

Semester-V						
S. No.	Code	Course Name	L	T	P	C
1	ITL 501	Compiler Design	3	1	0	4
2	ITL 502	Design and Analysis of Algorithms	3	0	0	3
3	ITL 503	Micro Processor and Interfacing	3	1	0	4
4	XXXXXX	Elective-I	3	0	0	3
5	ITL 506	Professional Communication and Soft Skills	2	0	0	2
6	ITL507	Project-I	0	0	8	4
7	ITL 508	Compiler Design Lab	0	0	2	1
8	ITL 509	Design and Analysis of Algorithms Lab	0	0	2	1
9	IITL 510	Micro Processor and Interfacing Lab	0	0	2	1
10	ITL 511	Summer Internship	0	0	0	0
Total credits			14	02	14	23
Total hours			30			

<b>Course Code</b>	<b>ITL 501</b>
<b>Course Title</b>	<b>Compiler Design</b>
<b>Number of Credits</b>	<b>3-1-0-4</b>
<b>Course Type</b>	<b>GIR</b>

### Course Objectives

- To understand the various phases of compiler and its use.

### Course content

#### Unit I Introduction To Compiler

Phases of Compilation – Lexical Analysis, Regular Grammar and regular expression for common programming language features, pass and Phases of translation, interpretation, bootstrapping, data structures in compilation – LEX lexical analyzer generator.

#### Unit II Parsing Technique

Context free grammars, Top down parsing – Backtracking, LL (1), recursive descent parsing, Predictive parsing, Preprocessing steps required for predictive parsing. Bottom up parsing: Shift Reduce parsing, LR and LALR parsing, Error recovery in parsing , handling ambiguous grammar.

#### Unit III Syntax-Directed Translation

**Semantic analysis** : Intermediate forms of source Programs – abstract syntax tree, polish notation and three address codes, Syntax directed translation, Conversion of popular Programming languages language Constructs into Intermediate code forms, Type checker. **Symbol Tables**: Symbol table format, organization for block structures languages, hashing, tree structures representation of scope information. Block structures and non-block structure storage allocation: static, Runtime stack and heap storage allocation, storage allocation for arrays.

#### Unit IV Symbol Tables

**Code optimization**: Consideration for Optimization, Scope of Optimization, local optimization, loop optimization, frequency reduction, folding, DAG representation.

**Data flow analysis**: Flow graph, data flow equation, global optimization, redundant sub expression elimination, Induction variable elements, Live variable analysis, Copy propagation.

#### Unit V Code Generation

Object code generation: Object code forms, machine dependent code optimization, register allocation and assignment generic code generation algorithms, DAG for register allocation

### Course Outcomes

- Ability to apply the knowledge of lex tool & yacc tool to develop a scanner & parser
- Ability to design and develop software system for backend of the compiler

### Text Books

- Aho, Alfred V., Lam, Monica S., Sethi, Ravi and Ullman, Jeffrey D. "*Compilers Principles, Techniques and Tools*". Pearson Education Limited Boston, 2014.
- Hollub, Allen I. "*Compiler Design in C*". Prentice-Hall Inc. New Jersey, 1990.

### Reference Books

- Louden, Kenneth C. "*Compiler Construction: Principles and Practice*". Course Technology, 1997.

<b>Course Code</b>	<b>ITL 508</b>
<b>Course Title</b>	<b>Compiler Design Lab</b>
<b>Number of Credits</b>	<b>0-0-2-1</b>
<b>Course Type</b>	<b>GIR</b>

#### Lab Objectives

- The lab course provides the complete description about inner working of a compiler.
- The main focus is to understand working of compiler in detailed manner

#### List of Experiments

- Conversion of infix notation to postfix notation.
  - To Recognize declarative statements
  - Program to recognize arithmetic expression
  - Program to Check valid If statements in C program and report errors to users
  - Program to Check for unterminated, multiline comment statements in C program
  - To Create an assembler that will display warning/errors when symbols are used but not defined and vice versa
  - Write a program that will create and display content of Symbol table
  - Implementation for lexical analyzer
  - To write a C program to implement type checking
  - Implement Predictive parser using C.
- Note** The Instructor may add /tune experiments, wherever he/she feels in a justified manner

#### Course Outcomes

To understand different phases of compiler design

<b>Course Code</b>	<b>ITL 502</b>
<b>Course Title</b>	<b>Design and Analysis of Algorithms</b>
<b>Number of Credits</b>	<b>3-0-0-3</b>
<b>Course Type</b>	<b>GIR</b>

### Course Objectives

- To understand the importance of algorithm and its complexity.
- To design and implement various programming paradigms and their complexity.

### Course Content

**Unit-I Algorithm Design paradigms:** motivation, concept of algorithmic efficiency, run time analysis of algorithms, Asymptotic Notations.

Structure of divide-and-conquer algorithms: sets and disjoint sets, Union, Path compression, quick sort, Finding the maximum and minimum, Quick Sort, Merge sort, Heap and heap sort.

**Unit-II Greedy Algorithms:** Optimal storage, Knapsack problem, Job sequencing with deadlines,, Huffman coding, Minimum Spanning trees: Prim’s algorithm & Kruskal’s algorithm, Huffman codes.

**Unit-III Dynamic programming:** Overview, difference between dynamic programming and divide and conquer, Matrix chain multiplication, longest Common sequence, 0/1 knapsack., bellmonford and Floyd-Warshall algorithm.

**Unit-IV Backtracking:** Queen Problem, vertex cover graph coloring, Hamiltonian cycles. Branch and bound and its applications, 0/1 Knapsack problem, Traveling Salesman Problem.

**Unit-V Computational Complexity:** Complexity measures, Polynomial vs non-polynomial time complexity; NP-hard and NP-complete classes, examples.

### Course Outcomes

- Ability to analyze the time and space complexity, given an algorithm.
- Ability to apply the techniques of algorithm in solving real world problems.
- Ability to develop systematically an algorithm for solving a problem.

### Text Books

1. E. Horowitz, S. Sahni and Rajasekaran, “*Fundamentals of Computer Algorithms*”, Universities Press, 2008.
2. Cormen T. H., Leiserson C. E. and Rivest R. L. and Stein Clifford, “*Introduction to Algorithms*”, Prentice Hall of India, Third Edition, 2010.
3. Skiena Steven S., “*The Algorithm Design Manual*”, Springer, 2nd edition, 2008.

### Reference Book

1. A.V. Aho, J.E. Hopcroft and J.D. Ullman, “*The Design and Analysis of Computer Algorithms*”, Addison Wesley, 2009.

<b>Course Code</b>	<b>ITL 509</b>
<b>Course Title</b>	<b>Design and Analysis of Algorithms Lab</b>
<b>Number of Credits</b>	<b>0-0-2-1</b>
<b>Course Type</b>	<b>GIR</b>

**Course Objectives:**

- To learn how to analyze the complexity of algorithms.
- To learn to program brute force, divide and conquer, decrease and conquer, transform and conquer, greedy, and dynamic techniques.

**List of Experiments:**

- Data structures
- Sorting
- Maximum and minimum problem using divide and conquer strategy.
- Binary search.
- Heap Sort algorithm.
- Kruskal's algorithm.
- Prim's algorithm.
- Matrix chain multiplication
- Dijkstra's algorithm.
- Bellman-Ford algorithm.
- Depth-first search (DFS) on an graph.
- Breadth-first search (BFS) on an graph.
- Advanced data structures.
- Illustrating the different paradigms of algorithm design.
- Problems in string manipulation, graph theory, optimization.

**Course Outcome:**

- Ability to solve and analyze general algorithms based on space and time complexity.

<b>Course Code</b>	<b>ITL 503</b>
<b>Course Title</b>	<b>Microprocessors and Interfacing</b>
<b>Number of Credits</b>	<b>3-1-0-4</b>
<b>Course Type</b>	<b>GIR</b>

### Course Objectives

- To understand interfacing of 16-bit microprocessor with memory and peripheral chips involving system design.
- To understand techniques for faster execution of instructions and improve speed of operation and performance of microprocessors.

### Course Content

**Unit-I Introduction to Microprocessor:** History and Evolution, types of microprocessors, Block diagram of 8085, Pin Diagram of 8085, Addressing modes, Types of Instructions.

**Unit-II Assembly Language Programming and Timing Diagram:** Assembly language programming in 8085, Macros, Labels and Directives, Microprocessor timings, Instruction cycle, Machine cycles, T states, Timing diagram for different machine cycles.

**Unit-III Serial I/O and Interrupts:** Serial I/O using SID, SOD. Interrupts in 8085, Issues in implementing interrupts, multiple interrupts and priorities, Daisy chaining, Interrupt handling, Enabling, disabling and masking of interrupts.

**Unit-IV Data transfer techniques:** Programmed data transfer, Parallel data transfer using 8155. Programmable parallel ports and handshake input/output, Programmable interrupt controller 8259A. DMA transfer, cycle stealing and burst mode of DMA, 8257 DMA controller.

**Unit-V Microprocessor Interfacing Techniques Interfacing memory and I/O devices:** Addressing memory, interfacing static RAMs, Interfacing and refreshing dynamic RAMs, interfacing a keyboard, interfacing a printer, Interfacing A/D converters, D/A converters. Architecture of 8086: Pin diagram of 8086, addressing modes, Comparison of 8086 and 8088, minimum mode maximum mode, system timing, introduction to Pentium and further series of microprocessors.

### Course Outcomes

- Identify various addressing modes Perform various microprocessor based programs.
- Interpret & Solve various automation based problems using microprocessor.

### Text Books

1. Gaonkar, Ramesh S, "*Microprocessor architecture, Programming and applications with 8085*", 6<sup>th</sup> Edition, Prentice Hall, 2013.
2. Brey, Barry B., "*The Intel Microprocessor, 8086/8088, 8018/80188, 80286, 80386, 80486, Pentium and Pentium pro-processors – architecture, Programming and interfacing*", 8<sup>th</sup> Edition, Prentice Hall 2012.

### Reference Book

1. Ufferbeck John, "*The 8080/85 Family: Design, Programming & Interfacing*", PHI India,

<b>Course Code</b>	<b>ITL 510</b>
<b>Course Title</b>	<b>Microprocessors and Interfacing Lab</b>
<b>Number of Credits</b>	<b>0-0-2-1</b>
<b>Course Type</b>	<b>GIR</b>

### **Lab Objectives**

- To become familiar with the architecture and Instruction set of Intel 8085 microprocessor.
- To provide practical hands on experience with Assembly Language Programming.
- To familiarize the students with interfacing of various peripheral devices with 8085 microprocessor.

### **List of Experiments**

- Introduction of microprocessor 8085 trainer kit – 85AD
- The addition of two 8-bit numbers.
- The subtraction of two 8-bit numbers.
- The addition with carry of two 8-bit numbers.
- The subtraction with borrow of two 8-bit numbers.
- The addition of two BCD numbers.
- The subtraction of two BCD numbers.
- The multiplication of two 8-bit numbers by repeated addition method.
- The multiplication of two 8-bit numbers by bit Rotation method.
- The division of two 8-bit numbers by repeated addition method.
- The division of two 8-bit numbers by bit rotation method.
- The square of given numbers in array.
- To find largest number in an array.
- Study of 8086 microprocessor kit
- The addition of two 16-bit numbers.

### **Course Outcomes**

- Explain the architecture, pin configuration of various microprocessors.
- Identify various addressing modes.
- Perform various microprocessor-based programs.

<b>Course Code</b>	<b>ITL 505</b>
<b>Course Title</b>	<b>Professional Communication and Soft Skills</b>
<b>Number of Credits</b>	<b>2-0-0-2</b>
<b>Course Type</b>	<b>GIR</b>

### **Course Objective**

- To enhance the holistic development of students and improve their employability skills.

### **Course Content**

**Unit-I Introduction to Soft Skills & Professional ethics:** Aspects of Soft Skills, Effective Communication Skills, Personality Development, Importance of Professional Ethics.

**Unit-II Team Building:** To know the nature of the team, To understand personal as well as professional goals of the members of the group, To work effectively in a team through building relation and interpersonal communication.

**Unit-III Art of Negotiation:** What is negotiation, Ways of negotiating, To understand the power of language and non-verbal communication.

**Unit-IV Organizing Meetings:** How to call the meeting, How to organize a meeting, How to design the agenda and prepare minutes of the meeting.

**Unit-V Presentation Skills:** Reading, structure of presentation, verbs often required, language focus, importance of body language in presentation, preparing an outline of a presentation, ending the presentation.

**Unit-VI Stress Management & Time Management:** Kinds of stress, Identify the right reason/s of stress, How to handle the pressure, Techniques to cope with the stressful situation at a workplace. Goal setting, Understand the importance of time and How to prepare the time line.

**Unit-VII Group Discussion & Public Speaking:** Nature of discussion, Ways to form and present the arguments. To learn the skills of appearing in an interview and being successful in it.

### **Course Outcomes**

- Understand and recognize the importance of interpersonal skills.
- Understand the realistic perspective of work and work expectations.

### **Text books**

1. Rizvi, Ashraf., *“Effective Technical Communication”*, Tata McGraw Hill ,2008.
2. Mohan, Krishan., *“Developing Communication Skills”*, Mac Millan India Limited, 2009.

### **Reference Books**

1. Dale, Carnegie., *“How to win Friends and Influence People”*, New York: Simon & Schuster, 1998.
2. Coleman, Daniel. *“Emotional Intelligence”*. Bantam Book, 2006.



<b>Course Code</b>	<b>ITL 507</b>
<b>Course Title</b>	<b>Project-I</b>
<b>Number of Credits</b>	<b>0-0-8-4</b>
<b>Course Type:</b>	<b>GIR</b>

The project work will involve detailed literature survey, implementation, and experimentation plan. At the end of the 5th semester project work should have been demonstrated and work done will be evaluated.

<b>Course Code</b>	<b>ITL 511</b>
<b>Course Title</b>	<b>Summer Internship</b>
<b>Number of Credits</b>	<b>0-0-0-0</b>
<b>Course Type</b>	<b>Non Credit</b>

The students will carry out summer internship of minimum 4 weeks in Industry\* after 4<sup>th</sup> semester. A Project report based on training and corresponding proofs of training will be submitted by the student at the start of 5<sup>th</sup> semester, In this course, the credits will not be counted for the calculation of the final CGPA. Only the grades will appear in the Grade card and transcript.

### List of program Electives

<b>Course Code</b>	<b>ITL 504</b>
<b>Course Name</b>	<b>Information Security</b>
<b>Number of Credits</b>	<b>3-0-0-3</b>
<b>Type of Course</b>	<b>PE</b>

**Course Objective:**

- To provide understanding of principal concepts, issues and approaches of security.

**Unit -I Overview:** Computer Security Concepts, Security Functional Requirements, Fundamental Security Design Principles, Attack Surfaces and Attack Trees, Computer Security Strategy.

**Unit -II Access Control:** Access Control Principles, Subjects-Objects and Access Rights, Discretionary Access Control, UNIX File Access Control, Role- Based Access Control, Attribute-Based Access Control, Trust Frameworks. Case Study: RBAC System for a Bank.

**Unit -III Database Security:** The need for Database Security, RDBMS and SQL Injection attacks, Database Access Control, Inference, Database Encryption.

**Unit -IV Authentication and Authorization:** Introduction, Authentication Methods, Passwords, Biometrics, Two-Factor Authentication, Single Sign-On and Web Cookies. Steganography, Authorization: A Brief History, Access control Matrix, Multilevel Security Models, Covert Channels, Inference Control, CAPTCHA.

**Unit -V Firewalls and Intrusion Detection and Prevention Systems:** Firewall Characteristics and Access Policy, Types of Firewall, Firewall Biasing, Firewall Location and Configuration, Intrusion Detection Systems, Intrusion Prevention Systems, Unified Threat Management Products.

**Course Outcomes:**

- Acquire a practical overview of the issues involved in the field of information security and assurance.

**Text Books:**

1. Stallings William and Brown Lowrie, “*Computer Security: Principles and Practice*”, Pearson, Fourth Edition, 2018.
2. Stamp Mark, “*Information Security: Principles and Practices*”, Wiley Publication, Second Edition, 2011.

**Reference Book:**

1. Stallings William, “*Cryptography and Network Security: Principles and Practice*”, Pearson, Seventh Edition, 2017.

<b>Course Code</b>	<b>ITL 505</b>
<b>Course Name</b>	<b>Natural Language Processing</b>
<b>Number of Credits</b>	<b>3-0-0-3</b>
<b>Course Type</b>	<b>PE</b>

### **Course Objectives**

- To understand the application of computational methods in linguists.
- To apply statistical and probabilistic methods for parameter estimation and inference.
- To know how the computational methods, give insight into observed human language phenomena.

### **Course Content**

**Unit-I Sound:** Biology of Speech Processing; Place and Manner of Articulation; Word Boundary Detection; Argmax based computations; HMM and Speech Recognition.

**Unit-II Words and Word Forms:** Morphology fundamentals; Morphological Diversity of Indian Languages; Morphology Paradigms; Finite State Machine Based Morphology; Automatic Morphology Learning; Shallow Parsing; Named Entities; Maximum Entropy Models; Random Fields.

**Unit-III Structures:** Theories of Parsing, Parsing Algorithms; Robust and Scalable Parsing on Noisy Text as in Web documents; Hybrid of Rule Based and Probabilistic Parsing; Scope Ambiguity and Attachment Ambiguity resolution.

**Unit-IV Meaning:** Lexical Knowledge Networks, Wordnet Theory; Indian Language Wordnets and Multilingual Dictionaries; Semantic Roles; Word Sense Disambiguation; WSD and Multilinguality; Metaphors; Coreferences.

**Unit-V Web 2.0 Applications:** Sentiment Analysis; Text Entailment; Robust and Scalable Machine Translation; Question Answering in Multilingual Setting; Cross Lingual Information Retrieval (CLIR).

### **Course Outcomes**

- Ability to compare and contrast approaches to natural language processing
- Ability to comprehend and analyze the various elements of speech processing
- Ability to design and develop machine learning techniques in the area of NLP

### **Text Books**

1. Jurafsky, Dan and Martin, James, "*Speech and Language Processing, 2nd Edition*", Prentice Hall, 2013.
2. Manning, Christopher and Heinrich, Schutze, "*Foundations of Statistical Natural Language Processing*", MIT Press, 1999.

### **Reference Books**

1. Allen, James, "*Natural Language Understanding, 2<sup>nd</sup> edition*", Benjamin Cumming, 2002.
2. Charniack, Eugene, "*Statistical Language Learning*", MIT Press, 1996.