

---

# Laboratorio 0: Configuración del Ambiente de Trabajo

## **Profesores:**

Carlos Andrés Lozano,  
Germán Adolfo Montoya,

## **Profesor de Laboratorio:**

Juan Andrés Mendez

## **1 Objetivo**

El objetivo de este laboratorio es que los estudiantes desarrollen las habilidades necesarias para configurar un entorno de trabajo adecuado que les permita implementar, resolver y analizar modelos matemáticos de optimización durante el curso. Los estudiantes aprenderán a instalar y configurar **Pyomo**, una librería clave para modelado matemático en Python, junto con diversos solvers de optimización como **GLPK**, **Ipopt**, y otros. Además, explorarán diferentes métodos de instalación y resolución de problemas comunes, asegurando que su entorno esté funcional y listo para las actividades prácticas y el proyecto final del curso.

## **2 Introducción**

**Pyomo** (Python Optimization Modeling Objects)<sup>1</sup> es un paquete de software de código abierto basado en Python que ofrece una amplia gama de capacidades para formular, resolver y analizar modelos de optimización. A diferencia de otros lenguajes de modelado algebraico como **AMPL**, **AIMMS** y **GAMS**, **Pyomo** integra sus objetos de modelado dentro de un lenguaje de programación completo, lo que proporciona

---

<sup>1</sup><https://www.pyomo.org/>

---

una rica variedad de bibliotecas de soporte.

Este paquete se destaca por su flexibilidad y capacidad para trabajar con una amplia gama de solvers, como GLPK, Ipopt, y CBC, lo que lo convierte en una solución integral para abordar problemas de optimización lineal, no lineal, entera-mixta y más. Además, Pyomo admite diversos tipos de problemas, incluyendo programación lineal, cuadrática, no lineal, programación estocástica y programación de restricciones generales, entre otros.

Gracias a su robustez y versatilidad, Pyomo es una herramienta esencial para investigadores, ingenieros y profesionales que buscan resolver problemas complejos de optimización y aplicar sus soluciones a escenarios del mundo real.

Existen diversas formas de instalar Pyomo y configurar su entorno de trabajo, lo que permite a los usuarios adaptarlo a sus necesidades y preferencias. Una de las opciones más comunes es la instalación local utilizando **Anaconda**, que facilita la gestión de paquetes y dependencias. Alternativamente, los estudiantes pueden optar por trabajar en la nube utilizando **Google Colab**, eliminando la necesidad de instalaciones locales y aprovechando un entorno preconfigurado. También es posible realizar la instalación en un **ambiente virtual de Python** en sistemas operativos como Windows o Linux, lo que permite un control completo sobre las versiones y bibliotecas utilizadas. Estos métodos aseguran flexibilidad y accesibilidad para implementar Pyomo de manera efectiva en cualquier entorno.

## 3 Guía de Instalación

### 3.1 Instalación con Anaconda

1. Descargue e instale Anaconda desde su **página oficial**.
2. Cree un nuevo ambiente con el siguiente comando:

```
1 conda create -n pyomo_env python=3.9
2 conda activate pyomo_env
```

3. Instale Pyomo y los solvers necesarios:

```
1 conda install -c conda-forge pyomo
2 conda install -c conda-forge glpk ipopt
```

---

4. Para instalar **Ipopt**, también puede usar:

```
1 conda install conda-forge::ipopt
```

#### **Notas adicionales:**

Existe una posibilidad de que los comandos anteriores no funcionen en su sistema. Para este caso se recomienda ir a la documentación del paquete de anaconda que están intentando instalar y utilizar un comando alternativo.

- Puede encontrar la documentación oficial de GLPK en: <https://anaconda.org/conda-forge/glpk>
- Puede encontrar la documentación oficial de **Ipopt** en: <https://anaconda.org/conda-forge/ipopt>

#### **3.1.1. Usuarios de Windows**

Para usuarios de Windows, es necesario abrir el **Anaconda Prompt** en modo administrador. Para ello:

- Busque **Anaconda Prompt** en el menú de inicio.
- Haga clic derecho y seleccione **Ejecutar como administrador**.
- Ejecute los comandos descritos anteriormente desde esta terminal.

#### **3.1.2. Preparar las herramientas de trabajo**

- En sistemas Windows, Anaconda Navigator incluye una versión de **VSCode**. Sin embargo, si ya tiene **VSCode** instalado, puede configurarlo para usar el ambiente de Anaconda como el *interpreter* de Python.
- Asegúrese de tener instalado el **plugin de Python** en **VSCode**.
- Cambie el *interpreter* de Python a la ruta del ambiente de Anaconda (**pyomo\_env**). Esto se puede realizar seleccionando el ambiente desde la barra inferior o el *Command Palette*.

---

## 3.2 Instalación en Google Colab

1. El notebook oficial se encuentra en el siguiente **enlace**. Sin embargo, para poder realizar la instalación solo necesitan ejecutar una unica celda en su Colab.
2. La celda de código se encuentra en la sección **1.0.2 Cloud Computing with Google Colab**.

```
1  ## Tip: Please put code like this at the top of your notebook.
2  # We want all of the module/package installations to start up front
3
4  import sys
5  if "google.colab" in sys.modules:
6      !wget "https://raw.githubusercontent.com/ndcbe/CBE60499/main/
7          notebooks/helper.py"
8      import helper
9      helper.install_idaes()
10     helper.install_glpk()
11     helper.install_ipopt()
```

**Nota:** Este script también tiene la opción de instalar otros solvers como lo son: **cbc**, **bonmin** y **couenne**. Los cuales no vamos a utilizar por el momento pero serán de gran utilidad en un futuro.

## 3.3 Instalación en un ambiente virtual de Python

### 3.3.1. Instalación en Ubuntu

1. Instale Python 3 y pip:

```
1  sudo apt-get update
2  sudo apt-get install python3 python3-pip
3  sudo apt-get install swig ipython python-dev python-numpy
   python-scipy python-matplotlib --install-recommends
```

2. Cree un ambiente virtual:

```
1  python3 -m venv pyomo_env
2  source pyomo_env/bin/activate
```

3. Instale Pyomo y los solvers necesarios:

```
1  pip install pyomo
2  sudo apt-get install glpk-utils
3  sudo apt-get install coinor-libipopt-dev
```

---

### 3.3.2. Instalación en MacOS con Homebrew

1. Instale Homebrew desde su **página oficial**.

2. Instale Python 3 y pip:

```
1 brew install python3
```

3. Cree un ambiente virtual:

```
1 python3 -m venv pyomo_env
2 source pyomo_env/bin/activate
```

4. Instale Pyomo y los solvers necesarios:

```
1 pip install pyomo
2 brew install glpk
3 brew install ipopt
```

## 3.4 Instalación en un PC de la Universidad

1. Abrir *Anaconda Prompt* y cree un nuevo ambiente con el siguiente comando:

```
1 conda activate pyomo_env
```

2. Instale Pyomo y los solvers necesarios:

```
1 conda install -c conda-forge pyomo
2 conda install -c conda-forge glpk ipopt
```

3. Para instalar Ipopt, también puede usar:

```
1 conda install conda-forge::ipopt
```

## 4 Instalación de Librerías Adicionales

Para garantizar el correcto funcionamiento de los scripts y herramientas necesarias para este laboratorio, es importante instalar las siguientes librerías adicionales. Cada una cumple un rol específico en los códigos que se utilizarán:

---

Librería	Uso
numpy	Manejo de arreglos y operaciones matemáticas eficientes.
pandas	Manipulación y análisis de datos estructurados en tablas.
matplotlib	Generación de gráficos para visualizar resultados.
networkx	Análisis y representación de grafos y redes.

Cuadro 1: Librerías necesarias y sus usos.

## 4.1 Instrucciones de Instalación

Dependiendo del gestor de paquetes que utilice, puede instalar estas librerías con los siguientes comandos:

- Usando **pip**:

```
1 pip install numpy pandas matplotlib networkx
```

- Usando **conda**:

```
1 conda install numpy pandas matplotlib networkx
```

**Nota:** La versión completa de **anaconda** y **Google Colab** ya incluye estas librerías. Solo es necesario instalarlas si está utilizando un ambiente virtual de Python o si instalaron miniconda.

## 5 Entregables

Elija **uno** de los métodos de instalación descritos anteriormente y siga los pasos para instalar Pyomo y los solvers necesarios. Una vez instalado:

- Ejecute los archivos que se encuentran en el contenido del curso en la ruta: Labs/Lab0. Estos archivos son:
  - lab0\_test\_lp.py
  - lab0\_test\_mip.py
  - lab0\_test\_nlp.py
  - lab0\_test\_additional\_libs.py

---

**Nota:** Si la ejecución de los cuatro anteriores archivos no generaron errores, significa que la instalación de Pyomo y sus librerías fue correcta.

El entregable consistirá en un documento PDF con el nombre `lab0_nombre_apellido.pdf`, que contenga:

- Capturas de pantalla del proceso de instalación incluyendo:
  - Instalación de Pyomo y los solvers.
  - Ejecución de los cuatro \*.py suministrados anteriormente (los que están en `Labs/Lab0`). La captura debe ser de lo que aparece en la consola de resultados del IDE empleado (Spyder, VScode, entre otros).
- Comentarios breves sobre cualquier error encontrado y cómo lo solucionó (si aplica).
- **Plazo máximo de entrega:** Miércoles 29 de Enero hasta las 11:59pm.
- No se reciben entregas por fuera del plazo máximo y tampoco por correo. Las entregas solo se reciben por **Bloque Neón**.
- Este laboratorio se debe entregar de manera individual.