Course Description Form

Course Description

This course description provides a brief, well-documented description of the most important course characteristics and the learning outcomes expected of the student to achieve.

Whether he made the most of the learning opportunities available. This must be linked to the program description.

.1 Educational institution	**Shatt al-Arab Private University / College of Science**
.2 Scientific Department/ Center	**Department of Computer Science**
.3 Course Name/Code	**translators**
4. Forms of attendance Available	**Mandatory attendance**
.5 Semester/Year	Second semester 2024-2025
.6 Number of study hours (total)	4 hours (2 theoretical + 2 practical)
.7 Date this description was prepared	
8. Course objectives	
The objective of this course is to explain how a translator works, its basic components, and the steps involved in designing a translator. Building a translator is one of the most important objectives of this course.	

9. Course outcomes, teaching, learning and assessment methods

A- Cognitive objectives A-1 Understand the general structure of translators A-2 Understand the basic techniques used in building the translator A-3 Understand the basic data structure used in building the translator A-4 Practical application of the above concepts

B - Course Skill Objectives B1 - Developing the student's ability to understand the general structure of the translator B2 - Developing the student's ability to understand the basic techniques used in building the translator B3 - Developing the student's ability to understand the basic data structure used in building the translator B4 - Developing the student's ability to design a complete translator

Teaching and Learning Methods

-1 Theoretical lectures -2 Practical applications of the theoretical material -3 Assignments -4 Projects

Evaluation methods

-1 Monthly exams

-2 Instant exams

-3 Practical exams

-4 Scientific reports

Emotional and value-based objectives C-1 Benefiting from daily experiences and human behaviors in solving problems and transferring them to the computer C-2 Developing the student's existing skills and employing them in solving problems C-3 Instilling a spirit of creativity in the student

Teaching and learning methods

-1 Theoretical lectures reinforced with illustrative examples that foster a spirit of interaction and discussion among students. -2 Laboratory experiments that reinforce the theoretical material.
Evaluation methods
-1 Continuous evaluation and follow-up of the student -2 Group assessment of students -3 Mutual assessment of students and the subject teacher
D- General and transferable skills (other skills related to employability and personal development). D-1 Skill in using Calculator D-2 Design skill D-3 Teamwork skill D-4 Skill in developing algorithmic methods for solving problems

10. Course structure

Evaluation method	Teaching method	Unit name/topic	Required learning outcomes	watches	The week
Theoretical and practical tests	Theoretical lectures	• **Introduction** to Compilers: • The role of language translation in the programming process; • Comparison of interpreters and compilers,	General Introduction About the translators	4	1
	Theoretical lectures and practical applications	• language translation phases, • machine dependent and machine independent aspects of translation, • language translation as a software engineering activity	framework Learn the translator and how it works	4	2
	Theoretical lectures and practical applications	• **LexicalAnalysis:** Application of regular expressions in lexical scanners, • Lexical analysis: hand coded scanner vs. automatically generated scanners Lexical Analysis: formal definition of tokens,	Learn to install And the work of a linguistic vocabulary analyst to design it How	12	5-3

		implementation of final state automata.			
	Theoretical lectures and practical applications	• **Syntax Analysis:** • Revision of formal definition of grammars, • BNF and EBNF; • bottom-up vs. top-down parsing,	Learn how to install and operate grammar analyzer Linguistics and how His design	8	7-6
	Theoretical lectures and practical applications	• Syntax Analysis: • tabular vs. recursive-descent parsers, error handling,	Learn how to install and operate grammar analyzer Linguistics and how His design	8	9-8
	Theoretical lectures and practical applications	Parsers Implementation: automatic generation of tabular parsers,	Learn to install The work of the language analyzer and how it is designed	8	11-10
	Theoretical lectures and practical applications	• symbol table management, • the use of tools in support of the translation process	Learn to install And make a table Token Management How to design it	8	13-11
	Theoretical lecture and discussion	Project presentation	an offer The project	4	14

11.Infrastructure

Aho, Alfred V. *Compilers: Principles, Techniques and Tools (for Anna University), 2/e.* Pearson Education India, 2007.	1- Required textbooks
	2- Main references (sources)

W. Appel, Modern Compiler Implementation in Java, Prentice Hall, 2002	
	A- Books and references recommended in the reports (scientific journals, etc.)
	B - Electronic references, websites The Internet....