

Report

High Level Architecture

Repository structure mirrors hierarchy of MIPS objects: - Program - Function - Instruction - Operand

The program has a print function that hierarchically calls print on instructions children, e.g. Function will print instructions. The MipsFunction class additionally sandwiches the translation of instruction list with a prologue and epilogue, which takes care of the stack and saved variables.

Low Level Architecture

Certain instructions are dependent on certain phases of the backend compilation. For example, for greatest efficiency, call functions in the IR can only be translated to a concrete MIPS code form after the number of saved registers and temp registers are known. Hence, the representation of the code in this compiler utilizes references between objects (e.g. instructions and the parent function), which allows the instruction to exist in a partially complete state while maintaining references to the locations that information necessary for the final code transformation will be stored in the future.

A lot of these instructions demand more virtual registers than given in the original IR. So, I made a function that adds virtual registers, and tracks them so that they will get a position on the stack. This meant that the stack frame layout of a function could not be known until after instruction selection, interestingly.

Challenges

Main issue I had was time lol. I did not have enough time to give this project my best shot due to other coursework, tutoring, and research. Besides that though, the main challenge was learning to mentally translate the produced assembly back into a semantic understanding in order to debug more effectively.

Also, I worked alone, which didn't help much. In hindsight I should have found a team to work with much earlier.

Bugs

BFS currently fails for unknown reasons. Additionally, the greedy algorithm doesn't give improvement.

Extra tests

I did not create any extra tests for performance, but I did produce one for debugging the array assignment IR instruction translation, since that was missing from the sample IRs given in the repo.