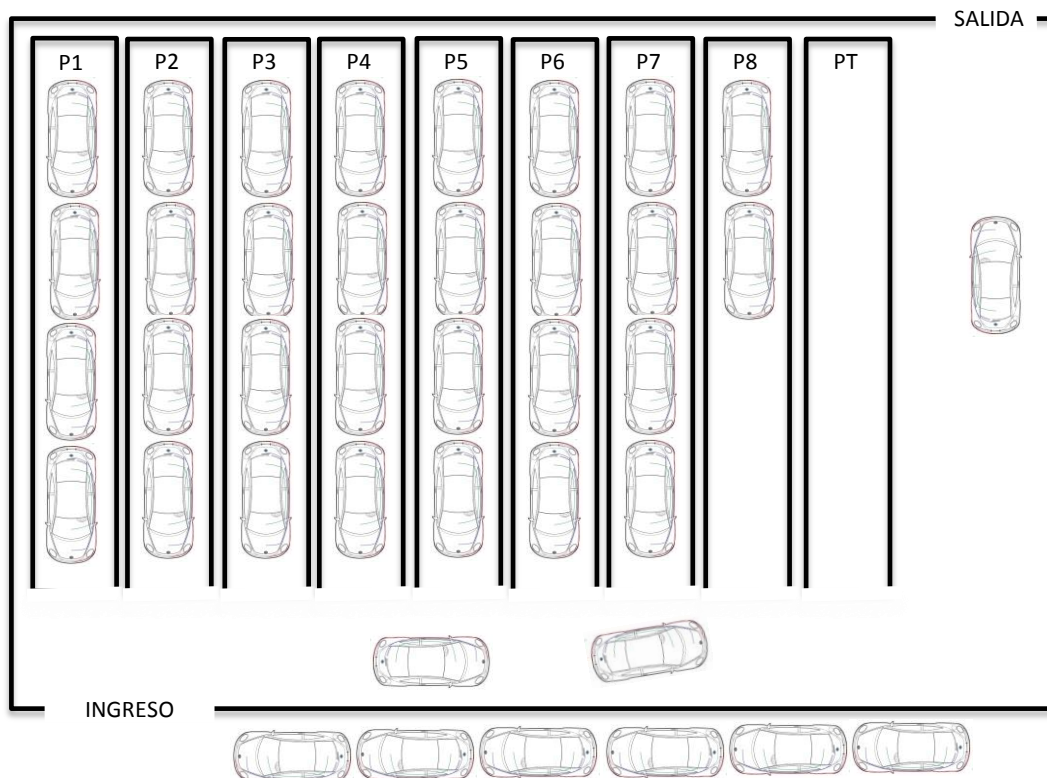
Taller 3: Colas y Pilas

Objetivos

- Aprender a diseñar y construir una cola
- Aprender a diseñar y construir una pila
- Familiarizarse con el uso de colas y listas como estructura de datos
- Aplicar el concepto y diseño de colas y listas en una aplicación

Descripción General

City Parking S.A.S ha decidido poner una franquicia en el centro de la ciudad y dadas las condiciones de espacio limitado no puede funcionar como tradicionalmente lo ha hecho. En ese sentido se decidió implementar una nueva dinámica para aparcar los carros que funciona de la siguiente manera:



Son ocho parqueaderos principales (P1, P2, ..., P7, P8) y un parqueadero temporal PT. Cada uno de los parqueaderos cuenta con cuatro lugares, que funcionan como una pila (LIFO). Si llega un cliente a parquear su carro, este debe ser registrado con la matrícula del vehículo, el nombre de quien conduce, el color del vehículo y la hora de registro en milisegundos (`getTime()`). En ese momento el cliente se puede ir y el parqueadero hace el resto.

Una vez registrado el vehículo, este queda en una cola (FIFO) aguardando para ser estacionado. Desde la central del parqueadero (en donde se administra todo), se puede dar la orden de estacionar el vehículo al interior de uno de los parqueaderos. Si al dar la orden no hay ningún lugar disponible en ningún parqueadero, el carro permanece en la cola y se debe esperar a que algún cliente recoja su carro y así se libere un lugar. Ahora bien, si se da la orden y en efecto hay lugares disponibles el carro debe ser estacionado en el primer lugar disponible encontrado.

Cuando un cliente va a buscar su vehículo, le facilita a la central la matrícula del mismo. Para localizarlo, se debe iterar sobre las pilas. Si después de iterar sobre las pilas el vehículo no se encuentra, es porque definitivamente no está estacionado en el parqueadero. Ahora bien, si el vehículo se encuentra en alguno de los parqueaderos, se deben sacar todos los carros que hay (`.pop()`) antes del mismo y se deben meter (`.push()`) en el parqueadero temporal (PT). Una vez se ha sacado el carro buscado, se deben sacar los carros guardados en el parqueadero temporal (PT) y se deben meter de nuevo en el parqueadero en el que estaban (Pn).

Lo que usted debe hacer

Usted debe implementar una aplicación capaz de satisfacer las siguientes necesidades:

- ✓ El dueño del parqueadero desea que desde la central se puedan registrar los clientes.
- ✓ También le ha pedido que haga posible dar las órdenes de parquear los carros y que determine en qué parqueadero van a quedar los mismos. Esto implica buscar que parqueaderos tienen cupo y dar la orden de parquear el carro en el primero encontrado con cupo.
- ✓ Le ha solicitado que desde la aplicación se pueda revisar el estado en el que se encuentran los parqueaderos y que en pantalla se ilustre gráficamente la situación de cada uno.
- ✓ La aplicación debe permitirle al usuario registrar la salida de un cliente, mostrar el tiempo en minutos que utilizó el servicio de parqueadero y calcular el valor a pagar. Tenga en cuenta que City Parking S.A.S tiene una tarifa de 25 pesos el minuto.
- ✓ Por último, la aplicación debe permitirle al usuario visualizar la cola de carros que aguardan para ser estacionados.

Parte 1 – Trabajo en casa

- 1) Descargue el taller e implemente sus estructuras de datos: tenga en cuenta que necesita las estructuras Cola y Pila y que deben ser genéricas para almacenar todo tipo de datos.

- 2) Implemente (complete) la clase **Carro.java**. Puede guiarse con la documentación que encuentra en la misma. Recuerde los atributos de cada vehículo (descritos en la sección “Descripción general”), No olvide tener en cuenta asignar a cada vehículo en su registro la hora de ingreso.

Parte 2 – Trabajo en clase

- 1) Implemente (complete) la clase **Central.java**. Puede guiarse con la documentación que encuentra en la misma.
- 2) Implemente (complete) la clase **Main.java**. Puede guiarse con la documentación que encuentra en la misma.

Entrega

1. Verifique que su proyecto cumple con los requisitos de la [entrega de talleres](#).
2. Entregue su taller por medio de **BitBucket (repositorio personal)**. Recuerde, si su repositorio no tiene el taller o está vacío, su taller no será calificado por más de que lo haya desarrollado.