Instruction to run Inline pass

To compile and run the pass:

1. Install latest LLVM by “git clone --depth 1 https://github.com/llvm/llvm-project.git"
1. cd llvm-project
2. mkdir build
3. cd build
4. cmake -G Ninja ../llvm
2. Copy HelloWorld.h to llvm/lib/Transforms/Utils/HelloWorld.h
3. Copy and Replace HelloWorld.cpp to llvm/lib/Transforms/Utils/HelloWorld.cpp
4. Add the “FUNCTION_PASS("helloworld", HelloWorldPass())” to llvm/lib/Passes/PassRegistry.def in the FUNCTION_PASS section
5. Add "#include "*llvm/Transforms/Utils/HelloWorld.h”” to* llvm/lib/Passes/PassBuilder.cpp
6. Compile the pass in the llvm-project directory with “ ninja -C build/ opt”

To run the tests:

1. First use the following command to your target .c file. The file must be in the correct path, for example if you are at llvm-project directory and the .c file is in test_codes/example0.c, then the path should be test_codes/example0.c. It will generates a .ll file.
1. clang -O0 -Xclang -disable-O0-optnone -emit-llvm test_codes/example0.c -S -o test_codes/example0.ll
2. Then(again with the correct path). This will create a .bc file.
1. build/bin/opt test_codes/example1.ll > test_codes/example1_inline.bc -passes=helloworld
3. Disassemble this .bc file using. This will creates another .ll file, which is the file we need.
1. llvm-dis test_codes/example1_inline.bc

If any command doesn’t work, you can try “export PATH="/usr/local/opt/llvm/bin:$PATH””