

Assignment 1: Install and Build environment (4%)

Due: Jan. 26, 2022

Install Ubuntu OS

There are two possible options to install Ubuntu 20.04 if you are using Windows/MacOS. One is dual-boot linux and the other is to install VirtualBox on your OS. If you wish to do so, you can also choose to work on a cloud desktop (e.g. AWS, Azure, etc.).

(a) Dual-boot Ubuntu

You could follow the instructions on [this website](#). The latest stable version is [Ubuntu 20.04.1 LTS](#)

(b) VirtualBox

VirtualBox could be downloaded [here](#). Then install Ubuntu OS in VirtualBox shown [here](#).

Install assignment dependencies

In order to make sure our skeleton code runs on your system, we require you to use ANTLR4 4.9, LLVM 11.0, and Clang 11.0.

Install cmake and antlr4-runtime from source:

```
# Install required tools
sudo apt-get install cmake uuid-dev pkgconf
# download source
mkdir antlr-src && cd antlr-src
wget http://www.antlr.org/download/antlr4-cpp-runtime-4.9-source.zip
# unzip file
unzip antlr4-cpp-runtime-4.9-source.zip
# create build directory for cmake
mkdir build && cd build
# create makefiles
cmake .. -DCMAKE_INSTALL_PREFIX=/usr/local -DANTLR4_INSTALL=True
# build and install
make
sudo make install
```

Possibly, you may need llvm 11.0 instead of default llvm. The detailed instructions are shown in [here](#).

```
# For convenience there is an automatic installation script that installs LLVM for you.
# To install LLVM 11:
wget https://apt.llvm.org/llvm.sh
```

```
chmod +x llvm.sh
sudo ./llvm.sh 11
```

Then add llvm location to your \$PATH.

```
export PATH=/usr/lib/llvm-11:$PATH
# Or you may add the line to your .bashrc
echo 'export PATH=/usr/lib/llvm-11:$PATH' >> ~/.bashrc
```

Compile skeleton codes

Download code.zip from Quercus and unzip it. In the code root directory, do the following commands to try to compile:

```
mkdir -p build
cd build
cmake ..
make
```

If you find that the `find_package` is giving you errors, then try manually setting the ANTLR directories:

```
set(ANTLR4_INCLUDE_DIR /usr/local/include/antlr4-runtime)
set(ANTLR4_LIB_DIR /usr/local/lib)
```

Deliverables

On Markus, please submit:

- Submit a screenshot that shows the `cmake ..` and `make` command succeeding for the skeleton code. As an additional sanity check, you can try running the executable (`src/minicc`).
- Two sample valid MiniC programs (10-50 lines of code) and the descriptions of the two sample programs. Recall that MiniC is a subset of C, so a valid MiniC program will also be a valid C program. You can check that your program is valid by compiling against Clang and checking that it meets all the specifications of the MiniC language.

If you have any questions or issues, please post on the course Piazza website.