Advanced Compiler Optimizations-Programming Assignement 2

Shruti Wasnik (Roll no :20111062)

November 2020

Key points

- For the instructions with ϕ function, the operands in both lhs and rhs are added into the *kill* set of current basic block and each operand in rhs has been made live along the edge from the corresponding predecessor block.
- All the program points in each basic blocks are considered. Although, for instructions with ϕ function, the operands are not considered as *gen* and live variables are calculated as per the pervious point.
- Program point `0` refers to the program point at `IN[B]`, i.e., at the `IN[first instruction of BB]`, program point `1` refers to the OUT[first instruction of BB,] and so on.

Commands

- To compile and generate .ll file:

```
clang -Xclang -disable-O0-optnone testcase1.c -O0
-S -emit-llvm -o  testcase1.ll
```

- To create a new .ll file with phi functions from testcase1.ll file:

```
opt testcase1.ll -mem2reg -instnamer -S -o
testcase1_phi.ll
```

- Generate .bc file from testcase1_phi.ll:

```
llvm-as testcase1_phi.ll -o testcase1.bc
```

- Run pass on testcase1.bc:

```
opt -load  {path/to/llvm-project}/build/lib/
LLVMLiveVariableAnalysis.so -instnamer
-LiveVariableAnalysis <testcase1.bc> /dev/null
```