Milestone #3: Code Generation

Overview

For this milestone, your program must generate valid Sparc assembly for each input program. This requires that your program translate from its interal ILOC-like representation into actual Sparc assembly, generate appropriate activation record management instructions (i.e., `save` and `restore`), compute stack offsets, and allocate registers.

Part 1: Instruction Selection

Generate Sparc assembly with appropriate activation record management and stack offsets for loads and stores. For this part you need not allocate registers.

Part 2: Register Allocation

Complete the implementation of this milestone by implementing a global register allocator. The resulting output from this stage of the milestone should be valid Sparc assembly that can be assembled, linked, and executed.

Command-line Options

You may find it useful to provide a command-line option to prevent register allocation. This will allow you to view the assembly code just prior to allocation which can help a great deal in both debugging and tracing any changes made by subsequent phases (if you were to implement any).