COMP 506, Spring 2019

Project 2 Questionnaire

Name: \_\_\_\_\_\_\_\_\_\_\_Nai-Fan Chen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

NetId: \_\_\_\_\_\_\_\_\_\_\_\_\_\_nc41\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Briefly discuss your experience building the ILOC code generator for Demo. What things were easy to do? What things were hard?

Making the symbol table and implementing a hash table are easier than other parts.

The most difficult part is if else since I cannot directly add my emit function in it. If I do so, it caused syntax conflict. Therefore, I separated if else into four different parts to generate code. Building a new non-terminal is difficult since it need to have correct grammar without redundancy.

1. Are there topics that we should cover in class to better prepare you for this project?

Segmentation elaboration, AHSDT and array are very useful in this project especially the fake zero. Lots of theories in the course help me build this lab. However, the grammar of the DEMO needs to be more clear. The pdf file is not very comprehensive. If we can have some elaborations of it in class, it will help us save time spending on guessing the grammar and asking ambiguity on piazza.

1. What, if anything, would you do differently if you could start over on this project?

I would like to reduce the usage of registers as many as possible. I found that br and cbr will need more cycles than other operations, so I would like to replace them with other representations as well. Sperating stmt into smaller pieces like whilestmt, forstmt since it may have more flexibility to generate efficient code.