

Course specification

(4203 Compiler Theory)

Faculty: HICIT- Higher Institute for Computers & Information Technology

Programme(s) on which the course is given: Under graduate program in Computer Science

Major or minor element of programme: Elective

Department offering the programme: Department of Computer Science

Department offering the course: Department of Computer Science

Year / Class: 4th Year – 2nd Semester

Date of specification approval: 22/2/2016

A- Basic Information

Title: Compiler Theory

Code: 4203

Weekly Hours:

Lecture : 3

Exercise: -

Practical: 3

Total: 6

B- Professional Information

1- Course Objectives:

The student will learn principles of compiler construction and operation. Topics will include: lexical analysis, symbol tables, parsing, type checking, optimization, and code generation.

2- Program ILOs Covered by Course

| Program Intended Learning Outcomes | | | |
|------------------------------------|---------------------|-----------------------------------|---------------------------------|
| Knowledge and understanding | Intellectual Skills | Professional and practical skills | General and Transferable skills |
| a10 | b1, b2, b4, b15 | c10 | d12 |

3 - Intended learning outcomes of course (ILOs)

Successful completion of this course will lead to the following learning outcomes:

a- Knowledge and Understanding

Students should:

- a1. Understand the basic phases of Compilation.
- a2. Understand how compilers operate.

b- Intellectual skills

The ability to:

- b1. Analyze and recognize the significance of the several phases through which a typical program is compiled.

c- Professional and practical skills

The ability to:

- c1. Implement typical compilation phases.
- c2. Design a simple compiler.

d- General and transferable skills

The ability to:

d1. Work in a group in order to design and implement a typical high level language compiler.

d2. Present the implemented compiler and make a demo.

4- Contents

| Topic | Hours | Lec. | Exc/Lab |
|---|-------|------|---------|
| Introduction to compiler theory | 6 | 3 | 3 |
| Scanning & Finite Automata | 18 | 9 | 9 |
| Context- Free Grammars & Parsing | 18 | 9 | 9 |
| Semantic Analysis | 6 | 3 | 3 |
| Runtime Environment | 6 | 3 | 3 |
| Code Generation | 12 | 6 | 6 |
| Implementation project in compiler design | 12 | 6 | 6 |

5 -Teaching and learning methods

5.1 Lectures

5.2 Tutorial Exercises

5.3 Practical Lab

5.4 Discussions.

6-Student assessment methods

6.1 Midterm Exam: To assess the knowledge and understanding achieved by the student during the previous weeks.

6.2 Final Exam: To evaluate what the student gain at the end of the course, and to assess: the knowledge and understanding, general skills, and intellectual skills.

6.3 Course Project: To allow students work in team, and to evaluate knowledge, understanding, intellectual, and transferable skills.

6.4 Course Work &Quizzes: To keep the student always in the course, and to evaluate knowledge, understanding, intellectual, and transferable skills.

Assessment Schedule

| Assessment | Week # |
|----------------------|--------|
| Mid Term Exam | 8 |
| Final Exam | 16 |
| Course Project | 3-14 |
| Course Work &Quizzes | 2-14 |

Assessment Weight

| Assessment | Weigh % |
|----------------------|---------|
| Mid Term Exam | 5% |
| Final Exam | 80% |
| Course Project | 10% |
| Course Work &Quizzes | 5% |
| Total | 100 |

Course Work &Quizzes: (Short Exams, Assignments, Researches, Reports, Presentations, Class/Project discussion)

7- List of references

7.1 Text Books

- Compiler Construction: Principles and Practice, Kenneth C. Louden

7.2 Internet Location :

- www.cs.sjsu.edu/~louden/cmptext/
- There are some quite useful online books, notes, and information sources. Searching on Google will lead you to many sources. Among are the following:

| | |
|---|---|
| http://lambda.uta.edu/cse5317/notes.pdf | Design and Construction of Compilers, Leonidas Fegaras, 2004. |
| http://scifac.ru.ac.za/compilers | Compilers and Compiler Generators: an introduction with C++, P. D. Terry, Rhodes University, 1997. |
| http://www.compilers.net/Dir/Free/Books | A set of links to free compiler related books on the net. |
| http://www.thefreecountry.com | Free programmers' resources including links to many free compilers, some of whom's source code is readily available to inspect. |
| http://www.sai.msu.su/sal/F/1 | Scientific applications on Linux, which includes an extensive list of links to compilers. |
| http://www.compilerconnection.com | Links to compiler information. |

8- Required Facilities

7.1 Tools/Software

- .NET framework

9-Course Matrices

9.1-Course Content/ILO Matrix

| Course Contents | a1 | a2 | b1 | c1 | c2 | d1 | d2 |
|----------------------------------|----|----|----|----|----|----|----|
| Introduction to compiler theory | √ | | | | | | |
| Scanning & Finite Automata | | √ | √ | √ | √ | | |
| Context- Free Grammars & Parsing | | √ | √ | √ | √ | | |
| Semantic Analysis | | √ | √ | √ | √ | | |
| Runtime Environment | | | √ | | | | |
| Code Generation | | √ | √ | √ | √ | | |
| Course Project | | | | | | √ | √ |

9.2-Learning Method /ILOs Matrix

| Learning Methods | a1 | a2 | b1 | c1 | c2 | d1 | d2 |
|--------------------|----|----|----|----|----|----|----|
| Lectures | √ | √ | √ | √ | √ | | |
| Tutorial Exercises | | | √ | √ | √ | | |
| Practical Lab | | | √ | √ | √ | | |
| Discussions. | | | √ | √ | √ | √ | √ |

9.3- Assessment Methods /ILOs Matrix

| Assessment Methods | a1 | a2 | b1 | c1 | c2 | d1 | d2 |
|-----------------------|----|----|----|----|----|----|----|
| Mid Term Exam | √ | √ | √ | √ | √ | | |
| Final Exam | √ | √ | √ | √ | √ | | |
| Course Project | √ | √ | √ | √ | √ | √ | √ |
| Course Work & Quizzes | √ | √ | √ | √ | √ | √ | √ |

Course Coordinator: Prof.Dr. Ahmed El-Abbassy ()

Head of Department: Dr. Farouk Shabaan ()

Date: 22/2/2016