



IIC2115 – Programación como Herramienta para la Ingeniería (II/2020)

Actividad Práctica 1 - Programación Orientada a Objetos

Objetivos

- Aplicar los contenidos fundamentales de programación orientada a objetos para modelar el comportamiento en una autopista.

Entrega

- **Lenguaje a utilizar:** Python 3.6 o superior
- **Lugar:** repositorio privado en GitHub. Recuerde incluir todo en una carpeta de nombre **P01**.
- **Entrega:** jueves 20 de agosto a las 23:59 hrs.
- **Formato de entrega:** archivo python notebook (**P01.ipynb**) y archivo python (**P01.py**) con la solución de este enunciado. Los archivos deben estar ubicados en la carpeta **P01**. No se debe subir ningún otro archivo a la carpeta. Utilice múltiples celdas de texto y código para facilitar la revisión de su programa.
- **NO SE ADMITEN ENTREGAS FUERA DE PLAZO**
- **Entregas con errores de sintaxis y/o que generen excepciones serán calificadas con nota 1.0.**
- Las *issues* del Syllabus en GitHub son parte de este enunciado.

Introducción

Con el fin practicar las técnicas de POO, en esta actividad deberá modelar una autopista en la que circulan distintos tipos de vehículos. Esto significa, definir las **clases** correspondientes, sus relaciones y comportamientos. Una vez finalizado esto, **y sólo como tarea opcional**, deberá simular lógica de esta autopista.

Descripción del problema

Considere un tramo de 100 *km* de autopista, con dos pistas por sentido:

- Norte1: Pista norte izquierda rápida
- Norte2: Pista norte derecha lenta
- Sur1: Pista sur derecha lenta
- Sur2: Pista sur derecha lenta

En la autopista circulan vehículos cuyas llegadas se rigen por una distribución de *Poisson*, con tasas de llegada λ_n y λ_s desde el norte y sur, respectivamente. Siempre ingresan por la pista derecha de cada sentido. Si las llegadas se rigen por una distribución de *Poisson* de tasa λ , entonces, los tiempos entre llegada de vehículos son generados por la función exponencial `random.expovariate(lambda)` de la librería `random` de python.

Los vehículos pueden ser de tres tipos: livianos, pesados y de carrera, los que llegan en las proporciones p_l , p_p y p_c , respectivamente.

Cada vehículo fluye en la carretera a flujo libre a menos que se tope con otro vehículo más lento. En este caso, los vehículos pueden quedarse detrás del vehículo lento o adelantarlo con probabilidad $p_{adelantar}$ si es que fuera posible (Los vehículos pesados no pueden adelantar).

La velocidad de cada vehículo es constante y se determina mediante una variable aleatoria solo al ingresar a la autopista. Solo puede superarse esta velocidad en el proceso de adelantamiento (por la pista izquierda), luego de adelantar, debe volver a la velocidad inicial. Solo si el vehículo es de carreras, entonces no reduce su velocidad luego de adelantar. Si el vehículo decide quedarse detrás del vehículo, entonces debe mantener la velocidad del vehículo de adelante.

Modelo de clases

El modelo a implementar debe contener al menos los siguientes elementos:

- Una clase base donde se relacionarán sus objetos
- Cómo mínimo debe haber tres clases que hereden de otra.
- Una clase que participe como atributo en otra (indique en una celda de texto cuál sería)
- Una sobrecarga de métodos (*override* - polimorfismo)

Se recomienda fuertemente comentar el código y definir en celdas de texto qué es cada clase. Considere además que para definir las clases, sus relaciones y comportamientos, no es necesario tener desarrollada la simulación.

Opcional: Simulación

Para la simulación, considere la siguiente secuencia de instrucciones como posible guía:

```
# creación de variables y objetos relevantes
for t in range(t_max):
    if (llega un vehículo nuevo al sistema por el norte):
        # Se determina su tipo, su velocidad e ingresa a la vía
    else (viene del sur):
        # Se determina su tipo, su velocidad e ingresa a la vía
    # se avanzan todos los vehículos
    if si se topan
        # se evalúa el adelantamiento
```

Entregue información sobre el proceso cada vez que ocurre algún evento: Llegada de vehículos, y adelantamientos. No olvide utilizar clases para modelar las entidades.

Política de Integridad Académica

“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad.”

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por “copia” o “plagio” se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.