	Daffodil International University **Department of Computer Science and Engineering (CSE)** **Course Outline**	

Course Code:	CSE 332		
Course Title:	Compiler Design Lab		
Program:	B.Sc. in CSE		
Faculty:	Faculty of Science and Information Technology (FSIT)		
Semester:	Summer	**Year:**	2020
Credit:	1.00	**Contact Hour:**	3.00 Hrs. / Week
Course Level:	L3T3	**Prerequisite:**	CSE 123, CSE135, CSE 232
Course Category:	Core Engineering.		

1.1.Course Objective

This laboratory course is intended to make the students experiment on the basic techniques of compiler construction and tools that can used to perform syntax-directed translation of a high-level programming language into an executable code. Students will design and implement language processors in C by using tools to automate parts of the implementation process. This will provide deeper insights into the more advanced semantics aspects of programming languages, code generation, machine independent optimizations, dynamic memory allocation, and object orientation.

1.2.Course Outcomes (CO's)

CO1	The ability to research, understand and implement computer programs in the areas related to algorithms
CO2	The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.
CO3	The ability to employ modern computer languages, environments, and platforms in creating innovative career paths, to be an entrepreneur, and a zest for higher studies.

Upon the completion of Compiler Design practical course, the student will be able to:

- **Understand** the working of lex and yacc compiler for debugging of programs.
- **Understand** and define the role of lexical analyzer, use of regular expression and transition diagrams.

- **Understand** and use Context free grammar, and parse tree construction.
- **Learn** & use the new tools and technologies used for designing a compiler.
- **Develop** program for solving parser problems.
- **Learn** how to write programs that execute faster.

1.3.Program Outcomes (PO's)

Program Outcomes	
PO1	**Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2	**Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	**Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
PO4	**Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	**Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6	**The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7	**Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO8	**Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	**Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	**Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11	**Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	**Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

1.4.Course Plan/Lesson Plan

Sl No.	List of Experiments
1	Design a lexical analyzer for given language and the lexical analyzer should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value. Simulate the same in C language.
2	Write a C program to identify whether a given line is a comment or not.
3	Write a C program to recognize strings under 'a', 'a*b+', 'abb'.
4	Write a C program to develop a lexical analyzer to recognize a few patterns in C.
5	Write a C program to test whether a given identifier is valid or not.
6	Implementation Of Symbol Table
7	Write a C program for implementing the functionalities of predictive parser for the mini language.
8	Write a C program for constructing of LL (1) parsing.
9	Write a C program to implement Shift-Reduce Parser.
10	Write a C program to construct of DAG (**Directed Acyclic Graph**)
11	Generation of a code for a given intermediate code

1.5.CO-PO Mapping

Exp. No.	Experiment	Program Outcomes Attained	Program Specific Outcomes Attained
1	Design a lexical analyzer for given language and the lexical analyzer should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value. Simulate the same in C language.	PO1, PO2, PO3	CO1
2	Write a C program to identify whether a given line is a comment or not.	PO1	CO1
3	Write a C program to recognize strings under 'a', 'a*b+', 'abb'.	PO1, PO2	CO1, CO2
4	Write a C program to develop a lexical analyzer to recognize a few patterns in C.	PO1, PO2	CO1, CO2
5	Write a C program to test whether a given identifier is valid or not.	PO1	CO1
6	Implementation Of Symbol Table	PO1, PO2, PO3	CO1, CO2
7	Write a C program for implementing the functionalities of predictive parser for the mini language.	PO1, PO2	CO1
8	Write a C program for constructing of LL (1) parsing.	PO1, PO2, PO4, PO5	CO1
9	Write a C program to implement Shift-Reduce Parser.	PO1, PO2, PO3, PO4	CO1, CO2
10	Write a C program to construct of DAG (**Directed Acyclic Graph**)	PO1, PO2, PO4	CO1
11	Generation of a code for a given intermediate code	PO1, PO2	CO1, CO2

1.6.Assessment Methods

Lab Course
❖ Class Attendance: 10%
❖ Lab Submissions: 40%
❖ Lab Performance: 25%
❖ Report Submission: 25%

1.7. Grading System

Numerical Grade	Letter Grade	Grade Point
80-100	A+	4.00
75-79	A	3.75
70-74	A-	3.50
65-69	B+	3.25
60-64	B	3.00
55-59	B-	2.75
50-54	C+	2.50
45-49	C	2.25
40-44	D	2.00
Less than 40	F	0.00

1.8.Additional Support for Students

- Student Portal: http://studentportal.diu.edu.bd/
- Academic Guidelines https://daffodilvarsity.edu.bd/article/academic-guidelines
- Rules and Regulations of DIU https://daffodilvarsity.edu.bd/article/rules-and-regulation
- Career Development Center: https://cdc.daffodilvarsity.edu.bd/
- For general queries: http://daffodilvarsity.edu.bd/