

## Construcción de Compilador YAPL

### Generación de Ensamblador

La generación de código, última fase del diseño de nuestro compilador. Al haber traducido el lenguaje de alto nivel a una representación intermedia, junto con la generación de las direcciones de memoria y la optimización de temporales queda como última fase del proyecto la traducción de dicho código intermedio en código assembler (o cualquier código objeto que pueda luego interpretarse para su ejecución). Los objetivos principales de esta fase del proyecto son:

- Implementar algoritmo de generación de secuencias de llamadas y secuencia de retornos de procedimientos.
- Implementar el algoritmo de asignación de registros o uso de pila, dependiendo de la arquitectura de máquina que estén generando.
- Generar código assembler o similar para su posterior ejecución por medio de una tercera herramienta.

El objetivo principal de este proyecto es la generación del código que se pueda ensamblar y que realice correctamente la ejecución, semánticamente hablando. Es posible implementar la optimización del código intermedio o del código generado sin embargo primero asegúrense que se realice una correcta traducción para luego realizar las optimizaciones que consideren pertinentes.

### ¿Qué entregar?

- Interfaz de Usuario
  - Que permita la escritura de programas en YAPL y despliegue del código assembler generado
- Ejecución exitosa del código assembler generado para las pruebas propuestas.

### Sugerencias

- Realicen el mapeo de sus instrucciones de código intermedio a código assembler para tener la guía a utilizar durante la programación.
- Realicen pruebas de su generación en forma incremental, es decir empiezan traduciendo programas simples hasta llegar a lo más complejo.