Principles Of Compiler Design

- The exercise and project grades are broken into discrete sections, which can be seen in following table

section	Performance
Exercise 1	0.7
Exercise 2	0.7
Exercise 3	0.7
Project: Phase 1	1.25
Project: Phase 2	1.5
Project: Phase 3	1.25
Total	6

Information

Teacher Assistant: Soheil Moharramkhani, Sharzad Aminshirazi
Office: Social Network Lab (Supervisor: dr.Musavi)
E-mail: *S.MKH@AUT.AC.IR*

Exercise

Exercise1: lexical analysis, theory of languages and automata
Exercise2: syntax analysis, parser
Exercise3: semantic analysis, intermediate code, Compiler concepts

Project

Phase1: lexical analysis
Phase2: syntax analysis
Phase3: generating intermediate code

Project demands

1- 5 valid input from given grammar
2- Design and implement lexical analysis by Lex tools for given grammar
3- Debug, edit and fixing ambiguity of grammar (report requires)
4- Design and implement syntax analysis by Yacc tools for given grammar
5- Add symbol table
6- Design and implement intermediate code generator (three address code) for arithmetic expression, for given grammar

Useful link for project: http://www.professionalcipher.com/2017/07/intermediate-code-generation-using-lex-and-yacc-using-switch-case-and-control-flow.html