Report 3

Gregor Robertson

27 Oct 2016

Progress This Week

Over the course of the past week I have not gotten as much done as I would have liked - something that I will rectify for the next week. So far I have written a small LLVM pass and have started to convert my reporting between using Libre Office Write and using LaTeX.

The LLVM pass I wrote is based on the skeleton pass provided by [1]. The original pass was an extension of the Function Pass and would print out the name of the function. I converted it so that it would loop over all of the instructions in the function and print random metadata strings next to them. While this is obviously not very useful, it was simply to understand how to print the metadata next to each instruction. I have tried to convert the pass to use Module and not Function, but this was not successful - so far. The reason that I am converting the Function Pass to be a Module Pass instead is because a Function Pass is not allowed/supposed to maintain data between each function that it is run on, something I imagine may be useful as, at the moment, I think that reading in the information from a file would be the best way to get the information to the compiler.

As I mentioned above I also started to use LaTeX to write these weekly reports. This was mainly to get practice when using LaTeX during the main report I will have to write at the end of the year. By doing this I will also have all of my references stored in a main bibliography in the correct format which should reduce the time I spend on them later on.

I have also had a think about the different possible ways that I could let a programmer input their data for us to use. As it stands, given what both of you have said, I think that it might be best to go with some kind of preprocessor which will write the information to a file, then an LLVM pass which will take that information and apply it to the instructions of the IR. [**howToAddAnAttribute**] talks about adding an attribute but seems to suggest that to add an attribute you have to edit the CLANG code itself, however [**clangDoc**] suggests that there is a way to add pragma handlers without having to edit the CLANG code base.

Aims For Next Week

For next time I am hoping to do a few things:

- Decide upon the method that the programmer will input the extra information required for the LLVM pass. `Medium`

- Get the Module Pass working. `Hard`

- Finish getting the reporting set up `Easy`

- Set up a repository on GitHub for the project `Easy`

Questions

For now I don't have any questions about the project. If you have any suggestions, comments or requests I would be happy to hear them but at the time of writing this report I do not have any questions.

References

[1] sampsyo. *llvm-pass-skeleton*. URL: https://github.com/sampsyo/llvm-pass-skeleton.