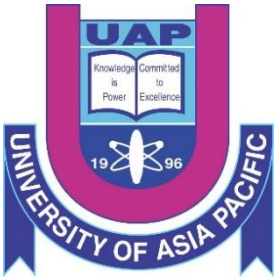

Department of Computer Science and Engineering

Course Code: CSE 430 **Course Title: Compiler Design Lab**

Full Marks: 40 **Spring 2023**

Compiler Design Lab Project

This is an individual project.

Aim of the Project

In our regular lab sessions, we have seen various phases of a compiler. From lexical analysis to intermediate code generation, we have seen how the source code is tokenized and how the intermediate code is generated. We have also seen how the symbol table interacts with the phases. Now your task is to create a ***mini compiler*** which will take a small amount of code as input and it will generate the machine code after compilation. So, basically you need to merge all the phases which you have learned till now and create the compiler.

You can create your own simple programming language as input with similar syntax like C/ C++, Java or Python. The language should consist of some basic features such as variables, expressions, control structures (if-else, loops), and functions. For the code generation part, you can convert your intermediate code into simple assembly code as you please.

Note: Your compiler does not need to be a complex one, it is up to you if you want a single pass or multi-pass compiler. Also, you must define proper grammar rules for the operations. It is better if you can show the output of each of the phases. You can incorporate simple error detection mechanism to improve your compiler.

Guidelines

- This is an individual project.
- You will have to submit a project report.
- Your project report should consist of 3 parts. Design, Implementation and Results.
- The Design part should show the complete flow graph or a block diagram of the project. It should show how the input is getting compiled phase-by-phase.
- The Implementation part is mainly the coding part. Your implementation part should contain all the source code.
- Lastly, the result part should contain the output. You can paste the output or attach screenshots of the output also.
- The project report should contain a title page, a content page, the design and implementation part, results, and reference (if-any).
- Use standard page format and font size.
- Create a PDF file and upload the project in google classroom.
- You need to show the project demonstration within the given time.
- You will have to submit the raw code file(s) also.
- **Plagiarism is strictly prohibited.**