



IIC2115 – Programación como Herramienta para la Ingeniería (I/2024)

## Laboratorio 1

### Aspectos generales

- **Objetivo:** evaluar individualmente el aprendizaje sobre el uso de técnicas de POO y estructuras de datos en un problema práctico de modelación y simulación.
- **Lugar de entrega:** Parte 1 lunes 25/03 a las 17:30, Parte 2 miércoles 03 de abril a las 23:59 hrs, ambas en el repositorio privado.
- **Formato de entrega:** archivos Python Notebook L1\_1.ipynb y L1\_2.ipynb con las soluciones de las partes 1 y 2 del laboratorio. Los archivos deben estar ubicados en la carpeta L1. Utilice múltiples celdas de texto y código para facilitar el trabajo del cuerpo docente. Entregas que no cumplan el formato tendrán un descuento de 0,5 ptos.
- **Entregas atrasadas:** el descuento por atraso para la Parte 1 es de 1 punto cada 10 minutos o fracción. El descuento por atraso para la Parte 2 es de 1 punto por cada hora o fracción.
- **Issues:** Las discusiones en las *issues* del Syllabus que sean relevantes para el desarrollo de la evaluación, serán destacadas y se considerarán como parte de este enunciado. Así mismo, el uso de librerías externas que solucionen aspectos fundamentales del problema no podrán ser utilizadas. Solo se podrán utilizar las que han sido aprobadas en las *issues*, previa consulta de los estudiantes.
- **Entregas con errores de sintaxis y/o que generen excepciones en todas las ejecuciones** serán calificados con nota 1.0.

# Invernadero inteligente

La crisis climática global está forzando la búsqueda de formas más sustentables y económicos para producir alimento, en particular para el consumo personal. Frente a esta problemática, usted ha decidido construir un invernadero y plantar algunos alimentos, aprovechando la oportunidad para automatizar algunos de los procesos de este.

Antes de embarcarse en la construcción del invernadero, usted decide realizar una simulación del comportamiento de este, con el fin de evaluar las potenciales ventajas de la automatización, utilizando Programación Orientada a Objetos y Estructuras de Datos.

## El invernadero

El invernadero corresponde a una estructura que alberga plantas, sensores y artefactos, en la que podrá realizar cosechas de insumos para poder alimentarse. El invernadero esta posicionado en el exterior, donde se conoce a toda hora la cantidad de **luz** y **temperatura**:

- La **luz exterior** corresponde a un indicador numérico que se mueve en el intervalo  $[0, 100]$ . Se obtiene a partir de la hora del día de acuerdo a las siguientes reglas:
  - 6:00 am - 12:00 pm: Crece linealmente de 0 a 100.
  - 12:01 pm - 7:00 pm: Decrece linealmente de 100 a 0.
  - 7:01 pm - 5:59 am: Se mantiene en 0
- La **temperatura exterior** corresponde a un indicador medido en grados Celsius. Se obtiene a partir de la medición de un sensor de luz  $L$ , a través de la fórmula  $L * 3/10 + aleatorio(-10, 10)$ , donde *aleatorio* corresponde a una variable aleatoria que se mueve entre -10 y 10 (use para esto la librería `random` de Python).

## Plantas

Las plantas corresponden a lo que se cultivará dentro del invernadero. Se dispone de variados tipos que presentan características diferentes. Todas deben tener el método `cosechar()` y deben estar identificadas con:

- **fecha\_plantacion**: Variable tipo *datetime* que almacena la fecha en que fue plantada. (use para esto la librería `datetime` de Python).

- **humedad\_optima:** Variable de tipo numérico (número real) que almacena la humedad óptima de la tierra para que la planta maximice la producción.
- **humedad\_actual:** Variable de tipo numérico (número real) que almacena la humedad actual en la tierra de la planta.

Además, las plantas deben ser de los siguientes tipos:

- **lechuga:** corresponde a una planta de lechuga que además incorpora la variable **n\_hojas** que guarda el número de hojas de la lechuga y la variable **tamano** que puede ser *‘grande’* (sobre 40 hojas), *‘mediana’* (entre 10 y 40 hojas) o *‘chica’* (menos de 10 hojas). El método cosechar extrae el número de hojas indicado en el método y debe actualizar correctamente el nuevo **tamano**.
- **tomate:** corresponde a una planta de tomates que además incorpora la variable **raza** que puede ser *‘normal’* o *‘cherry’*, la variable **n\_tomates** que guarda el número de tomates en la planta y la variable **acidez** que guarda el porcentaje de acidez del tomate. El método cosechar extrae el número de tomates indicado en el método y si se trata de un tomate cherry no debe permitir sacar tomates antes de medio día.
- **cilantro:** corresponde a una planta de cilantro que además incorpora la variable **n\_ramas** que guarda el número de ramas del cilantro. El método cosechar extrae el número de ramas indicado en el método.
- Planta definida por ud: tal como lo indica el nombre, debe definir una planta (real o inventada), con reglas y miembros propios, pero respetando la interfaz definida anteriormente.

## Sensores

Los sensores se instalan en el invernadero o en cada planta para medir parámetros. Todos los sensores almacenan siempre la **ultima\_medicion** y tienen el método **medir()**. Los tipos de sensores son:

- **Humedad:** este sensor se instala en cada planta para medir el nivel de humedad de la tierra. El método **medir()** debe retornar la humedad entre 0 y 100. Para simular la medición de humedad, reste un número aleatorio entre 1 y 30 de la **ultima\_medición**, considerando que mientras más plantas haya en el invernadero, más rápido disminuirá la humedad de cada planta.
- **Temperatura:** este sensor se instala a nivel de invernadero para medir la temperatura en el aire de este. El método **medir()** debe retornar la temperatura entre -30 y 50 Celsius. Para simular la medición de temperatura, sume o reste un número aleatorio entre 1 y 5, siempre buscando acercarse

a la temperatura exterior, considerando que mientras más plantas haya en el invernadero, más rápida aumentará la temperatura.

- Luz: Este sensor se instala a nivel de invernadero para medir la luz dentro del invernadero. El método `medir()` debe retornar la intensidad de luz entre 0 y 100. Para simular la medición de luz, reste un número aleatorio entre 5 y 10 de la luz exterior.

## Regador

Instrumento para regar una planta cuando se alcance una humedad límite en base a condiciones. Por ejemplo, una planta se debe regar a plena luz del día (luz superior a 60), cuando la temperatura sea superior a 10 grados y la humedad este por debajo de 12. **Para cada tipo de planta, cree su propia lógica (distinta de la anterior) para regarla hasta alcanzar su humedad óptima.** Considere finalmente que para cada día, el invernadero tendrá disponible una cantidad limitada de agua para el riego, por lo que no siempre las plantas podrán ser regadas.

## Crecimiento

Defina el método `crecer()` para cada tipo de planta, de modo que aumente bajo algún criterio el tamaño de esta, por ejemplo, cada X riegos. Entiéndase por tamaño, el número de hojas, tomates, ramas, etc.

## Parte 1: modelación

Realice la modelación completa de todos los elementos indicados anteriormente, utilizando POO y estructuras de datos. Defina e implemente clases y métodos, respetando la interfaz descrita. Para esta parte no es necesario instanciar objetos, solo la modelación de estos. Si bien se espera que los distintos métodos estén definidos e implementados, no es importante para esta parte que estos estén 100% correctos en cuanto a la lógica. Finalmente, no considere acá clases y comportamientos que tengan que ver con la simulación y la coordinación de los distintos objetos.

## Parte 2: simulación y consultas

Utilizando la modelación previamente realizada, desarrolle un programa que permita simular la situación descrita y le permita evaluar la idoneidad de los parámetros elegidos para el invernadero. Para ello, básiase en la estructura descrita a continuación:

```

# Defina su invernadero
# Defina sus plantas y sensores
h = 0 # Variable que lleva el conteo de la hora
dias_a_simular = 30 # Total de dias que va a simular
while h <= 30*24:
    # Determine los parametros externos
    # Evalúe los parametros
    # Coseche cuando se pueda

```

Durante cada simulación, deberá almacenar e imprimir en pantalla información sobre a la ejecución de esta. Deberá considerar al menos lo siguiente:

- El estado del invernadero y de las plantas en su interior.
- Eventos, como por ejemplo riego, crecimiento o cosecha.

Realice múltiples simulaciones, cambiando en cada una de ellas el estado inicial del invernadero y el número y proporción de las plantas que contiene. Una vez realizado esto, conteste las siguientes preguntas en base a los datos almacenados.

- Para cada tipo de planta, encuentre las 3 plantas que más cosecha entregaron.
- Encuentre las 3 simulaciones con más cosecha de tomate.
- Verifique si la hora de inicio de la simulación tiene impacto en la cosecha.
- Verifique si existe un tamaño óptimo para el invernadero (cantidad y proporción de plantas), en base a la cosecha.

## Corrección

Para la corrección de este laboratorio, se revisará el modelo de clases y la implementación de la lógica descrita. Para que obtenga una buena calificación, se le exigirá el uso de las siguientes estructuras y contenidos:

- Herencia
- Polimorfismo
- Listas y diccionarios

## Política de Integridad Académica

Los/as estudiantes de la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile deben mantener un comportamiento acorde a la Declaración de Principios de la Universidad. En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los/as estudiantes que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada estudiante conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Docencia de la Escuela de Ingeniería.

Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un/a estudiante para los efectos de la evaluación de un curso debe ser hecho **individualmente** por el/la estudiante, **sin apoyo en material de terceros**. Por “trabajo” se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros.

En particular, si un/a estudiante copia un trabajo, o si a un/a estudiante se le prueba que compró o intentó comprar un trabajo, **obtendrá nota final 1.1 en el curso** y se solicitará a la Dirección de Pregrado de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral.

Por “copia” se entiende incluir en el trabajo presentado como propio, partes hechas por otra persona. En caso que corresponda a “copia” a otros estudiantes, la sanción anterior se aplicará a todos los involucrados. En todos los casos, se informará a la Dirección de Pregrado de la Escuela de Ingeniería para que tome sanciones adicionales si lo estima conveniente.

También se entiende por copia extraer contenido sin modificarlo sustancialmente desde fuentes digitales como Wikipedia o mediante el uso de asistentes inteligentes como ChatGPT, Gemini o Copilot. Se entiende que una modificación sustancial involucra el análisis crítico de la información extraída y en consecuencia todas las modificaciones y mejoras que de este análisis se desprendan. Cualquiera sea el caso, el uso de fuentes bibliográficas, digitales o asistentes debe declararse de forma explícita, y debe indicarse cómo el/la estudiante mejoró la información extraída para cumplir con los objetivos de la actividad evaluativa.

Obviamente, está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, **siempre y cuando se incluya la referencia correspondiente**.

Lo anterior se entiende como complemento al Reglamento del Estudiante de la Pontificia Universidad Católica de Chile (<https://registrosacademicos.uc.cl/reglamentos/estudiantiles/>). Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.

### **Compromiso del Código de Honor**

Este curso suscribe el Código de Honor establecido por la Universidad, el que es vinculante. Todo trabajo evaluado en este curso debe ser propio. En caso que exista colaboración permitida con otros/as estudiantes, el trabajo deberá referenciar y atribuir correctamente dicha contribución a quien corresponda. Como estudiante es un deber conocer el Código de Honor (<https://www.uc.cl/codigo-de-honor/>).