

CS6110/6116 Problem Set 4 Due Friday April 6, 2012  
A Design Assignment

The task is to design an extension of the simple imperative programming language IMP and explain it using ideas from the course. The extension is to add recursive procedures in the conventional imperative style, i.e. inputs are supplied using locations, as discussed in Lecture 25 and in notes about PL/CV procedures posted with Lecture 26. Ambitious students could treat a set of mutually recursive procedures.

Here are the subtasks:

1. Extend the IMP syntax to include procedure definitions and (recursive) calls. The main program can be considered as a special procedure, called main.
2. Give two examples of recursive procedures.
3. Write a "textbook" style explanation of the semantics and proof rules.
4. Provide a big step operational semantics.
5. Design a partial correctness style rule using Assume and Attain statements.
6. Design a total correctness rule in the style we studied for while loops.
7. Compare the notion of a recursive procedure to that of a recursive function in PL/CV (Lecture 26 notes). Note why procedures are simpler.
- 8.\* For extra credit discuss the impact of external (or global) variables.