COP 5021 — Program Analysis January 4, 2010

Homework 1: Introduction to Program Analysis

See the syllabus and listen in class for the due dates.
In this homework you will get an overview of program analysis and make some initial plans for your semester project.
If you wish, you can work in groups, and I recommend that for the semester project. However, but be sure to follow the process described in the course's grading policy if you work in groups.
Turn in this homework on paper during class.

Read chapter 1 of our textbook: *Principles of Program Analysis* [1]. There are also some online resources about program analysis listed on the course resources web page, `http://www.eecs.ucf.edu/~leavens/COP5021/resources.shtml`. You can also look for new books or online resources that are interesting. (Hint: to find new resources, look in Chapter 1 of the textbook for terms to search for. You may find more material under the specific approaches, such as data flow analysis or abstract interpretation, than under the general term "program analysis.")

1. [Concepts] Give a very brief answer (a sentence or two, at most a paragraph) to answer the following.
   (a) (10 points) What are the main ideas and goals of program analysis?
   (b) (10 points) Aside from optimizing compilers for programming languages, briefly describe one other area of computer science in which the ideas and goals of program analysis might be usefully applied, and how the ideas might apply in that area.

2. (20 points) [ImproveTools] Write a short proposal (about 2 pages in length) for a semester project that involves writing a static analysis tool to answer some question about software. This proposal should describe the problem you wish to solve (including the programming language your tool will analyze), and why it is interesting. You should also discuss any related work on this problem that you are aware of and why it does not solve the problem. Then describe, as precisely as you can, what question your tool will answer. Give one or two brief examples to illustrate the question. Explain why this question can be answered statically (without executing the code).

   It is best if you restrict yourself to an intraprocedural analysis; that is, it is best if your analysis does not need to use analysis results from outside of a given procedure (or method) when doing analysis on that procedure (or method).

   The programming language you analyze *cannot* be the WHILE language of the textbook [1]. However, it may be a very restricted (small) subset of a real language. Also, it will be best if you work with a language that is a first-order language (without procedure or function values), like the WHILE language of the textbook, because handling higher-order languages (e.g., Scheme or Haskell) is a topic that we will likely not cover in detail this semester (but see chapter 3 of the textbook).

   You will meet with the instructor to discuss this proposal, as a group if you are working in a group.

References

[1] Flemming Nielson, Hanne Riis Nielson, and Chris Hankin. *Principles of Program Analysis*. Springer-Verlag, 1999.