Homework H3

1 Description

Write an LLVM pass starting from the code you have developed for H2.

The goal of this new pass is to compute IN and OUT sets of reaching definition data-flow analysis **for the CAT language** starting from the GEN and KILL sets you've already defined for H2. As it was the case for H2, even for this homework the definitions you need to analyze are only those that define CAT variables.

You need to define the IN and OUT sets for every instruction of a program given as input. At the end of your pass, you need to have stored all IN and OUT sets in your data structures. Before ending your pass, you need to print IN and OUT sets for each instruction.

1.1 Assumptions

For the H3 homework, you can take advantage of the following assumptions about the C code that invokes CAT functions.

1. A variable used to store the return value of `CAT_create_signed_value` (i.e., reference to a CAT variable) is defined statically not more than once in the function it has been declared.

2. A variable that includes a reference to a CAT variable does not get copied to other variables.

3. A variable that includes a reference to a CAT variable does not get copied in any data structure.

4. A variable that includes a reference to a CAT variable does not escape the function where it has been declared.

2 Testing your work

To test your work, go to a test you have available:

```
cd H3/tests/test0
```

set the path of your pass in `LLVMPASSPATH` of Makefile.

Now, invoke your pass

```
make
```

Check the output generated by your pass against the oracle output:

```
make check
```

If you've passed the test, you'll see the following output:

```
Test passed!
```

Otherwise, you'll see the following:

```
Test failed
Output differences can be found in "./diff/"
```

Good luck with your work!

3 What to submit

Submit via Canvas

- The C++ file you've implemented (CatPass.cpp)

For your information: my solution for H3 added 60 lines of C++ code to H2 (computed by `sloccount`).

4 Homework due

10/28