

Techniques for Improving Software Productivity

16/17 Fall Semester

Homework Assignment 5: Abstract Interpretation

The Python code used in class to demonstrate chaotic iteration and abstract interpretation can be found at: https://bitbucket.org/tausigplan/soft-prod16/src/master/demos/chaotic_iteration/

The skeleton code for this assignment can be found at: <https://bitbucket.org/tausigplan/soft-prod16/src/master/exercises/ex5/>

1. Implement the abstract domain for may-be-garbage analysis that was described in class. The code outline can be found in the file `mbg.py`
2. Demonstrate the analysis on two interesting examples where the analysis gives a precise result, in files `mbg_prog1.py` and `mbg_prog2.py`
3. Create in `mbg_prog3.py` an example where the analysis loses precision, and gives a false alarm. This means the analysis indicates that some variable may be garbage, when in fact it is not garbage in any program execution

Submission Guidelines

You should submit 4 python files – `mbg.py`, `mbg_prog1.py`, `mbg_prog2.py` and `mbg_prog3.py`, as well as a pdf explaining your programs and the results of running abstract interpretation on all 3 programs.

The abstract domain you write should be entirely contained in the file `mbg.py`. You can add as many functions as you like, but **you should not change any of the skeleton code** that is in the file as it appears in the repository.

Note that the abstract domain you write in `mbg.py` will be tested against various program, not just the programs you submit, and should work correctly on all. You are encouraged to test your abstract domain on as many examples as you can think of.