

Report for LLVM Project Part 1-InstCounter

Wei Zhang

Description of the Implementation

In the InstCounter.cpp, there are only two functions which need to be altered, add_init_call and count_dynamic_bbs.

For the count_dynamic_bbs function, I iterate over basic blocks and iterate over instructions inside. In the instructions' iterations, I insert the "increase" function call before the current instruction. If the current instruction is a Alloca Instruction or PHInode, I will skip that instruction.

For the add_init_call function, if the name of the current function is "main", I will take iteratively skip all the instructions which are Alloca or PHInode instructions. Then I add the "init" function call before the first instruction of the main function.

I also transform the InstCounting.c to make the output correctly formatted by add a "\n" before the Total instructions executed: %d\n

Test result:

```
root@e12d19763381:/cs255-llvm-pass/test# ./test
```

Enter the string :cs255

Reverse string is :552sc

Total instructions executed: 77

I transform the test file to a new test case.

The test result is :

Function: welcome

No. 0bb: 4 statements.

Function: calculator

No. 0bb: 7 statements.

No. 1bb: 5 statements.

No. 2bb: 4 statements.

No. 3bb: 5 statements.

No. 4bb: 4 statements.

No. 5bb: 5 statements.

No. 6bb: 4 statements.

No. 7bb: 5 statements.

No. 8bb: 3 statements.

No. 9bb: 2 statements.

Function: main

No. 0bb: 13 statements.

No. 1bb: 4 statements.

No. 2bb: 25 statements.

No. 3bb: 5 statements.

```
/LLVM_ROOT/build/bin/llc -filetype=obj test.bc.opt -o test.o
```

```
/LLVM_ROOT/build/bin/clang test.o ../build/runtime/libInstCounting_rt.a -o test
test.o: In function `main':
test.bc.opt:(.text+0x1e2): warning: the `gets' function is dangerous and should not be used.
root@e12d19763381:/cs255-llvm-pass/test# ./test
```

```
welcome to the program: cs255
Enter the string :cs255
```

```
Reverse string is :552sc
255 - 128 = 127
Total instructions executed: 102
```