



## Course Specifications

<b>Course Title:</b>	Compilers
<b>Course Code:</b>	CS412
<b>Program:</b>	Computer Science
<b>Department:</b>	Computer Science
<b>College:</b>	Computer and Information Sciences
<b>Institution:</b>	Majmaah University



## Table of Contents

<b>A. Course Identification</b> .....	<b>3</b>
6. Mode of Instruction (mark all that apply).....	3
<b>B. Course Objectives and Learning Outcomes</b> .....	<b>3</b>
1. Course Description .....	3
2. Course Main Objective .....	3
3. Course Learning Outcomes.....	4
<b>C. Course Content</b> .....	<b>4</b>
<b>D. Teaching and Assessment</b> .....	<b>4</b>
1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods .....	4
2. Assessment Tasks for Students .....	5
<b>E. Student Academic Counseling and Support</b> .....	<b>5</b>
<b>F. Learning Resources and Facilities</b> .....	<b>5</b>
1. Learning Resources .....	5
2. Facilities Required.....	6
<b>G. Course Quality Evaluation</b> .....	<b>6</b>
<b>H. Specification Approval Data</b> .....	<b>6</b>



## A. Course Identification

<b>1. Credit hours:</b> 3 (3,1,0)
<b>2. Course type</b>
a. University <input type="checkbox"/> College <input checked="" type="checkbox"/> Department <input type="checkbox"/> Others <input type="checkbox"/>
b. Required <input checked="" type="checkbox"/> Elective <input type="checkbox"/>
<b>3. Level/year at which this course is offered:</b> Level 10
<b>4. Pre-requisites for this course (if any):</b> CS270 – Programming Languages
<b>5. Co-requisites for this course (if any):</b> NIL

## 6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	44	100%
2	Blended		
3	E-learning		
4	Distance learning		
5	Other		

## 7. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	Lecture	33
2	Laboratory/Studio	11
3	Tutorial	
4	Others (specify)	
	<b>Total</b>	44

## B. Course Objectives and Learning Outcomes

### 1. Course Description

In this course students will develop a deeper understanding of modern compiler techniques applied to general purpose programming languages. It will give students a working knowledge of the foundations, tools, and engineering approaches used in developing formal language translators.

### 2. Course Main Objective

The student is expected to be able to:

- Understand the principles of compilers construction
- Understand the basic components of a compiler (e.g. lexical analysis, top-down, bottom-up parsing, context-sensitive analysis, and intermediate code generation)
- Design and implement a simple compiler
- Use automatic tools in the development of compilers (e.g. Lex and Yacc).



### 3. Course Learning Outcomes

CLOs		Aligned PLOs
<b>1</b>	<b>Knowledge and Understanding</b>	
1.1	Students will learn, understand and explain the main techniques and algorithms used in compilers.	K1
1.2		
1.3		
1...		
<b>2</b>	<b>Skills :</b>	
2.1	Students learn how to use tools to generate lexical analyzers, parsers, translators and code generators.	S1
2.2		
2.3		
2...		
<b>3</b>	<b>Values:</b>	
3.1	Students use open source and current software tools to build a micro compiler.	C1
3.2		
3.3		
3...		

### C. Course Content

No	List of Topics	Contact Hours
1	Introduction to Compilers: language translation, comparison of interpreters and compilers, language translation phases	3
2	Lexical Analysis: regular expressions role in lexical scanners, comparison of hand-made scanner and automatically generated scanners, formal definition of tokens, use of finite state automata.	6
3	Syntax Analysis: formal definition of grammars, BNF, bottom-up vs. top-down parsing, tabular vs. recursive-descent parsers, error handling	6
4	Parsers Implementation: tabular parsers, symbol tables, the use of tools in support of the translation process	6
5	Semantic Analysis: data types, type-checking models, type-checking algorithms.	3
6	Intermediate Representation, Code Generation: intermediate and object code, intermediate representations, implementation of code generators, tree walking; context sensitive translation	3
7	Code optimization: data-flow analysis; loop optimizations	3
8	Error Detection, Recovery, and Repair, Compiler Implementation	3
9	Review Week	
<b>Total</b>		<b>33</b>

### D. Teaching and Assessment

#### 1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	<b>Knowledge and Understanding</b>		



Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.1	Students will learn, understand and explain the main techniques and algorithms used in compilers.	Lectures, Discussions	Exams, Assignments, Participation
1.2			
...			
2.0	<b>Skills</b>		
2.1	Students learn how to use tools to generate lexical analyzers, parsers, translators and code generators.	Lectures, Labs	Exams, Assignments, Participation
2.2			
...			
3.0	<b>Values</b>		
3.1	Students use open source and current software tools to build a micro compiler.	Lectures, Labs	Lab Project
3.2			
...			

## 2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Attendance and Participation	11	5%
2	Mid-Term Exam	8	20%
3	Mini-project, exercise	11	15%
4	Assignments, Quiz	5, 8	20%
5	Final Exam	11	40%
6			
7			
8			

\*Assessment task (i.e., written test, oral test, oral presentation, group project, essay, etc.)

## E. Student Academic Counseling and Support

**Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice :**

Two office hours per week are dedicated to students.

## F. Learning Resources and Facilities

### 1. Learning Resources

<b>Required Textbooks</b>	A V Aho, R Sethi and J D Ullman, "Compilers: principles, techniques and tools", Pearson Education Limited; Pearson New International Edition; 2nd ed. (2007). ISBN-10: 1292024348, ISBN-13: 978-1292024349
<b>Essential References Materials</b>	



<b>Electronic Materials</b>	<ul style="list-style-type: none"> <li>• <a href="http://www.sdl.edu.sa">http://www.sdl.edu.sa</a></li> <li>• <a href="http://lms.mu.edu.sa">http://lms.mu.edu.sa</a></li> </ul>
<b>Other Learning Materials</b>	

## 2. Facilities Required

Item	Resources
<b>Accommodation</b> (Classrooms, laboratories, demonstration rooms/labs, etc.)	Class Room. Lab.
<b>Technology Resources</b> (AV, data show, Smart Board, software, etc.)	Computer. LEX and YACC Tools.
<b>Other Resources</b> (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	Projector and Smart board.

## G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Extent of achievement of course learning outcomes	Students	Indirect

**Evaluation areas** (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)

**Evaluators** (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify))

**Assessment Methods** (Direct, Indirect)

## H. Specification Approval Data

<b>Council / Committee</b>	Computer Science Department
<b>Reference No.</b>	CS Meeting # 7
<b>Date</b>	14-2-1441 H