Name of Institute: IITE Indus University Ahmedabad

Name of Faculty: Prof. Toral Desai

Course code: **CE0421**

Course name: **Core Java Programming**

Pre-requisites: C, C++

Credit points: 4

Offered Semester: IV

Course Coordinator

Full name: Prof. Toral Desai

Department with sitting location: CSE department, 4th floor Bhanwar Building.

Email: toraldesai.cse@indusuni.ac.in

Consultation times: 3.00 P.M. to 5.00 P.M. (Monday to Friday)

Course lecturers

Full Name: Prof. Madhvi Bera

Department with sitting location: CSE department, 4th floor Bhanwar Building.

Email: madhvibera.ce@indusuni.ac.in

Consultation times: 3.00 P.M. to 5.00 P.M. (Monday to Friday)

Course lecturers

Full Name: Prof. Himali Shah

Department with sitting location: CSE department, 4th floor Bhanwar Building.

Email: himalishah.ce@indusuni.ac.in

Consultation times: 3.00 P.M. to 5.00 P.M. (Monday to Friday)

Students will be contacted throughout the session via mail with important information relating to this course.

Course Objectives:

1. To understand object oriented programming concepts and implement in java.
2. Comprehend building blocks of OOPs language, inheritance, package and interfaces.
3. Identify exception handling methods.
4. Implement multi-threading in object oriented programs.
5. Prepare UML diagrams for software system
6. To enhance the programming skills of students into field of Java Programming and to create their interest in the same field

Course Outcomes (CO)

After successful completion of the course, student will able to:

1. Apply the object oriented concepts for the given problem and able to do work in OOP Concept technology.
2. Use and create packages in a java program and manage project web classes in proper order.
3. Create applet application as per customer requirement and develop skill in desktop application development.
4. Use exceptions, threads, collections, logs of Java for the given problem. So they can able design user friendly application.
5. Use graphical user interface using applet in Java programs and able to work in GUI design requirement in industry.
6. Understand and know about basic knowledge of different java framework and able to select future way of interested framework.

Course Outline

UNIT-I [12 hours]

Basics of Java: Features of Java, Byte Code and Java Virtual Machine, JDK, Data types, Operator, Control Statements – If, else, nested if, if-else ladders, Switch, while, do-while, for, for-each, break, continue.

Array and String: Single and Multidimensional Array, String class, StringBuffer class, Operations on string, Command line argument, Use of Wrapper Class.

Classes, Objects and Methods: Class, Object, Object reference, Constructor, Constructor Overloading, Method Overloading, Recursion, Passing and Returning object form Method, new operator, this and static keyword, finalize() method, Access control, modifiers, Nested class, Inner class, Anonymous inner class, Abstract class.

UNIT-II [12 hours]

Inheritance and Interfaces:

Use of Inheritance, Inheriting Data members and Methods, constructor in inheritance, Multilevel Inheritance – method overriding Handle multilevel constructors – super keyword, Stop Inheritance-Final keywords, Creation and Implementation of an interface, Interface reference, instanceof operator, Interface inheritance, Dynamic method dispatch, Understanding of Java Object Class, Comparison between Abstract Class and interface, Understanding of System.out.println –statement

Package: Use of Package, CLASSPATH, Import statement, Static import, Access control.

Exception Handling: Exception and Error, Use of try, catch, throw, throws and finally, Built in Exception, Custom exception, Throwable Class.

UNIT-III [12 hours]

Networking with java.net:

InetAddress class, Socket class, DatagramSocket class, DatagramPacket class.

IO Programming: Introduction to Stream, Byte Stream, Character stream, Readers and Writers, File Class, File InputStream, File Output Stream, InputStreamReader, OutputStreamWriter, FileReader, FileWriter, Buffered Reader.

Collection Classes: List, AbstractList, ArrayList, LinkedList, Enumeration, Vector, Properties, Introuduction to Java.util package.

UNIT-IV [12 hours]

Multithreaded Programming: Use of Multithread programming, Thread class and Runnable interface, Thread priority, Thread synchronization, Thread communication, Deadlock.

Generics: Generics Fundamentals, Bounded Types, Using wildcard arguments & bounded wildcards, Generic methods, constructors, class hierarchies & Interfaces.

Applets: Applet basics, complete skeleton, initialization & termination, repainting, using status window & passing parameters to applets.

Method of delivery

Chalk and Board, PowerPoint presentation, self-study material.

Study time

3 Hours/week Theory

2 Hours/week Practical

CO-PO Mapping (PO: Program Outcomes)

1 Program Outcomes (PO's)

Engineering Graduates will be able to:

- **PO1. Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

- **PO2. Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

- **PO3. Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

- **PO4. Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis, and interpretation of data, and synthesis of the information to provide valid conclusions.

- **PO5. Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

- **PO6**. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

- **PO7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

- **PO8. Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

- **PO9. Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

- **PO10. Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

- **PO11. Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

- **PO12. Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

2. Programme Specific Outcome

- **PSO1. Basics of Computer System**: Should able to understand the principles and working of computer systems. Students can assess the hardware and software aspects of computer systems.

- **PSO2. Program Design**: Design and develop computer programs in the areas related to algorithms, networking, web design, cloud computing, IoT and data analytics.

- **PSO3.Software Development**: Should able to understand the structure and development methodologies of software systems with the use of a various programming languages and open source platforms.

COURSE OUTCOME (CO) and PROGRAM OUTCOME (PO) Matrix

(1-Low, 2-Medium, 3- High)

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C0 1	2	2	3	-	3	-	-	-	3	-	3	3
C0 2	1	1	2	2	-	-	-	-	-	-	2	-
C0 3	1	2	3	1	2	-	-	-	2	1	2	2
C0 4	-	1	2	2	3	-	-	-	1	-	1	-
C0 5	-	2	2	2	3	-	-	-	-	-	2	2
C0 6	-	2	1	2	2	-	-	-	2	-	2	2
CE0421	**2.0**	**1.66**	**2.16**	**2.0**	**2.16**	**-**	**-**	**-**	**2.0**	**1.0**	**2.0**	**2.25**

COURSE OUTCOME and PROGRAM SPECIFIC OUTCOME Matrix

Core Java Programming(CE0421)

CO	PSO1	PSO2
CO 1	1	3
CO2	-	2
CO 3	2	3
CO 4	2	2
CO 5	3	2
CO 6	2	2
CE0421	**2.0**	**2.33**

Blooms Taxonomy and Knowledge retention (For reference)

(Blooms taxonomy has been given for reference)

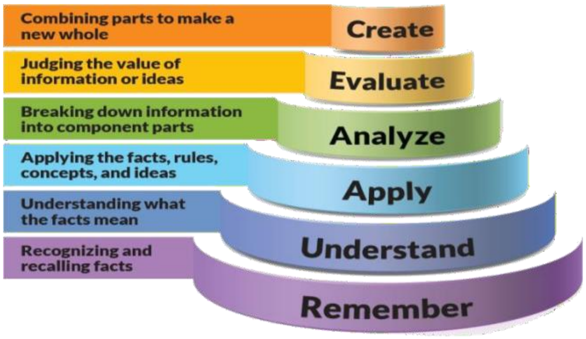

Combining parts to make a new whole
Create
Judging the value of information or ideas
Evaluate
Breaking down information into component parts
Analyze
Applying the facts, rules, concepts, and ideas
Apply
Understanding what the facts mean
Understand
Recognizing and recalling facts
Remember

Figure 1: Blooms Taxonomy

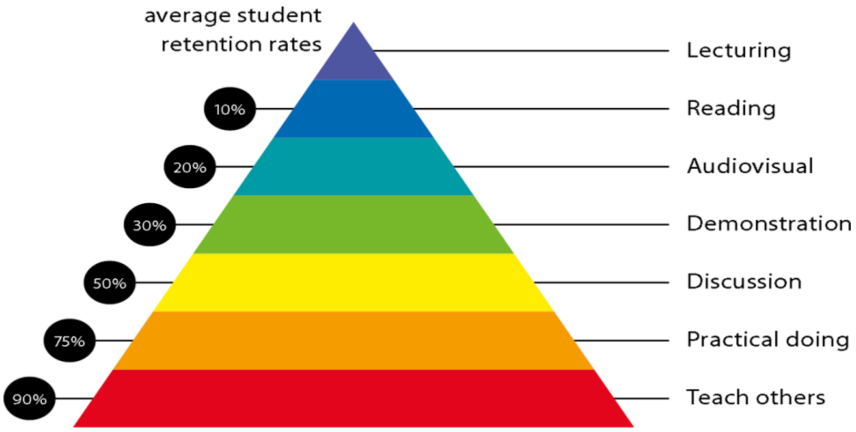


Figure 2: Knowledge retention

Practical work:

Week No.	Class Activity	List of Practical
01	Lab 1	1. Write a program to display “Welcome to Java World”. 2. Write a program to find whether the number is prime or not. 3. Write a program to find a greater number among given three numbers using a) ternary operator b) nested if. 4. Write a program to print the Fibonacci series. 5. Write a program to Print following pattern: * * * * * * * * * * * *
02	Lab 2	1. Write a program to find the average of n numbers stored in an Array.
03	Lab 3	1. WAP to replace substring with other substring in the given string. 2. WAP that to sort given strings into alphabetical order. 3. Create a String Buffer with some default string. Append any string to ith position of original string and display the modified string. Also display the reverse of modified string.
04	Lab 4	1.1) WAP that declares a class named Person. It should have instance variables to record name, age and salary. Use new operator to create a Person object. Set and display its instance variables. 1.2) Add a constructor to the Person class developed above. 2. The employee list for a company contains employee code, name, designation and basic pay. The employee is given HRA of 10% of the basic and DA of 45% of the basic pay. The total pay of the employee is calculated as Basic pay+HRA+ DA. Write a class to define the details of the employee. Write a constructor to assign the required initial values. Add a method to calculate HRA, DA and Total pay and print them out. Write another class with a main method. Create objects for three different employees and calculate the HRA, DA and total pay.

05	Lab 5	1. Write a program which defines base class Employee having three data members, namely name[30], emp_numb and gender and two methods namely input_data() and show_data(). Derive a class SalariedEmployee from Employee and adds a new data member, namely salary. It also adds two member methods, namely allowance (if gender is female HRA=0.1 *salary else 0.09* salary. DA= 0.05*salary) and increment (salary= salary+0.1*salary). Display the gross salary in main class. (Tip: Use super to call base class's constructor0). 2. WAP that illustrates method overriding. Class A3 is extended by Class B3. Each of these classes defines a hello(string s) method that outputs the string "A3: Hello From" or "B3: Hello From" respectively. Use the concept Dynamic Method Dispatch and keyword **super**.
06	Lab 6	1. Write an abstract class shape, which defines abstract method area. Derive class circle from shape. It has data member radius and implementation for area function. Derive class Triangle from shape. It has data members height, base and implementation for area function. Derive class Square from shape. It has data member side and implementation for area function. In main class, use dynamic method dispatch in order to call correct version of method. 2. Create an interface Shape2D which declares a getArea() method. Point 3D contains coordinates of a point. The abstract class Shape declares abstract display() method and is extended by Circle class. it implements the Shape2D interface. The Shapes class instantiates this class and exercises its methods.
07	Lab 7	1. Create a package "employee" and define a Class Employee having three data members, name, emp_num, and gender and two methods- input_data and show_data. Inherit class SalariedEmployee from this class and keep it in package "employee". Add new variable salary and methods allowance (if female hra=0.1* salary else 0.09* salary. DA= 0.05*salary) and increment (salary= salary+0.01 * salary). Calculate gross salary in main class defined in the same package.
08	Lab 8	1. WAP using try catch block. User should enter two command line arguments. If only one argument is entered then exception should be caught. In case of two command line arguments, if fist is divided by second and if second command line argument is 0 then catch the appropriate exception. 2. Define an exception called "NoMatchException" that is thrown when a string is not equal to "India". Write a program that uses this exception.
09	Lab 9	1. The program to creates and run the following three threads. The first thread prints the letter 'a' 100 times. The second thread prints the letter 'b' 100 times. The third thread prints the integer 1 to 100. 2. Write the thread program -1using Runnable interface.

10	Lab 10	1. Write a program that takes two files names (source and destination) as command line argument. Copy source file's content to destination file. Use character stream class. Also do same using byte stream and buffer stream. 2.Write a program which generates random integers and stores them in a file named "rand.dat". The program then reads the integers from the file and displays on the screen.
11	Lab 11	Write the program that demonstrate the use of Stack, Vector and ArrayList classes.
12	Lab 12	Write a Network program that client sends the data as radius of circle to server and server received that data and send the resultant area of circle to requested client.
13	Lab 13	1.Write a program to count occurrence of character in a string. 2.Write a program to check whether a string is a Palindrome. 3.Write a Program to check whether the input number is Armstrong or not.
		Practical Beyond syllabus
14	Lab 14	Write a program to create simple calculator using applet.
15	Lab 15	Write program to create student registration form in applet and store data in database.
16	Lab 16	Write program to create to display a clock using applet.

Lecture/Tutorial times:

As Per Individual Class time table

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 semester examinations.

Text Books:

1. Java Fundamentals, A comprehensive introduction by Herbert Schildt, Dale Skrien, McGraw Hill Education, First Edition, 2013, ISBN 13:978125900659

Reference Books:

1) Programming with Java A Primer – E.Balaguruswamy, McGrawhill, 4th Edition, 2009,ISBN- 9780070141698
2) The Complete Reference, Java 2 Herbert Schildt, TMH, 7th Edition, 2007, ISBN: 978-0-07- 163177-8
3) Core Java Volume-I Fundamentals Horstmann & Cornell, - Pearson Education, 8th Edition, 2008, ISBN -9780132354769
4) Object Oriented Modeling and Design with UML Michael Blaha and James Rambaugh – Pearson Publication, 2nd Edition, 2005, ISBN -

9780131968592

5) UML Distilled: A Brief Guide to the Standard Object Modeling Language by Martin Fowle, 3rd Edition, 2004, ISBN -0321193687

Web Resources:

1. OOP, Basic of Java:
   http://www.nptelvideos.com/java/java_video_lectures_tutorials.php
2. Exceptions and Functions:
   http://www.nptelvideos.com/java/java_video_lectures_tutorials.php?pn=1
3. Multithreading:
   http://www.learnerstv.com/Free-Computers-Video-lectures-ltv006-Page1.htm
4. Networking Basics:

   http://nptel.ac.in/courses/106105084/

ASSESSMENT GUIDELINES

Your final course mark will be calculated from the following:

<table>
<tr><th colspan="2">CIE-Theory (60 Marks)</th><th colspan="2">CIE-Practical (60 Marks)</th></tr>
<tr><td>Class Regularity</td><td>05 Marks</td><td>Minor Project</td><td>20 Marks</td></tr>
<tr><td>Assignment + Quiz</td><td>15 Marks</td><td>Viva / Quiz</td><td>20 Marks</td></tr>
<tr><td>Mid Semester Exam</td><td>40 Marks</td><td>Practical File + Performance</td><td>20 Marks</td></tr>
<tr><th colspan="2">ESE-Theory (40 Marks)</th><th colspan="2">ESE-Practical (40 Marks)</th></tr>
<tr><th colspan="2">Total: 100 Marks</th><th colspan="2">Total: 100 Marks</th></tr>
</table>

SUPPLEMENTARY ASSESSMENT

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

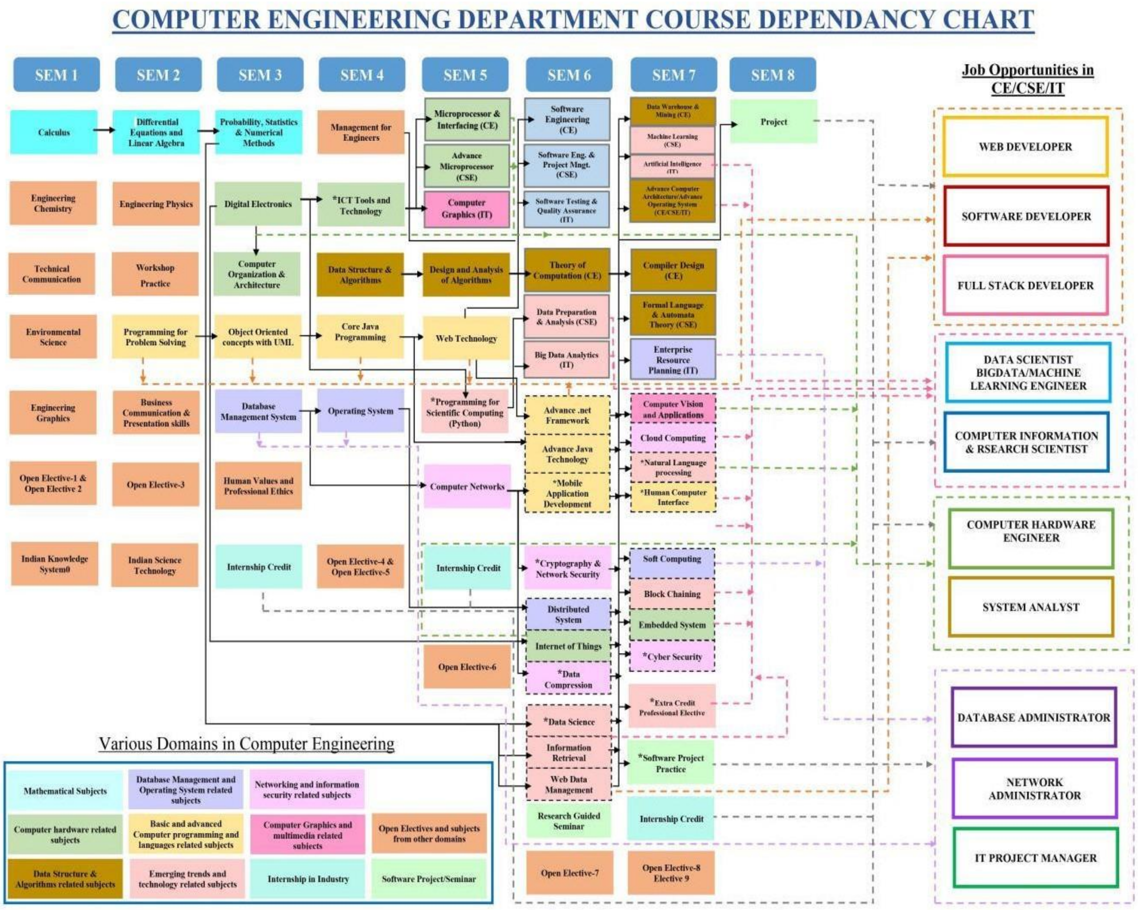
Do not copy the work of other students.

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

Course schedule

Week #	Topic & contents	CO Addressed	Teaching Learning Activity (TLA)
Week 1	**Basics of Java**: Features of Java, Byte Code and Java Virtual Machine, JDK, Data types, Operator, Control Statements – If , else, nested if, if-else ladders, Switch, while, do-while, for, for-each, break, continue.	I	Chalk & Board, Discussion
Week 2	**Array and String**: Single and Multidimensional Array, String class, StringBuffer class, Operations on string, Command line argument, Use of Wrapper Class.	I	Presentation, Chalk& Board
Week 3	**Classes, Objects and Methods**: Class, Object, Object reference, Constructor, Constructor Overloading, Method Overloading, Recursion, Passing and Returning object form Method, new operator, this and static keyword, finalize() method, Access control, modifiers, Nested class, Inner class, Anonymous inner class, Abstract class.	I	Presentation, Chalk & Board
Week 4	**Inheritance and Interfaces**: Use of Inheritance, Inheriting Data membersand Methods, constructor in inheritance, Multilevel Inheritance –method overriding Handle multilevel constructors – super keyword, Stop Inheritance - Final keywords, Creation and Implementation of an interface, Interface reference, instanceof operator, Interface inheritance, Dynamic method dispatch ,Understanding of Java Object Class, Comparison between Abstract Class and interface, Understanding of System.out.println –statement	II	Presentation, Chalk & Board
Week 5	**Package**: Use of Package, CLASSPATH, Import statement, Static import, Access control.	II	Presentation, Chalk & Board
Week 6	**Exception Handling**: Exception and Error, Use of try, catch, throw, throwsand finally, Built in Exception, Custom exception, Throwable Class.	IV	Model presentation
Week 7	**Networking with java.net**: InetAddress class, Socket class, DatagramSocket class, DatagramPacket class.	III	Presentation, Chalk & Board, Demonstration

Week 8	**IO Programming**: Introduction to Stream, Byte Stream, Character stream, Readers and Writers, File Class, File InputStream, File Output Stream, InputStreamReader,OutputStreamWriter, FileReader, FileWriter, Buffered Reader.	III,IV	Presentation, Chalk & Board, Demonstration
Week 9	**Collection Classes**: List, AbstractList, ArrayList, LinkedList, Enumeration, Vector, Properties, Introuduction to Java.util package.	II	Presentation, Chalk & Board
Week 10	**Multithreaded Programming**: Use of Multithread programming, Thread class and Runnable interface, Thread priority, Thread synchronization, Thread communication, Deadlock.	IV	Presentation, Chalk & Board
Week 11	**Generics**: Generics Fundamentals, Bounded Types, Using wildcard arguments & bounded wirldcards, Generic methods, constructors, class hierarchies & Interfaces.	IV	Presentation, Chalk & Board
Week 12	**Applets**: Applet basics, complete skeleton, initialization & termination, repainting, Using status window & passing parameters to applets.	V,VI	Presentation, Chalk & Board


COMPUTER ENGINEERING DEPARTMENT COURSE DEPENDANCY CHART
SEM 1
SEM 2
SEM 3
SEM 4
SEM 5
SEM 6
SEM 7
SEM 8
Calculus
Differential Equations and Linear Algebra
Probability, Statistics & Numerical Methods
Management for Engineers
Microprocessor & Interfacing (CE)
Advance Microprocessor (CSE)
Computer Graphics (IT)
Software Engineering (CE)
Software Eng. & Project Mngt. (CSE)
Software Testing & Quality Assurance (IT)
Data Warehouse & Mining (CE)
Machine Learning (CSE)
Artificial Intelligence (IT)
Advance Computer Architecture/Advance Operating System (CE/CSE/IT)
Project
Engineering Chemistry
Engineering Physics
Digital Electronics
*ICT Tools and Technology
Technical Communication
Workshop Practice
Computer Organization & Architecture
Data Structure & Algorithms
Design and Analysis of Algorithms
Theory of Computation (CE)
Compiler Design (CE)
Formal Language & Automata Theory (CSE)
Environmental Science
Programming for Problem Solving
Object Oriented concepts with UML
Core Java Programming
Web Technology
Data Preparation & Analysis (CSE)
Big Data Analytics (IT)
Enterprise Resource Planning (IT)
Engineering Graphics
Business Communication & Presentation skills
Database Management System
Operating System
*Programming for Scientific Computing (Python)
Advance .net Framework
Advance Java Technology
*Mobile Application Development
Computer Vision and Applications
Cloud Computing
*Natural Language processing
*Human Computer Interface
Open Elective-1 & Open Elective 2
Open Elective-3
Human Values and Professional Ethics
Computer Networks
Indian Knowledge System0
Indian Science Technology
Internship Credit
Open Elective-4 & Open Elective-5
Internship Credit
*Cryptography & Network Security
Soft Computing
Block Chaining
Distributed System
Embedded System
Internet of Things
*Cyber Security
Open Elective-6
*Data Compression
*Extra Credit Professional Elective
*Data Science
Information Retrieval
Web Data Management
*Software Project Practice
Research Guided Seminar
Internship Credit
Open Elective-7
Open Elective-8 Elective 9
Job Opportunities in CE/CSE/IT
WEB DEVELOPER
SOFTWARE DEVELOPER
FULL STACK DEVELOPER
DATA SCIENTIST BIGDATA/MACHINE LEARNING ENGINEER
COMPUTER INFORMATION & RSEARCH SCIENTIST
COMPUTER HARDWARE ENGINEER
SYSTEM ANALYST
DATABASE ADMINISTRATOR
NETWORK ADMINISTRATOR
IT PROJECT MANAGER
Various Domains in Computer Engineering
Mathematical Subjects
Database Management and Operating System related subjects
Networking and information security related subjects
Computer hardware related subjects
Basic and advanced Computer programming and languages related subjects
Computer Graphics and multimedia related subjects
Open Electives and subjects from other domains
Data Structure & Algorithms related subjects
Emerging trends and technology related subjects
Internship in Industry
Software Project/Seminar