

TEMPLATE FOR COURSE SPECIFICATION

HIGHER EDUCATION PERFORMANCE REVIEW: PROGRAMME REVIEW

COURSE SPECIFICATION

This Course Specification provides a concise summary of the main features of the course and the learning outcomes that a typical student might reasonably be expected to achieve and demonstrate if he/she takes full advantage of the learning opportunities that are provided. It should be cross-referenced with the programme specification.

1. Teaching Institution	AL-Nahrain University
2. University Department/Centre	Computer science
3. Course title/code	Language translators I/COMP313a
4. Program (s) to which it contributes	Bsc
5. Modes of Attendance offered	Full time
6. Semester/Year	First semester/third year
7. Number of hours tuition (total)	5 hours (2 Lec.+2 Lab+1 tutorial)
8. Date of production/revision of this specification	15/10/2022
9. Aims of the Course:-	
	The main aim of the course is to introduce the students to the principles of compiler writing. It focuses on lexical analysis, parsing, and simple code generation. The students are expected to write a complete compiler for a very simple high level programming language.
	In this first semester, student will learn lexical analyzer, syntax analyzer and semantic. Symbol table generation and checking errors discussed in details in this semester.

10· Learning Outcomes, Teaching ,Learning and Assessment Method

A- Knowledge and Understanding

- A1. To teach students how the translator (Compiler and Interpreter) works and how to design it.
- A2. understanding how to translate the typical code to the machine code
- A3. learn the basic components of compiler
- A4. understanding the syntax by using the grammar

B. Subject-specific skills

- B1. build a whole project of compiler in java language
- B2. learning how the tokenizer performed.
- B3. using the grammars to build the rules of compiler.

Teaching and Learning Methods

Lectures, sheets to step to step by compiling a simple high level language & tutorials

Assessment methods

Exams , Quizzes, and daily assessment of the practical lab

C. Thinking Skills

- C1.
- C2.
- C3.
- C4.

D. General and Transferable Skills (other skills relevant to employability and personal development)

- D1. build a whole project.
- D2. design and analysis
- D3.

11. Course Structure					
Week	Hours	ILOs	Unit/Module or Topic Title	Teaching Method	Assessment Method
1	5		* Introduction of Programming Language & Translators * Open a text file in java program (read it and delete the comment)	* Lecture * tutorial & Lab	*Quiz & activity *Daily assessment
2	5		* The Structure of a Compiler(analysis). * Tokenize the contents of the text file part 1	* Lecture * tutorial & Lab	*Quiz & activity *Daily assessment
3	5		* The Structure of a Compiler(synthesis). * Tokenize the contents of the text file part 2	* Lecture * tutorial & Lab	*Quiz & activity *Daily assessment
4,5	10		* Symbol Table Management, Error Handling * build the LUT part 1	* Lecture * tutorial & Lab	*Quiz & activity *Daily assessment
6	5		* The Grouping of Phases into Passes *build the LUT part 2	* Lecture * tutorial & Lab	*Quiz & activity *Daily assessment
7	1		EXAM1	-	-
8	5		* The Role of the Lexical Analysis * Build the ST and print his contents	* Lecturer * tutorial & Lab	*Quiz & activity *Daily assessment
9	5		*Finite Automata(NFA-DFA) *Build the Error table	* Lecturer * tutorial & Lab	*Quiz & activity *Daily assessment
10	5		*Conversion of NFA into DFA *Parser (part1)	* Lecturer * tutorial & Lab	*Quiz & activity *Daily assessment
11	5		*Constructing of an NFA from a Regular Expression *Parser (part2)	* Lecturer * tutorial & Lab	*Quiz & activity *Daily assessment
12	5		*Minimizing the number of states of DFA *Parser (part3)	* Lecturer * tutorial & Lab	*Quiz & activity *Daily assessment
13	5		*Parse Representation, Properties of parse tree, Issues in parsing context-free grammars *Parser (part4)	* Lecturer * tutorial & Lab	*Quiz & activity *Daily assessment
14	1		EXAM 2	-	-

12. Infrastructure	
Required reading: · CORE TEXTS · COURSE MATERIALS · OTHER	Compiler (principles, techniques, and tools) second edition Alfred V.Aho Columbia University Monica S.Lam Stanford University Ravi Sethi Avaya Jeffrey D.Ullman Compiler Writing (the theory and practice) Jean-Paul Tremblay Paul G.Sorenson Department of Computational Science/university of Saskatchewan, Saskatoon, Canada
Special requirements (include for example workshops, periodicals, IT software, websites)	Computer labs , java & ASM software
Community-based facilities (include for example, guest Lectures , internship, field studies)	