Name of Institute: Indus Institute of Technology & Engineering
Name of Faculty: Ms. Madhvi Bera

Course code: CE0702
Course name: Compiler Design
Pre-requisites: System Software and basic concept of translator
Credit points: 3
Offered Semester: 7th

Course Coordinator (weeks 01 - 15)
Full Name: Ms. Madhvi A. Bera
Department with siting location: Department of Computer Engineering (4th Floor Faculty Room, Bhanvar Building)
Email: madhvibera.ce@indusuni.ac.in
Consultation times: 03:00 pm to 04:30 pm

Course Lecturer (weeks 01 - 15)
Full name: Ms. Madhvi A. Bera
Department with siting location: Department of Computer Engineering (4th Floor Faculty Room, Bhanvar Building)
Email: madhvibera.ce@indusuni.ac.in
Consultation times: 03:00 pm to 04:30 pm

Students will be contacted throughout the Session via Mail with important information relating to this Course.

Course Objectives

By participating in and understanding all facets of this Course a student will:

1) To study the design and implementation of assemblers.
2) To study the design and implementation of compilers.
3) To know the design and implementation of linkers and loaders.
4) To have an understanding of macro preprocessors.
5) To have an understanding of system software tools.

Course Outcomes (CO)

After successful completion of the course:

1. Students will have basic understanding of how language-processing system works, how assembler works. Student will be able to explain the concepts and different phases of compilation with compile time error handling.
2. Students will learn how scanner and parser work and can learn how regular and context free languages can be used in compiler. Student will be able to Represent

language tokens using regular expressions, context free grammar and finite automata and design lexical analyzer for a language.

3. Student will be able to Compare top down with bottom up parsers, and develop appropriate parser to produce parse tree representation of the input.
4. Students will learn basics of semantic analysis, code generation and code optimization. Student will be able to Design syntax directed translation schemes for a given context free grammar. Generate intermediate code for statements in high level language.
5. Student will be able to Apply optimization techniques to intermediate code and generate machine code for high level language program.
6. Students will learn basics of macros and Student will be able to Design of linker/loader.

Course Outline

Lexical Analyzer, Parsing, Error-recovery, Intermediate code generation, Code optimization and generation

Method of delivery

1. Chalk & Talk
2. PPT presentation

Study time

Lecture: 3 hours per week
Reading Time :5 Hours

CO-PO Mapping (PO: Program Outcomes)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	1	-	-	-	-	-	-	-	-	3
CO2	-	-	3	2	3	-	-	-	-	-	-	3
CO3	-	3	3	-	-	-	-	-	-	-	-	1
CO4	3	3	-	-	-	-	-	-	-	-	-	3
CO5	3	-	-	2	-	-	-	-	-	-	-	3
CO6	3	-	1	-	-	-	-	-	-	-	-	1

Blooms Taxonomy and Knowledge retention (For reference)

(Blooms taxonomy has been given for reference)

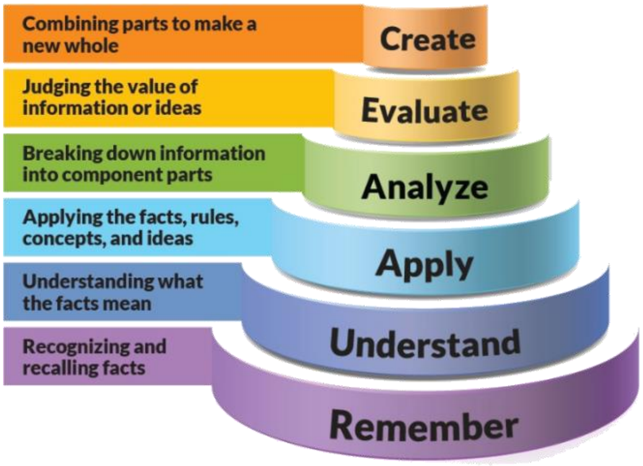

Combining parts to make a new whole
Create
Judging the value of information or ideas
Evaluate
Breaking down information into component parts
Analyze
Applying the facts, rules, concepts, and ideas
Apply
Understanding what the facts mean
Understand
Recognizing and recalling facts
Remember

Figure 1: Blooms Taxonomy

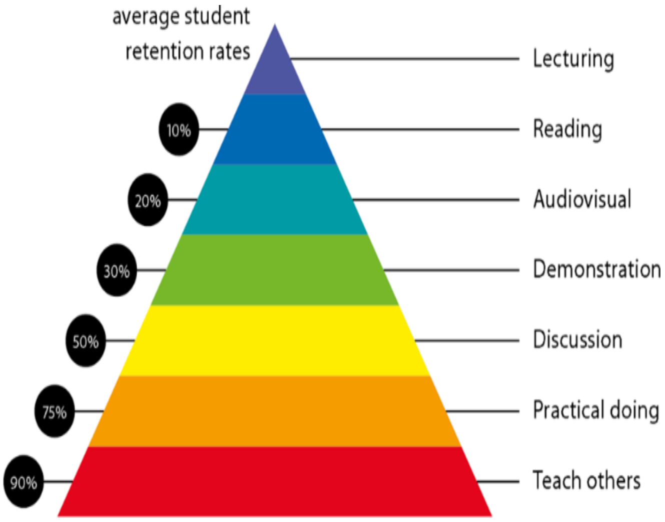


Figure 2: Knowledge retention

Graduate Qualities and Capabilities covered

(Qualities graduates harness crediting this Course)

General Graduate Qualities	Specific Department of ______Graduate Capabilities
Informed Have a sound knowledge of an area of study or profession and understand its current issues, locally and internationally. Know how to apply this knowledge. Understand how an area of study has developed and how it relates to other areas.	**1 Professional knowledge, grounding & awareness**
Independent learners Engage with new ideas and ways of thinking and critically analyze issues. Seek to extend knowledge through ongoing research, enquiry and reflection. Find and evaluate information, using a variety of sources and technologies. Acknowledge the work and ideas of others.	**2 Information literacy, gathering & processing**
Problem solvers Take on challenges and opportunities. Apply creative, logical and critical thinking skills to respond effectively. Make and implement decisions. Be flexible, thorough, innovative and aim for high standards.	**4 Problem solving skills**
Effective communicators Articulate ideas and convey them effectively using a range of media. Work collaboratively and engage with people in different settings. Recognize how culture can shape communication.	**5 Written communication**
	6 Oral communication
	7 Teamwork
Responsible Understand how decisions can affect others and make ethically informed choices. Appreciate and respect diversity. Act with integrity as part of local, national, global and professional communities.	**10 Sustainability, societal & environmental impact**

Lecture/tutorial times

Lecture	Tuesday	12.20 pm – 01.20 pm	Room
Lecture	Wednesday	02.00 pm – 03.00 pm	Room
Lecture	Thursday	11.10 am – 12.10 pm	Room

Attendance Requirements

The University norms states that it is the responsibility of students to attend all lectures, tutorials, seminars and practical work as stipulated in the Course outline. Minimum attendance requirement as per university norms is compulsory for being eligible for mid and end semester examinations.

Details of referencing system to be used in written work

Text books

1. Compilers: Principles, Techniques and Tools ByAho, Lam, Sethi, and Ullman, Pearson Second Edition, 2014, ISBN-13: 978-0321486813

Reference Books:

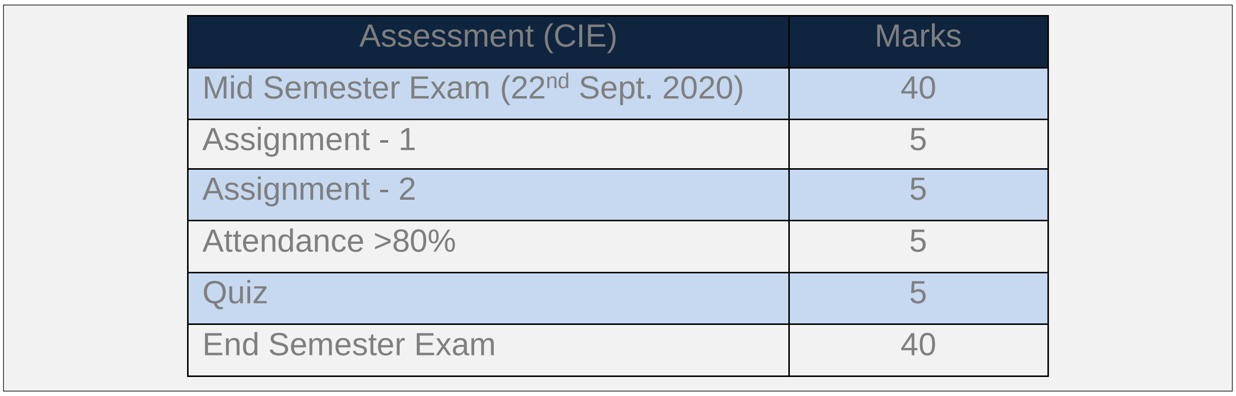
1. Compiler Design in C By Allen I. Holub, Prentice-Hall/Pearson.,1990, ISBN:0-13-1550454
2. Advanced Compiler Design and Implementation, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA,1998, ISBN:1-55860-320-4

Additional Materials

https://youtu.be/yxnbvS2t_QA
- http://nptel.ac.in/courses/106108113
- https://youtu.be/1qOMlqE6LhU
- https://youtu.be/EpAzj7zXrbk

ASSESSMENT GUIDELINES

Your final course mark will be calculated from the following:



Assessment (CIE)	Marks
Mid Semester Exam (22nd Sept. 2020)	40
Assignment - 1	5
Assignment - 2	5
Attendance >80%	5
Quiz	5
End Semester Exam	40

SUPPLEMENTARY ASSESSMENT

Students who receive an overall mark less than 40% in mid semester or end semester will be considered for supplementary assessment in the respective components (i.e mid semester or end semester) of semester concerned. Students must make themselves available during the supplementary examination period to take up the respective components (mid semester or end semester) and need to obtain the required minimum 40% marks to clear the concerned components.

Practical Work Report/Laboratory Report:

A report on the practical work is due the subsequent week after completion of the class by each group.

Late Work

Late assignments will not be accepted without supporting documentation. Late submission of the reports will result in a deduction of -% of the maximum mark per calendar day

Format

All assignments must be presented in a neat, legible format with all information sources correctly referenced. **Assignment material handed in throughout the session that is not neat and legible will not be marked and will be returned to the student.**

Retention of Written Work

Written assessment work will be retained by the Course coordinator/lecturer for two weeks after marking to be collected by the students.

University and Faculty Policies

Students should make themselves aware of the University and/or Faculty Policies regarding plagiarism, special consideration, supplementary examinations and other educational issues and student matters.

Plagiarism - Plagiarism is not acceptable and may result in the imposition of severe penalties. Plagiarism is the use of another person's work, or idea, as if it is his or her own - if you have any doubts at all on what constitutes plagiarism, please consult your Course coordinator or lecturer. Plagiarism will be penalized severely.

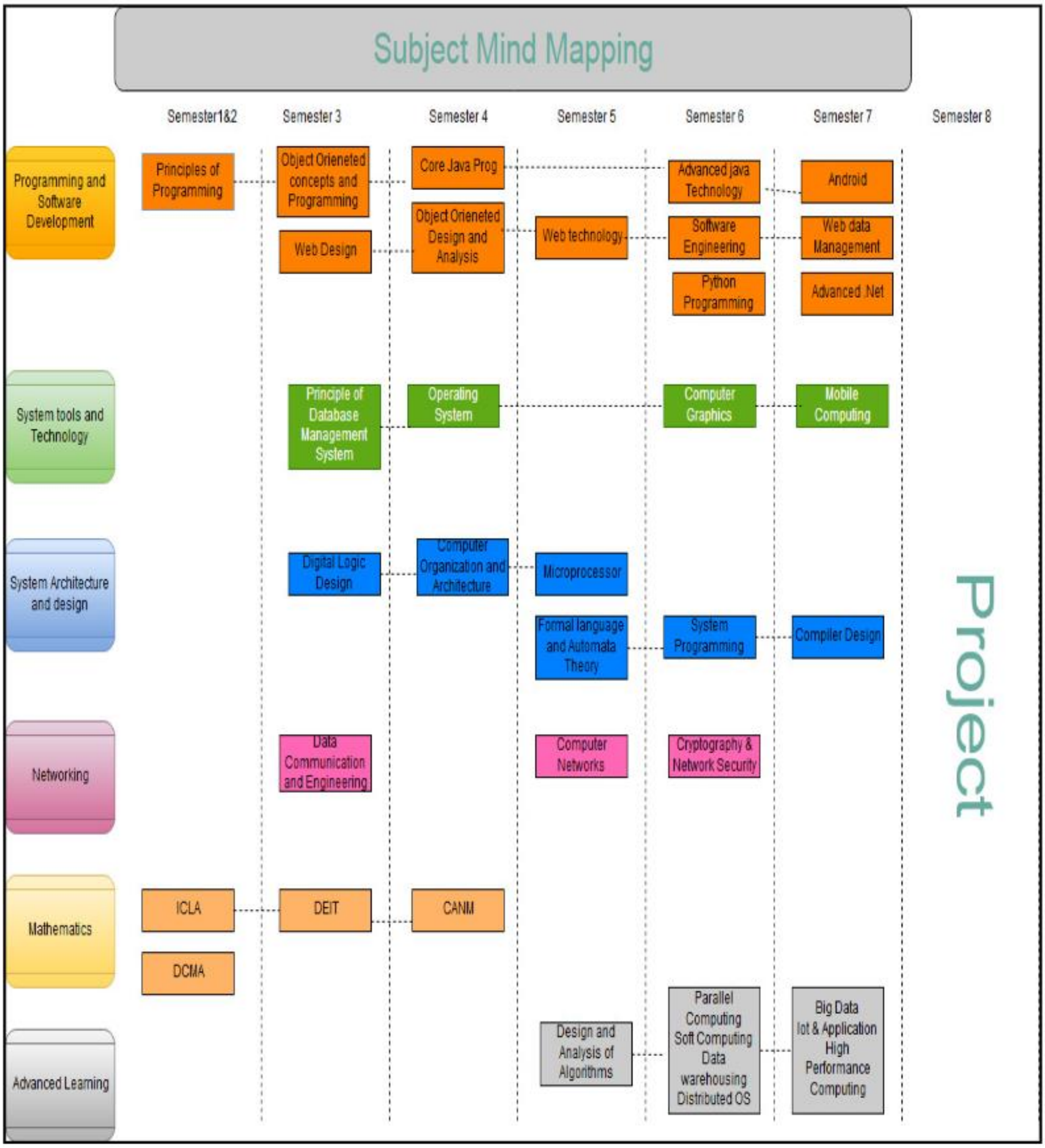
Do not copy the work of other students.
Do not share your work with other students (except where required for a group activity or assessment)

Course schedule (subject to change)

	Week #	Topic & contents	CO Addressed	Teaching Learning Activity (TLA)
	Weeks 1	Overview of the Translation Process, A Simple Compiler, Difference between interpreter, assembler and compiler. Overview and use of linker and loader, types of Compiler, Analysis of the Source Program, The Phases of a Compiler	CO1, CO2	Chalk & Board
	Weeks 2	Cousins of the Compiler, The Grouping of Phases, Lexical Analysis, Hard Coding and Automatic Generation Lexical Analyzers, Front-end and Back-end of compiler, pass structure	CO2	Chalk & Board
	Week 3	Introduction to Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens	*CO2*	Chalk & Board
	Week 4	A Language for Specifying Lexical Analyzers, Finite Automata From a Regular Expression, Design of a Lexical Analyzer Generator, Optimization of DFA	CO2	Chalk & Board
	Week 5	Top Down and Bottom up Parsing Algorithms, Top-Down Parsing, Bottom-Up Parsing, Operator-Precedence Parsing	CO2	Chalk & Board
	Week 6	LR Parsers, Using Ambiguous Grammars, Parser Generators, Automatic Generation of Parsers.	*CO2, CO3*	Assignment - 1
	Week 7	Syntax-Directed Definitions, Construction of Syntax Trees, Bottom-Up Evaluation of S-Attributed Definitions, L-Attributed Definitions, syntax directed definitions and translation schemes	CO3	Chalk & Board
	Week 8	Error Detection & Recovery, Ad-Hoc and Systematic Methods	CO2, CO3	Chalk & Board
	Week 9	Different Intermediate Forms, Syntax Directed Translation Mechanisms And Attributed Mechanisms And Attributed Definition.		Chalk & Board

	Week 10	Source Language Issues, Storage Organization, Storage-Allocation Strategies, and Access to Non local Names, Parameter Passing,		Chalk & Board
	Week 11	Symbol Tables, and Language Facilities for Dynamic Storage Allocation, Dynamic Storage Allocation Techniques.		Chalk & Board
	Week 12	Global Data Flow Analysis, A Few Selected Optimizations like Command Sub Expression Removal, Loop Invariant Code Motion, Strength Reduction etc		Assignment - 2
	Week 13	Issues in the Design of a Code Generator, The Target Machine, Run-Time Storage Management, Basic Blocks and Flow Graphs		Chalk & Board
	Week 14	Next-Use Information, A Simple Code Generator, Register Allocation and Assignment, The DAG Representation of Basic Blocks, Peephole Optimization		Chalk & Board
	Week 15	Generating Code from DAGs, Dynamic Programming Code-Generation Algorithm, Code Generator Generators		Quiz

Program Map for Bachelor of Engineering(CE/CS/IT)



Subject Mind Mapping

	Semester1&2	Semester 3	Semester 4	Semester 5	Semester 6	Semester 7	Semester 8
Programming and Software Development	Principles of Programming	Object Orieneted concepts and Programming Web Design	Core Java Prog Object Orieneted Design and Analysis	Web technology	Advanced java Technology Software Engineering Python Programming	Android Web data Management Advanced .Net	Project
System tools and Technology		Principle of Database Management System	Operating System		Computer Graphics	Mobile Computing	
System Architecture and design		Digital Logic Design	Computer Organization and Architecture	Microprocessor Formal language and Automata Theory	System Programming	Compiler Design	
Networking		Data Communication and Engineering		Computer Networks	Cryptography & Network Security		
Mathematics	ICLA DCMA	DEIT	CANM				
Advanced Learning				Design and Analysis of Algorithms	Parallel Computing Soft Computing Data warehousing Distributed OS	Big Data Iot & Application High Performance Computing	