# LLM4Decompile Setup

* Install and set up a virtual machine with Oracle VM Virtual Box (Ubuntu/Liger Labs)
* Download Anaconda
* Install Ghidra 11
* Install OpenJDK17: winget install EclipseAdoptium.Temurin.17.JDK
* Clone the repository for LLM4Decompile:

git clone https://github.com/albertan017/LLM4Decompile.git

cd LLM4Decompile

conda create -n 'llm4decompile' python=3.9 -y

conda activate llm4decompile

pip install -r requirements.txt

# Workflow

* When using the V2 model, the reverse engineer should start with auto-analyzing a binary in Ghidra—identifying and labeling functions with generated names while maintaining the structure of the code, then pinpoint the primary entry point, exported functions and find heavily interconnected functions using call graphs and cross-reference tools.
* Export the pseudo-code into a file and input into the V2 model by changing the func\_name and filename variables in the script to fit appropriately. Then run the script within a terminal.
* The script is “decompile-ref”