

Control 3  
Programación 2  
*Prof. Eduardo Godoy*

ESCUELA DE INGENIERÍA CIVIL INFORMÁTICA  
UNIVERSIDAD DE VALPARAÍSO

- *30pts.* Una empresa de autobuses le ha solicitado a Ud, como alumno de la Escuela de Ingeniería Civil Informática que desarrolle una simulación del proceso de subida y bajada de pasajeros. Los requisitos que le impone la empresa son:
  - La simulación se realiza sólo considerando un autobús
  - La capacidad máxima de un autobús es 45 pasajeros (sólo se consideran pasajeros sentados).
  - Si el autobús está vacío, no puede detenerse para bajar pasajeros.
  - Si el auto está lleno no puede detenerse para subir pasajeros.
  - La cantidad de pasajeros que suben y bajan es aleatoria entre 1 y 10.
  - La frecuencia de subida y bajada de pasajeros es aleatoria en 1 y 5 segundos.

**Condiciones de entrega:**

- **Debe compilar.**
- Debe considerar la creación de al menos dos thread para los proceso y un recurso compartido (clase que contiene los metodos sincronizados)
- Enviar al correo eduardo.gl@gmail.com **SÓLO** los archivos **\*.java**, eliminado de ante mano la instrucción **package**, en un comprimido ApellidoPaternoNombreC3.zip. El no cumplimiento del formato será penalizado con 10 punto de descuento.
- Entrega jueves 9 antes de las 2:00 am. Subir a la hora que se les indica en el certamen. El no cumplimiento con la hora de entrega, será penalizado con 1 punto de descuento por cada minuto de retraso.

¿Cómo será evaluado este Control?			
Tópico	Logrado	Medianamente logrado	No logrado
Tópico 1 - Clase Autobus.	<i>5pts</i> Crea la clase MonitorAutobus con sus atributos y constructor requerido.	<i>2pts</i> Crea clase con algunos atributo o métodos requeridos en el. Crea métodos o Atributos en otras clases no indicadas en el problema.	<i>0pts</i> No crea clases requeridas.
Tópico 1-a - Clase Autobus - Método de ingreso de pasajeros.	<i>20pts</i> Define e implementa correctamente el método relacionado <i>ingreso de pasajero</i> dentro de la clase MonitorAutobus obteniendo la salida requerida.	<i>10pts</i> Define e implementa método acercandose parcialmente a la salida esperada.	<i>0pts</i> No define método. Método definido pero no cumple con lo mínimo esperado.
Tópico 1-b - Clase Autobus - Método de salida de pasajeros.	<i>20pts</i> Define e implementa correctamente el método <i>salida de pasajero</i> dentro de la clase MonitorAutobus obteniendo la salida requerida.	<i>10pts</i> Define e implementa método acercandose parcialmente a la salida esperada.	<i>0pts</i> No define método. Método definido pero no cumple con lo mínimo esperado.
Tópico 2 - ThreadSalida	<i>20pts</i> Define e implementa la clase Thread para simular el ingreso de pasajeros con sus métodos y atributos requeridos.	<i>15pts</i> Define e implementa método acercandose parcialmente a la salida esperada.	<i>0pts</i> No define método. Método definido pero no cumple con lo mínimo esperado.
Tópico 3 - ThreadIngreso	<i>20pts</i> Define e implementa la clase Thread para simular la salida de pasajeros con sus métodos y atributos requeridos.	<i>10pts</i> Define e implementa método acercandose parcialmente a la salida esperada.	<i>0pts</i> No define método. Método definido pero no cumple con lo mínimo esperado.
Tópico 4 - Clase Inicializadora	<i>10pts</i> Define e implementa la clase de control que permite iniciar el proceso de simulación (clase impl) con su método main incluido.	<i>5pts</i> Define e implementa clase y método acercandose parcialmente a la salida esperada.	<i>0pts</i> No define método. Método definido pero no cumple con lo mínimo esperado.
Paradigma Orientación a Objetos	<i>5pts</i> Resuelve el problema utilizando el POO.	<i>3pts</i> Utiliza parte del POO para resolver el problema.	<i>0pts</i> No utiliza el POO para dar solución al problema.
Total máximo puntaje pregunta 2	<i>100pts</i>	<i>50pts</i>	<i>0pts</i>