



ECO-PCS

Programación Concurrente,
Informática

Cristopher Jason Ovaillos FAI-3162

Silvia Amaro

Año de cursada: 2022



índice

Recursos Compartidos	3
Hilos	3
Mecanismo	3
Explicación	4



Recursos Compartidos

Tipo	Nombre Clase
Principal	<ul style="list-style-type: none">• ECO-PCS
Actividades	<ul style="list-style-type: none">• SouvenirShop• PuestoSnorkel• SeccionComida• FaroMirador• CarreraGomones
Adicionales para Parque	<ul style="list-style-type: none">• Hora• EspacioCol• Salida

Hilos

Nombre Hilo	Uso
Visitante	El Visitante hace uso de los recursos de actividad
Reloj	El Reloj hace uso del recurso Hora
Asistente	El Asistente hace uso del recurso Snorkel
Tren Interno	El Tren hace uso del recurso CarreraGomones
Colectivo	El Colectivo hace del recurso ECO-PCS
Controlador	El Controlador hace uso del ECO-PCS

Mecanismo

Recurso	Mecanismo
ECO-PCS	→ Semaforos
SouvenirShop	→ Locks



PuestoSnorkel	→ Semáforos → BlockingQueue → Monitores de objetos
SeccionComida	→ Semáforos
FaroMirador	→ Semáforos
CarreraGomones	→ Locks → BlockingQueue
Hora	→ Monitores
EspacioCol	→ Monitores
Salida	→ Semáforos

Explicación

Controlador:

Esta clase realiza la creación de ECO-PCS la cual será nombrada como “parque”, en esta clase realizamos la creación y ejecución del Reloj, Colectiveros y los infinitos visitantes que se crearan solo si el parque se encuentra activo, además asignaremos el tipo de visitante que es (PARTICULAR O TOUR).

¿Como sabemos que los visitantes se crearan cuando el parque este activo?

```
public synchronized boolean isActive() {  
    while (!activo) {  
        try {  
            wait();  
        } catch (InterruptedException e) {  
            e.printStackTrace();  
        }  
    }  
    return activo;  
}
```

El hilo reloj es el que tiene el control de la hora, minutos y la variable activo, este hace las modificaciones de esta variable. Entonces cuando el parque este cerrado entran en un estado de espera hasta que Reloj que es el que avanza el tiempo vea la hora igual a 9 que es el momento de apertura y notifica a todos.

```
if (activo) {  
    notifyAll(); //si esta activo notifico todavia  
}
```



Metodo cerrar Parque:

```
public void cerrarParque() {  
    activo = false;  
    notifyAll();  
}
```

Visitante:

El visitante puede ser tour o particular, si este es particular puede viajar mediante un colectivo:

```
public void run() {  
    if (!tipoAcceso.equals("Particular")) {  
        parque.accionCole(id);  
    }  
    parque.Parque(this);  
}
```

accionCole() es un metodo que usa el visitante para subirse a algunos de los tres colectivos que existen estos son:

```
this.colA = new EspacioCol(SOUT, "CONCURRENTE");  
this.colB = new EspacioCol(SOUT, "SILVIA");  
this.colC = new EspacioCol(SOUT, "VICTORIA");
```

El visitante para entrar al parque debe esperar que este activo mediante

```
hora.isActiveo();// espero a que se pueda entrar, si esta cerrado espero
```

Este se queda esperando a causa de esto.

En el método Parque, es donde crearemos una variable random de 0-5 que indicara que actividad realizara de las disponibles, si este presenta ser el 5 entonces este ira del parque. Esto mediante un Switch que según sea el número nos dirigirá a un recurso con su propio mecanismo.

Reloj:

El Reloj hará uso de un método para avanzar el tiempo, este a través de este método notifica a los hilos que realicen el uso del método isActiveo() para que avancen. Los hilos que no son Reloj si o si se quedan esperando hasta que Reloj les avise que cambió la hora o el minuto.

Tren Interno:

Este único Hilo, realizara el transporte de los pasajeros hasta al inicio de la carrera en la actividad de CarreraGomones en el cual estarán presente los visitantes.

El Tren necesita si o si un mínimo de un Visitante adentro para avanzar, cuando tiene un pasajero les da un tiempo a los otros visitantes para que se suban de lo contrario este se va.

Esta operación 'accionesTren' realiza todo com que los pasajeros bajen y cuando bajen notificar que otros pasajeros puedan subir, realiza la simulación del tiempo que le tomara en llegar, además que el



Tren no podrá volver hasta que los pasajeros que estaban dentro no se bajen mediante una notificación que lo realiza el último pasajero que esta subido.

Asistente:

El asistente esta involucrado en la actividad Snorkel el cual esta encargado en la entrega de equipos, más bien notifica que hay equipos disponibles.

Metodo `notificarVisitantesEnEspera()` del recursos compartido Snorkel:

- Sincroniza la cola de espera para asegurar el acceso seguro a la misma.
- Si hay visitantes en espera, se obtiene el siguiente visitante en la cola y se le notifica que hay un equipo disponible.
- Utiliza `notify()` para despertar al visitante en espera.

Métodos que usan los visitantes en el `PuestoSnorkel`:

Metodo `solicitarEquipo(Visitante visitante)`:

- Sincroniza la cola de espera para asegurar que los visitantes se gestionen de manera ordenada. Si no hay equipos disponibles, el visitante se añade a la cola de espera y se le notifica que debe esperar.
- Cuando un equipo está disponible, el visitante lo adquiere (`equiposDisponibles.acquire()`) y se elimina de la cola de espera.
- Registra el evento de adquisición de equipo.

Metodo `devolverEquipo(Visitante visitante)`:

- Libera un equipo (`equiposDisponibles.release()`) para que otro visitante pueda utilizarlo.
- Registra el evento de devolución del equipo.
- Llama al método `notificarVisitantesEnEspera` para notificar a los visitantes en espera sobre la disponibilidad de un equipo.

Colectivero:

El colectivo esta involucrado en el parque ECO-PCS que lleva a los visitantes Tour.

El colectivero solo maneja si esta activo el parque.

Además, primero como el colectivo tiene acceso al recurso ECO-PCS se le asigna un colectivo de los tres que presenta el Parque, una vez asignado accede al recurso compartido que solo es para el llamado `EspacioCol` el cual presenta métodos para que los visitantes suban, bajen, además del método `manejarColectivo()`.