# Corte de Proyecto de Programación Avanzada

## Tema

Ejecución continua de algoritmos de análisis de datos

## Participantes

Edgar Diego Molina Trofimova

Roberto Sarmiento Trejo

## Descripción breve del proyecto

El proyecto consiste en crear una plataforma para la ejecución continua de algoritmos de análisis de datos a partir de la relación entre las entradas y salidas de dichos algoritmos (las entradas de unos son las salidas de los algoritmos anteriores). Esto lo haremos basándonos en el patrón de diseño Pipeline (de tuberías).

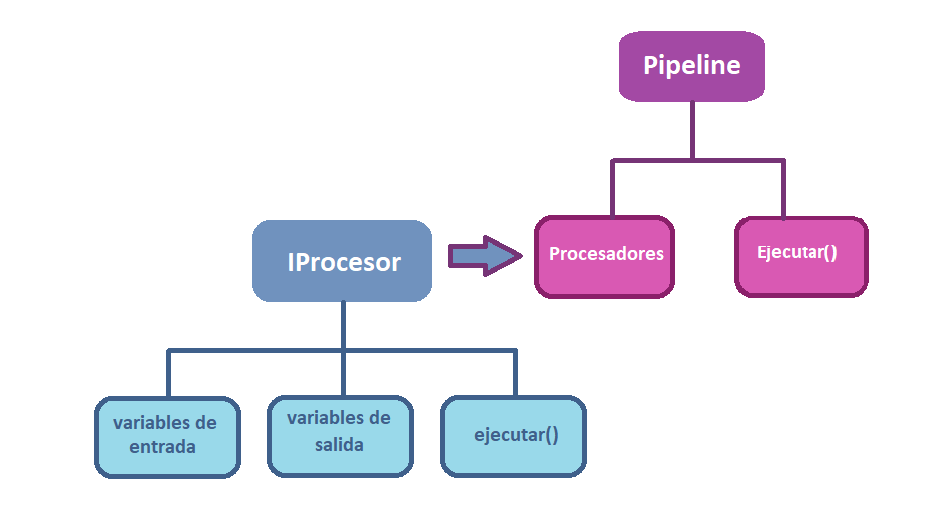
Un Pipeline no es más que una clase que contiene funciones específicas concatenadas de tal forma que la salida de funciones *padres* sean las entradas de funciones *hijas*. Esta estructura permite añadir y/o eliminar este tipo de funciones específicas, y a su vez darle un orden de ejecución deseado.

El Pipeline que modelaremos permitirá ejecuciones en paralelo, por lo que lo hará funcional en todo el tiempo de ejecución, evitando así de que, si en una ejecución del pipeline hubo alguna interrupción o error, se detengan otras ejecuciones.

## Requerimientos del proyecto

* Implementar un pipeline teniendo en cuenta dos partes fundamentales del mismo: su ejecución y su almacenamiento.

## Concepción de la solución



La clase Pipeline contendrá 2 elementos fundamentales: una lista de procesadores, los cuales serán las funciones específicas, las cuales cumplirán con la interfaz IProcesor. Como vemos, los procesadores tienen un método ejecutar, que la función propia del procesador, y dos campos principales: una lista con los nombres de las variables de entrada del procesador, y otra lista con los nombres de las variables de salida que debe tener ese procesador.

El Pipeline también tendrá su método ejecutar (), el cual se ejecutará en un hilo.