هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission

T-104
2022

Course Specification

Course Title:	**Theory of Compilers**
Course Code:	**433 COMP-3**
Program:	**Bachelor in Computer Science**
Department:	**Computer Science**
College:	**College of Computer Science and Information Technology**
Institution:	**Jazan University**
Version:	**V2**
Last Revision Date:	**12 September 2021**

Table of Contents:

Content	Page
A. General Information about the course	3
1. Teaching mode (mark all that apply) 2. Contact Hours (based on the academic semester)	4
B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods	4
C. Course Content	5
D. Student Assessment Activities	7
E. Learning Resources and Facilities	7
1. References and Learning Resources	7
2. Required Facilities and Equipment	7
F. Assessment of Course Qualit	8
G. Specification Approval Data	8

A. General information about the course:

Course Identification					
1. Credit hours:	3				
2. Course type					
a.	University ☐	College ☐	Department☒	Track☐	Others☐
b.	Required ☒	Elective☐			
3. Level/year at which this course is offered:	Level 13/Year 5				
4. Course general Description					
This course presents an introduction to compilers phases –analysis, synthesis, lexical analysis, regular expressions, finite automata NFA and DFA. Syntax Analysis, context-free grammar, ambiguity, top-down parsing - recursive descent and LL(K), bottom-up parsing, shift reduce parsing, introduction to LR parser, semantic analysis, syntax directed translation, intermediate code generator-three address code, storage organization, heap management, code generation, and code optimization.					
5. Pre-requirements for this course (if any): None					
6. Co- requirements for this course (if any): None					
7. Course Main Objective(s)					
The main purpose of the course is to provide the foundation for understanding the theory and practice of compilers. Learn programming language translation and compiler design concepts; language recognition, symbol table management, semantic analysis and code generation. • Gain an understanding of how compilers translate source code to machine executable. • Provide the students with a clear overview of how to construct grammar for given programming language. • Convert regular expression to its equivalent finite state machine to accept a specified language • Provide practical programming skills necessary for constructing a compiler. • Understand how compilers manage memory during runtime. • Familiarize students with syntax-directed translation techniques..					

1. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1.	Traditional classroom	44	80%
2.	E-learning		
3.	Hybrid • Traditional classroom • E-learning		
4.	Distance learning (Self Learning)	11	20%

2. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	Lectures	22
2.	Laboratory/Studio	22
3.	Field	
4.	Tutorial	
5.	Others (specify)	8
	Total	52

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			
1.1	**Recognize** and define the significance of the several phases through which a typical program is compiled.	K1	Class lectures and lecture notes	Midterm/ Assignment 1 / Final Exam/Final Lab
1.2	**Describe** the purpose of translating to intermediate code based on given code patterns with possible code optimization techniques.	K1	Class lectures/ lecture notes/ Case studies	Lab/ Assignment 2/ Final Exam
1.3	**Discuss** current trends of a parser in a compiler and relate the yield of a parse tree to a grammar derivation.	K2	Class lectures and lecture notes	Midterm/ Assignment 1 / Final Exam/Final Lab
2.0	Skills			

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
2.1	**Analyze** regular expressions to construct finite automata for accepting or generating a certain language.	S1	Class lectures/ lecture notes/Case studies	Final Exam/ Group Assignments
2.2	**Apply** an algorithm for a top-down or a bottom-up parser construction; construct a parser for a small context-free grammar	S4	Class lectures/ lecture notes/ Case studies / Brainstorming	Final Exam/ Assignments 1 /Group Assignments
2.3	**Apply** syntax directed translation techniques in type checking and intermediate code generation	S4	Class lectures/ lecture notes	Final Exam/ Group Assignments / Final Lab
2.4	**Explain** various aspects of the run-time environment into which the high-level code is translated	S2	Class lectures/ lecture notes	Final Exam/ Group Assignments / Final Lab
2.5	**Design** and implement a code generator based on the given optimized codes.	S3	Class lectures/ lecture notes	Final Exam/ Group Assignments / Final Lab
3.0	Values, autonomy, and responsibility			
3.1	**Demonstrate** the ability to work in team to review and design automata theory and compiler design concepts	V2	Small group discussion / Brainstorming/ Class discussion to train students to think independently	Group Assignments/ Final Exam
3.2				
...				

C. Course Content

No	List of Topics	Contact Hours
1.	**Chapter – 1 INTRODUCTION TO COMPILER** a) Language processor, The structure of compiler, Grouping of phases into passes, compiler construction tools.	

		2T + 2P
2.	**Chapter – 2 LEXICAL ANALYSIS** a) Lexical Analysis: The role of Lexical analysis, Regular expressions, b) Finite Automata, From regular expression to automata-conversion of NFA to DFA, c) Construction of an NFA from regular expression. Self-Study: Lexical error and issues	4T + 4P
3.	**Chapter – 3 SYNTAX ANALYSIS** a) The role of parser, Syntax error handling, b) Context free grammar, Ambiguity. c) Parsing- Top Down parsing, First & Follow, LL(1) Grammars. d) Bottom-up parsing, Shift Reduce parsing, Introduction to LR Parser. Self-Study: LR parser	4T + 4P
4.	**Chapter – 4 SYNTAX DIRECTED TRANSLATION** **a)** Syntax Directed Translation: Syntax Directed Definition. **b)** Evaluation order of SDD's, **c)** Construct of syntax tree, Syntax Directed translation schemes	2T + 2P
5.	**Chapter-5 INTERMEDIATE CODE GENERATION** a) Three Address Code, Three Address Statements. b) Quadruple, Triples, Indirect triple, Control Flow, Self-Study: Back patching	3T + 3P
6.	**Chapter – 6 RUNTIME ENVIRONMENT** a) Storage organization, Stack Allocation of spaces, Heap Management	2T + 2P
7.	**Chapter – 7 CODE GENERATION** a) Issues in the design of a code generator, b) Basic Blocks and flow graph, Optimization of basic blocks- c) The DAG representation of Basic blocks Self-Study: WAP 2.0 security Bluetooth security	3T + 3P

8.	**Chapter – 8 CODE OPTIMIZATION** a) Principle source of optimization, b) Code optimization techniques. c) Loop Optimization technique Self-Study: Loops in Flow Graph, Introduction to Dataflow Analysis	
9.	Lab Exam + Revision	2T + 2P

Total		22T+22P

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	Midterm Exam	6th-7th week	15%
2.	Assignment I	3rd week	10%
3.	Assignment II (Case Study/ Group assignment)	6th-7th week	15%
4.	Lab Exam + Lab Assignment	As per schedule	20%
5.	Final Theory Exam	As per schedule	40%
...			

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.)

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	Introduction to compiler and language design, Douglas Thain,Second Edition, 2020. ISBN 979-8655180260
Supportive References	i). Aho, Lam, Sethi, and Ullman, "Compilers: Principles, Techniques, & Tools", Second Edition, Pearson 2013. ISBN 13: 9789332518667. ii).
Electronic Materials	• http://nptel.ac.in/courses.php?branch=Comp • https://www.coursera.org/ • http://java.sun.com/docs/books/tutorial/ • http://ssw.jku.at/Misc/CC/
Other Learning Materials	Online tutorial

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	• Classroom equipped with projector, whiteboard, and sufficient seating arrangements. • Lab with software installed and individual computer terminal for each student.
Technology equipment (projector, smart board, software)	• Whiteboards and projectors for classroom and labs • Computer Lab equipped with 30 PCs having J2ME platform in Net beans 7.0 • An active internet connection.
Other equipment (depending on the nature of the specialty)	None

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching	Students	Indirect (Course evaluation survey form)
Effectiveness of students assessment	CRC / QAU / HoD	Direct (Course reports / result analysis)
Quality of learning resources	Track leaders / CRC	Indirect (Review, meetings and star rating with suggestions for further modification and improvements)
The extent to which CLOs have been achieved	CRC / QAU	Direct (CLO assessment template further verified at course coordinator, Track leader and QAU level)
Other		

Assessor (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

G. Specification Approval Data

COUNCIL /COMMITTEE	DEPARTMENT COUNCIL
REFERENCE NO.	
DATE	15/10/2022