Shri G.S. Institute of Technology & Science, Indore

Department of Computer Engineering

12/06/24

Minutes of Meeting of Board of Studies in Computer Engineering held on 12/06/24

A meeting of members of Board of Studies (BoS) in Computer Engineering was held on Tuesday, 12/06/24 at11:30 am in hybrid online-offline mode. The meeting was available on Google Meet platform for online mode.

Following members attended the meeting:

1.	Dr. Vandan Tewari(Professor & Head, Department of Computer Engineering)	Chairperson
2.	Dr. Aruna Tiwari, Professor, Department of Computer Engineering, IIT, Indore	Member, External Expert
3.	Mr. V. R. Sathe, IT Consultant	Member, External Expert
4.	Dr. D.C.Jinwala, Professor, Department of Computer Engineering, SVNIT, Surat; Adjunct Prof., IIT Jammu	Member, External Expert
5.	Prof. D.A. Mehta, Professor, Department of Computer Engineering	Member
6.	Dr. UrjitaThakar, Professor, Department of Computer Engineering	Member
7.	Mr. Surendra Gupta, Associate Professor, Department of Computer Engineering	Member
8.	Ms. Priyanka Bamne, Assistant Professor, Department of Computer Engineering	Member
9.	Ms. Neha Mehra, Assistant Professor, Department of Computer Engineering	Member

Dr. U.A. Deshpande (VNIT, Nagpur), Sh. J.K. Khatwani (Principle Engineer, The Modern Data Company), Dr. AnuradhaPurohit and Sh. Rajesh Dhakadcould not join the meeting.

Discussions were held on all the agenda items. Following are the deliberations

Item no 1: Review of Vision & Mission statements of the department.

The mission and vision of the Department were presented and no change was suggested.

Item no 2: To prepare new scheme and syllabi of B.Tech.(Computer Sc. & Engg.) as per NEP 2020

- The new scheme of B.Tech. program under NEP was prepared and discussed. The recommended scheme of II Yr. B.Tech. is attached as **Annexure-I**.
- Syllabi for II B.Tech. subjects were prepared and the recommended syllabi are attached as **Annexure-II.**
  For all the theory subjects in new scheme, syllabi are same OR with less than 10-15% change. Therefore common question paper may be set for all the theory subjects under old scheme and new scheme. New syllabi have also been prepared for all practical subjects and are attached as **Annexure-II.**

Item no 3: To discuss and prepare Multiple Exit Policy under NEP2020 for undergraduate courseB.Tech.(Computer Sc. & Engg.).

The exit policy which was discussed and recommended by DPAQIC in its meeting held on 09/01/24, was presented and the final recommendation is attached herewith as **Annexure-III.**

Item no 4: To discuss about running the schemes of both the semester (Groups) in each semester.

The proposal of the department to teach subjects of Semester 'A'(Group 'A') to section 'A' and subjects of Semester 'B' (Group 'B') to section B in first semester and vice-versa in second semester, was discussed.
Members were of the opinion that as long as there are no dependencies in offering of subjects, this can be done. In fact, this is an acceptable model of teaching, and hence can be recommended. However, implementation of the same is a matter under purview of Academic Council/Institute Administration.

Any other Item : Under this item no discussion took place.

The meeting ended with vote of thanks to the chair.

Signatures of members:-

1.	Dr. Vandan Tewari	(Chairperson)
2.	Mr. V.R. Sathe	(External Expert)
3.	Prof. D.C. Jinwala	(External Expert)
4.	Dr. Aruna Tiwari	(External Expert)
5.	Prof. D A Mehta	(Member)
6.	Dr. UrjitaThakar	(Member)
7.	Mr. Surendra Gupta	(Member)
8.	Ms. Priyanka Bamne	(Member)
9.	Ms. Neha Mehra	(Member)

Attended online
Attended online
Attended online

Annexure-I

Applicable for students admitted in academic year 2023-2024 and onwards

B.TECH. II YEAR COMPUTER SCIENCE & ENGINEERING (Under NEP-20)

SEMESTER 'A'

S. No.	Subject Category	Subject Code	Subject Name	Hours per Week			Credits			Max. Marks				
				L	T	P	Th.	Pr.	Total	Theory		Practical		Total
										Th.	CW	SW	Pr.	
1	BSC	MA24003	Mathematics -III	3	1	-	4	-	4	70	30	-	-	100
2	PCC	CO 24057	Object Oriented Programming Systems	3	1	-	4	-	4	70	30	-	-	100
3	PCC	CO 24009	Computer Architecture	3	-	-	3	-	3	70	30	-	-	100
4	ESC	EC	Microprocessors and Microcontrollers	3	-	-	3	-	3	70	30	-	-	100
5	HSMC	HU 24005	Economics for Engineers	3	-	-	3	-	3	70	30	-	-	100
6	LC	CO	Object Oriented Programming Lab	-	-	2	-	1	1	-	-	40	60	100
7	LC	CO	Computer Architecture Lab	-	-	2	-	1	1	-	-	40	60	100
8	ESC	EC	Microprocessors and Microcontrollers Lab	-	-	2	-	1	1	-	-	40	60	100
9	LC	CO	Design thinking Lab	-	-	2	-	1	1	-	-	40	60	100
10	ESC	EC 24498	Electronics Workshop	-	-	2	-	1	1	-	-	40	60	100
			Total	15	2	10	17	5	22	350	150	200	300	1000

SEMESTER 'B'

S. No.	Subject Category	Subject Code	Subject Name	Hours per Week			Credits			Max. Marks				
				L	T	P	Th.	Pr.	Total	Theory		Practical		Total
										Th.	CW	SW	Pr.	
1	BSC	MA 24554	Mathematics - IV	3	1	-	4	-	4	70	30	-	-	100
2	PCC	CO 24553	Discrete Structures	3	-	-	3	-	3	70	30	-	-	100
3	PCC	CO 24507	Data Structures	3	-	-	3	-	3	70	30	-	-	100
4	PCC	CO	Agile Software Methodology	3	-	-	3	-	3	70	30	-	-	100
5	OEC	EC 24509	Digital Communication	3	-	-	3	-	3	70	30	-	-	100
6	LC	CO	Data Structures Lab	-	-	2	-	1	1	-	-	40	60	100
7	LC	CO	Software Design Lab	-	-	2	-	1	1	-	-	40	60	100
8	OEC	EC	Digital Communication Lab	-	-	2	-	1	1	-	-	40	60	100
9	LC	CO	Mobile Application Development Lab	-	2	2	-	3	3	-	-	40	60	100
10	HSBC	HU 24881	Values, Humanities and Professional Ethics	-	2	-	2	-	2	-	100	-	-	100
11	MC	*HU	Constitution of India	-	2	-	-	-	-	-	-	-	-	0
			Total	15	7	8	18	6	24	350	250	160	240	1000

* Audit Course

Internship / Course / Training in industry or organization of minimum 2 weeks to be carried out after sem. "A" or Sem. "B" but before commencement of III Year Sem. "A". Evaluation shall be done in III Year Sem. "A".

B.TECH. III YEAR COMPUTER SCIENCE & ENGINEERING (Under NEP-20)

SEMESTER 'A'

S. No.	Category	Subject Code	Subject Nomenclature	Hours Per Week			No. of Credits			Maximum Marks				
				L	T	P	Th.	Pr.	Total	Th.	CW	SW	Pr.	Total
1	PCC	CO34002	Theory of Computation	3	1	-	4	-	4	70	30	-	-	100
2	PCC	CO34005	Data Base Management Systems	3	1	-	4	-	4	70	30	-	-	100
3	PCC	CO34007	Computer Networks	3	-	-	3	-	3	70	30	-	-	100
4	PCC	CO	Operating Systems	3	-	-	3	-	3	70	30	-	-	100
5	LC	CO	Data Base Management Systems Lab	-	-	2	-	1	1	-	-	40	60	100
6	LC	CO	Computer Networks Lab	-	-	2	-	1	1	-	-	40	60	100
7	LC	CO	Operating Systems Lab	-	-	2	-	1	1	-	-	40	60	100
8	SI	CO34481	Internship Evaluation - I	-	-	-	-	-	4	-	-	100	-	100
9	MC	*HU	Essence of Indian Traditional Knowledge	-	2	-	-	-	-	-	-	-	-	0
			Total	12	4	6	14	3	21	280	120	220	180	800

*Audit Course

SEMESTER 'B'

S. No.	Category	Subject Code	Subject Nomenclature	Hours Per Week			No. of Credits			Maximum Marks				
				L	T	P	Th.	Pr.	Total	Th.	CW	SW	Pr.	Total
1	PCC	CO34553	Machine Learning	3	-	-	3	-	3	70	30	-	-	100
2	PCC	CO34563	Design and Analysis of Algorithms	3	-	-	3	-	3	70	30	-	-	100
3	PCC	CO	Information Security	3	-	-	3	-	3	70	30	-	-	100
4	PEC	CO	Elective-I	3	-	-	3	-	3	70	30	-	-	100
5	LC	CO	Machine Learning Lab	-	-	2	-	1	1	-	-	40	60	100
6	LC	CO	Design and Analysis of Algorithms Lab	-	-	2	-	1	1	-	-	40	60	100
7	PCC	CO	Internet of Things Workshop	-	1	2	-	2	2	-	-	40	60	100
8	PROJ	CO34999	Major Project Planning and Seminar	-	-	8	-	4	4	-	-	40	60	100
			Total	12	1	14	12	8	20	280	120	160	240	800

Internship / Course / training in industry or organization of minimum 2 weeks to be carried out after sem. "A" or Sem. "B" but before commencement of IV Year Sem. "A". Evaluation shall be done in IV Year Sem. "A".

All Elective subjects may be offered in offline mode/ MooCs mode.

2/4

(Under (NEP-20)

B.TECH. IV YEAR COMPUTER SCIENCE & ENGINEERING (wef. 2023-24)

SEMESTER 'A'

S. No.	Category	Subject Code	Subject Nomenclature	Hours Per Week			No. of Credits			Maximum Marks				Total
				L	T	P	L	P	Total	Th.	CW	SW	Pr.	
1	PEC	CO	Elective – II	2	-	-	2	-	2	70	30	-	-	100
2	PEC	CO	Elective – III	2	-	-	2	-	2	70	30	-	-	100
3	PEC	CO	Elective – IV	2	-	-	2	-	2	70	30	-	-	100
4	LC	CO	Elective – III Lab	-	-	2	-	1	1	-	-	40	60	100
5	LC	CO	Product Development & QA Lab	-	-	2	-	1	1	-	-	80	120	200
6	PROJ	CO44498/ CO44998	Major Project Phase-I/ Major Project Phase-II	-	-	12	-	6	6	-	-	80	120	200
7	SI	CO44481/ CO44882	Internship Evaluation - II Internship Evaluation - III	-	-	-	-	-	8	-	-	100	-	100
			Total	6	0	16	6	8	22	210	90	300	300	900

Internship / Course / training in industry or organization of minimum 8 weeks to be carried out after sem. "A" but before commencement of IV Year Sem. "B". Evaluation shall be done in IV Year Sem. "B".

SEMESTER 'B'

S. No.	Category	Subject Code	Subject Nomenclature	Hours Per Week			No. of Credits		Total	Maximum Marks				Total
				L	T	P	L	P		Th.	CW	SW	Pr.	
1	PEC		Elective – V	2	-	-	2	-	2	70	30	-	-	100
2	PEC		Elective – VI	2	-	-	2	-	2	70	30	-	-	100
3	PROJ	CO44998/ CO44498	Major Project Phase – II / Major Project Phase – I	-	-	12	-	6	6	-	-	80	120	200
4	SI	CO44481/ CO44882	Internship Evaluation - III Internship Evaluation - II	-	-	-	-	-	8	-	-	100	-	100
			Total	4	0	12	4	6	18	140	60	180	120	500

All Elective subjects may be offered in offline mode/ MooCs mode.

3/4

List of Electives

Elective-I			Elective-II		
1	CO 34601	Data Science & Engineering	1	CO___	Computational Intelligence
2	CO___	Artificial intelligence	2	CO___	Advanced Data Structures
3	CO___	Software Architecture	3	CO 44242	Cloud Computing
4	CO 34602	Object Oriented Software Engineering	4	CO___	Advanced Mobile Computing
5	CO___	Computer Graphics	5	CO___	Multimedia System
6	CO___	Embedded Systems	6	CO___	Edge Computing
7	CO___	Pervasive computing	7	CO___	Real Time Systems
8	CO___	Web Technologies	8	CO___	Affective computing

Elective-III (with LAB)			Elective-IV		
1	CO 44251	Deep Learning	1	CO___	Reinforcement Learning
2	CO___	Advanced Algorithms	2	CO___	Advanced Databases
3	CO 44252	Big Data	3	CO 44307	Cyber Security and Forensics
4	CO___	Human Computer Interaction	4	CO 44308	Image Processing and Computer Vision
5	CO___	Drone Computing	5	CO___	Virtual Reality
6	CO___	Software Verification	6	CO___	Robotics
7	CO___	Compiler Construction	7	CO___	Software Testing
8	CO___	Quantum computing	8	CO___	Natural Language Processing

Elective-V			Elective-VI		
1	CO___	Bioinformatics Computing	1	CO___	Advanced Operating Systems
2	CO___	High Performance Computing	2	CO___	Network Management & Maintenance
3	CO___	Machine Learning for Security	3	CO 44706	Software Project Management
4	CO 44608	Game Design	4	CO___	Large language model
5	CO___	Digital Signal Processing	5	CO 44707	Block Chain Technology
6	CO___	Security in Resource Constrained Environment	6	CO___	Data Mining
7	CO___	Advanced Graph Theory	7	CO___	User Centric Computing
8	CO___	Soft Computing	8	CO___	UX design

4/4)

Annexure- III

Shri G.S. Institute of Technology & Science, Indore

Department of Computer Engg.

Date: 20/12/23

With reference to circular Dean/1221 dated 16/12/23, in order to prepare scheme for B.Tech. (COMP. Engg.) under NEP-2020,following are some suggestions for defining the multiple exit scheme:

As per Ordinance 4D, Student needs to complete 10 credit bridge courses, out of which 6 credits will be job specific internship where as 4 credits are to be earned with a course defined by BoS. The student can be permitted to complete these courses from any standard MoocS platform and is to be approved by BoS.

S. No.	Exit Year	Award	Remarks
1.	After I B.Tech.	Certificate for: 1. Data Entry operation 2. DTP (Proficiency in packages like (Word/Excel/PowerPoint etc.) 3. Systems Operation (A basic proficiency in operating systems installation, managing system resources/basic database operations etc.) 4. Networking and cabling 5. Network management 6. System Administration (Installations/backups/Recovery etc.) 7. Linux Operations 8. Hardware Maintenance & repair.	Equal to level 5 of NHEQF/NSQF* And unified credit level 4.5
2.	After II B.Tech.	Diploma in : 1. Systems Operation 2. Network management 3. System Administration 4. Data Management	Equal to level 6 of NHEQF/NSQF And unified credit level 5
3.	After III B.Tech.	B.Voc(Vocational Degree in Comp.Engg.)with specialization in: 1. Data Science and Machine Learning 2. Network and Cloud management 3. System Administration	Equal to level 7 of NHEQF/NSQF And unified credit level 5.5

*NHEQF/NSQF: National Higher Education Qualification Framework/ National Skills Qualification Framework

P.S. : gc.gov.in/pdfnews/2142241_NHEQF-Draft.pdf

V.R.S.

Annexure - II

OLD

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. II YEAR (4YDC)
SEMESTER-A
CO 24057: OBJECT ORIENTED PROGRAMMING SYSTEMS

*Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
4	1	2	4	1	70	30	40	60	200

PRE-REQUISITES: **CO10504: Computer Programming**

COURSE OBJECTIVES: The objective of course is to develop programming skills of students, using object oriented programming concepts, learn the concept of class and object using Java and develop real world applications.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain various concepts of object oriented terminology.
2. Define and implement the concepts of data encapsulation, abstraction, inheritance and polymorphism.
3. Design and execute quality programs using exception handling.
4. Solve the real world business problems as per specifications.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to Object Oriented Thinking & Object Oriented Programming: Comparison with Procedural Programming, features of Object oriented paradigm, Merits and demerits of OO methodology; Object model; Elements of OOPS, IO processing.

UNIT 2 Encapsulation and Data Abstraction- Concept of Objects: State, Behavior & Identity of an object; Classes: identifying classes and candidates for Classes Attributes and Services, Access modifiers, Static members of a Class, Instances, Message passing, and Construction and destruction of Objects.

UNIT 3 Relationships – Inheritance: purpose and its types, 'is a' relationship; Association, Aggregation. Concept of interfaces and Abstract classes.

UNIT 4 Polymorphism: Introduction, Method Overriding & Overloading, static and run time Polymorphism.

UNIT 5 Strings, Exceptional handling, Introduction of Multi-threading and Data collections. Case study like: ATM, Library management system.

COURSE ASSESSMENT (Th.):

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

COURSE ASSESSMENT (Pr.):

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, file etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

TEXT BOOKS RECOMMENDED:

1. Timothy Budd, "An Introduction to Object-Oriented Programming", Addison-Wesley Publication, 3rd Edition 2002.
2. Cay S. Horstmann and Gary Cornell, "Core Java: Volume I, Fundamentals", Prentice Hall publication, 2007.

REFERENCE BOOKS:

1. G. Booch, "Object Oriented Analysis & Design", Addison Wesley, 2006
2. James Martin, "Principles of Object Oriented Analysis and Design", Prentice Hall/PTR, 1992.
3. Peter Coad and Edward Yourdon, "Object Oriented Design", Prentice Hall/PTR, 1991.
4. Herbert Schildt, "Java 2: The Complete Reference", McGraw-Hill Osborne Media, 11th Edition, 2018.

NEW

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. II YEAR (4YDC)
SEMESTER-A
CO 24057: OBJECT ORIENTED PROGRAMMING SYSTEMS

*Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	1	-	4	-	70	30	-	-	100

PRE-REQUISITES: **CO10504: Computer Programming**

COURSE OBJECTIVES: The objective of course is to develop programming skills of students, using object oriented programming concepts, learn the concept of class and object using Java and develop real world applications.

COURSE OUTCOMES:
After completing the course student should be able to:

1. **Explain various concepts of object oriented terminology.**
2. **Define and implement the concept of data encapsulation, data abstraction.**
3. **Apply the concept of inheritance and polymorphism to implement various problems.**
4. **Design and execute quality programs using exception handling, multithreading and solve the real world business problems as per specifications.**

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to Object Oriented Thinking & Object Oriented Programming: Comparison with Procedural Programming, features of Object oriented paradigm, Merits and demerits of OO methodology; Object model; Elements of OOPS, IO processing.

UNIT 2 Encapsulation and Data Abstraction- Concept of Objects: State, Behavior & Identity of an object; Classes: identifying classes and candidates for Classes Attributes and Services, Access modifiers, Static members of a Class, Instances, Message passing, and Construction and destruction of Objects.

UNIT 3 Relationships – Inheritance: purpose and its types, 'is a' relationship; Association, Aggregation. Concept of interfaces and Abstract classes.

UNIT 4 Polymorphism: Introduction, Method Overriding & Overloading, static and run time Polymorphism,**String Handling**.

UNIT 5 Exceptional handling, Introduction of Multi-threading and Data collections. Case study like: ATM, Library management system.

COURSE ASSESSMENT (Th.):

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

COURSE ASSESSMENT (Pr.):

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, file etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

TEXT BOOKS RECOMMENDED:

1. Timothy Budd, "An Introduction to Object-Oriented Programming", Addison-Wesley Publication, 3rd Edition 2002.
2. Cay S. Horstmann and Gary Cornell, "Core Java: Volume I, Fundamentals", Prentice Hall publication, 2007.

REFERENCE BOOKS:

1. G. Booch, "Object Oriented Analysis & Design", Addison Wesley, 2006
2. James Martin, "Principles of Object Oriented Analysis and Design", Prentice Hall/PTR, 1992.
3. Peter Coad and Edward Yourdon, "Object Oriented Design", Prentice Hall/PTR, 1991.
4. Herbert Schildt, "Java 2: The Complete Reference", McGraw-Hill Osborne Media, 11th Edition, 2018.

OLD

DEPARTMENT OF COMPUTER ENGINEERING
B.Tech. II YEAR (4YDC)
SEMESTER-A
CO 24009: COMPUTER ARCHITECTURE

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	-	-	100

PRE-REQUISITES: NIL

COURSE OBJECTIVES:

The main objective of this course is to compare various architectures of Computers and their components like memory etc. and to develop the skills of the students to write the assembly language programs for various instructions.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Illustrate architecture of a computer, its components and their interconnection.
2. Describe execution of instruction in a computer.
3. Identify the addressing modes used in macro instruction.
4. Design programs in assembly language and justify the importance of parallel architecture.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction, Milestones in Computer Architecture, Von Neumann Model: Processor Organization- ALU, Control Unit; System Bus, Memory, I/O Devices. Multilevel model of Computer, Concept of Instruction Execution.

UNIT 2 Review of Combinational and Sequential Circuits. Memory Organization: Memory Hierarchy, Memory Properties, Main Memory, Associative Memory, Cache Memory. Machine Language Level (ISA level): Instruction Formats, Addressing Modes, Instruction Types, Flow of Control. RISC v/s CISC.

UNIT 3 Memory mapped I/O and I/O mapped I/O, I/O Techniques: Programmed I/O, Concept of Interrupts, Interrupt driven I/O and DMA, I/O Device Interfaces, I/O Processors. Serial and Parallel Communication. Computer Buses.

UNIT 4 Concept of Hardwired and Micro Programmed Control. Micro Instructions, Instruction Fetch and Queuing, Micro Instruction Control, Design of the Micro Architecture Level.

UNIT 5 Parallel Architectures: On-chip Parallelism- Instruction Level Parallelism, On-chip Multithreading, Multicore Processor Architecture. Pipelining: RISC Pipeline, Exception handling of Pipelining, Hazards of Pipelining.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. William Stallings, "Computer Organization and architecture", Ninth Edition, Pearson, 2012.
2. Tannenbaum and Austin, "Structured Computer Organization", Sixth edition, PHI, 2013.
3. Michael J. Flynn "Computer Architecture: Pipelined and Parallel Processor Design, First Edition, 1995.

REFERENCE BOOKS:

1. V. Carl Hamacher, "Computer Organization", Fifth Edition, 2011, McGraw Hill.
2. John P. Hayes, "Computer Architecture and Organization", Fourth Edition, TMH, 2003
3. Morris Mano, "Computer System Architecture", Third Edition, 2007, PHI.
4. David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", Fourth Edition, Morgan Kauffman, 2011

NEW

DEPARTMENT OF COMPUTER ENGINEERING
B.Tech. II YEAR (4YDC)
SEMESTER-A
CO24____ : OBJECT ORIENTED PROGRAMMING LAB

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	40	60	100

PRE-REQUISITES: Computer Programming

COURSE OUTCOMES (CO):

On completion of this Lab course, students will be able to

- **CO1** Identify and apply the concepts of classes, objects, members of a class and the relationships among them needed for finding the solution to a specific problem.
- **CO2** Demonstrate reusability of classes and interfaces using the concept of Inheritance and Packages.
- **CO3** Implement the concept of Multithreading, exceptions and file handling for constructing robust applications.
- **CO4** Design & execute programs using the concept of Collection frameworks.

MAPPING OF COURSE OUTCOMES TO PROGRAM OUTCOMES (H-3, M-2, L-1, or "-"):

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

UNIT 1: Identify State, Behavior & Identity of an object from real world scenarios, implement classes and candidates for Classes, Attributes and Services, methods & constructors.

UNIT 2: Identify types of relationships among objects & classes, Implement programs using the concept of inheritance & Abstraction.

UNIT 3: Implement programs demonstrating the concept of compile time & runtime polymorphism, Implement programs covering Strings classes.

UNIT 4: Demonstrate the usage of exception handling for solving real world problems. Implement the concept of multithreading using Thread class & Interface.

UNIT 5: Implementation of Collection Framework (Arraylist, Vector & Iterator). Implement programs using Tree & Hashmap. Mini Project.

NEW

DEPARTMENT OF COMPUTER ENGINEERING
B.Tech. II YEAR (4YDC)
SEMESTER-A
CO 24_____: COMPUTER ARCHITECTURE LAB

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	40	60	100

PRE-REQUISITES: NIL

COURSE OBJECTIVES:

The main objective of this course is to compare various architectures of Computers and their components like memory etc. and to develop the skills of the students to write the assembly language programs for various instructions.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Illustrate different architectures in respect of implementation and Categorize them on the basis of performance(power consumption, speed, reliability),design,cost .
2. Evaluating cache and its performance on the basis of different simulation parameters likesize of cache at different level,instruction set etc
3. Identifying limitations of MIPS programming and simulating different operations on MIPS architecture.
4. Designing various components of architecture and simulating their operation.

LAB PRACTICALS:

Unit 1: Study of computer architecture and organization, different microcontrollers and Microprocessors and commands on Linux to identify system specifications and their purpose.

Unit 2: Designing various logics on a simulator for understanding its working. Like AND, OR, NOT,universal gates.Designing various logics on a simulator for understanding its working. Like Multiplexer,adder etc.

Unit 3: Installation and understanding MIPS simulator like different types of registers(PC,integer register,floating point register etc) and basic programming.

Unit 4: Evaluating different types of instructions according to the underlying architecture using gem5.Realizing the Impact of cache size,associativity etc on the program performance using different configuration in gem5 and set of benchmark programs like mibench

Unit 5: MIPS programming to use different logical operators like add, sll, addi, srl, addiu, sub, addu, subu, and, nor, andi, xor, or, xori, ori.MIPS programming to use special set of registers and string input and its storage,manipulation.

NEW

DEPARTMENT OF COMPUTER ENGINEERING
B. E. II YEAR (4YDC)
SEMESTER-A
CO____: DESIGN THINKING LAB

* Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	40	60	100

PRE-REQUISITES: NIL

COURSE OBJECTIVES:

1. To introduce the fundamental concepts of design thinking.
2. To develop creative and innovative problem solving ability.
3. To enable students to conceive, conceptualize, design and demonstrate innovative ideas using prototypes.
4. To encourage students in adoption of innovative ways of tackling business and/or social problems.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Develop a fundamental understanding of the Design Process and how it can be applied for innovation.
2. Identify groups of stakeholders and empathize with them to understand the needs through different methods.
3. Originate creative ideas from different domains and evaluate them for appropriateness.
4. Create a prototype and perform exhaustive testing of prototype for validation.

COURSE CONTENTS:

UNIT I: Introduction to Design Thinking and its process model, Principles and tools, How to Empathize, Role of Empathy in design thinking, Empathy Maps Design, Creation of User Personas, Customer Journey Maps, Presentation of Empathy Phase.

UNIT II: Analysis and Identification of Conflict of Interest amongst Stakeholders, Problem analysis and Reformulation of the problem, Point of view phase, Problem clarification, Understanding of the problem, Problem Definition, Presentation of Define Phase.

UNIT III: Ideation phase: Need, Uses, Generation, Ideation Methods-Brainstorming-rules, Mindmap, How-Now-Wow, 4-Category Method, Presentation of Ideate Phase..

UNIT IV: Prototyping- Low and High Fidelity Prototype Generation, Lean Startup Method for Prototype Development, Presentation of Prototyping Phase..

UNIT V: Test prototyping and Validation, Kano Model, Desirability Testing, Overall presentation of complete process of Design Thinking on Case Study taken.

OLD

DEPARTMENT OF COMPUTER ENGINEERING
B.Tech. II YEAR (4YDC)
SEMESTER-B
CO 24553: DISCRETE STRUCTURES

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	-	-	100

PRE-REQUISITES: MA10001: Mathematics - I, MA10501: Mathematics - II

COURSE OBJECTIVES: To enable a student to learn the basic concepts of discrete elements such as sets, combinatorics, relations, functions, graphs, trees and their applications in diverse domains.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Solve problems which involve discrete structures such as sets, relations, functions, predicate and propositional logics.
2. Explain the notion of mathematical thinking, mathematical proofs, logical thinking, and combinatorics and be able to apply them in problem solving.
3. Describe the basic terminology and properties of graphs and trees and apply them to solve practical problems.
4. Apply algebraic techniques effectively to analyze basic discrete structures and algorithms for real world applications.

COURSE CONTENTS:

THEORY:

UNIT 1 Set Theory: Operations, Types, Set Identities, Computer Representation of sets; Principle of Mathematical Induction; Principle of Inclusion-Exclusion; Logic Theory: Propositional Logic: Well formed formula, Truth Table, Algebra of Proposition, Introduction to Predicate logic.

UNIT 2 Counting Techniques and Combinatorics; Relations and Functions: Properties and types of relations and functions, Application to RDBMS and Hashing; Pigeonhole Principle.

UNIT 3 Graphs and Trees: Basic terminologies, Types, Properties, Shortest path Algorithms, Cutsets, Hamiltonian and Eulerian paths and circuits; Tree Traversals; Spanning Trees; Applications in computer science.

UNIT 4 Recurrence Relations and Generating Functions: Introduction, Problem Solving: Introduction to Complexity of Problems and Algorithms.

UNIT 5 Algebraic Systems: Groups, Rings, Fields, Integral Domain, Lattices, Boolean Algebra.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

3. Feedback of students on attainment of cos.
4. Feedback of students on classroom learning.
5. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Kenneth H. Rosen, "Discrete Mathematics and its Applications", 7th Edition,McGraw Hill, 2017.
2. Liu C.L., "Elements of Discrete Mathematics", Fourth Edition, McGraw-Hill, 2012.
3. J.P. Tremblay and R. Manohar, "Discrete Mathematical Structures with Applications to Computer Science", First Edition, McGraw-Hill, 2017.

REFERENCE BOOKS:

1. Kolman, Busby, Ross and Rehman, "Discrete Mathematical Structures", Sixth Edition, Pearson Education, 2015.
2. NarsinghDeo, "Graph Theory with Applications to Engineering and Computer Science", First Edition, Dover Publications, 2016.
3. Seymour Lipschutz, Marc Laras Lipson, "Discrete Mathematics", Schaum's Outlines, Third Edition, 2017.

NEW

DEPARTMENT OF COMPUTER ENGINEERING
B.Tech. II YEAR (4YDC)
SEMESTER-B
CO 24553: DISCRETE STRUCTURES

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	-	-	100

PRE-REQUISITES: MA10001: Mathematics - I, MA10501: Mathematics - II

COURSE OBJECTIVES: To enable a student to learn the basic concepts of discrete elements such as sets, combinatorics, relations, functions, graphs, trees and their applications in diverse domains.

COURSE OUTCOMES:

After completing the course student should be able to:

5. Solve problems which involve discrete structures such as sets, relations, functions, predicate and propositional logics.
6. Explain the notion of mathematical thinking and logical thinking, combinatorics, Recurrence Relations and Generating Functions and be able to apply them in problem solving.
7. Describe the basic terminology and properties of graphs and trees and apply them to solve practical problems.
8. Apply algebraic techniques effectively to analyze basic discrete structures and algorithms for real world applications.

COURSE CONTENTS:

THEORY:

UNIT 1 Set Theory: Operations, Types, Set Identities, Computesr Representation of sets; **Proof Techniques: Proof by Contradiction, By Contraposition,** Principle of Mathematical Induction; Principle of Inclusion-Exclusion; Logic Theory: Introduction to Propositional Logic and Predicate Logic: Well formed formulas, Truth Table, Algebra of Proposition, **Theory of Inference, Predicates and Quantifiers**.

UNIT 2 Counting Techniques and Combinatorics; Relations and Functions: Properties and types of relations and functions, Application to RDBMS and Hashing; Pigeonhole Principle.

UNIT 3 Recurrence Relations: Introduction, **Linear Recurrence Relations;** Generating Functions; Introduction to Complexity of Problems and Algorithms, Applications and **Problem Solving;**

UNIT 4 Algebraic Systems: **Binary Operations and properties**; Groups, Rings, Fields, Integral Domain, Lattices, Boolean Algebra; **Applications and Problem Solving**.

UNIT 5 Graphs and Trees: Basic terminologies, Types, Properties, Shortest path Algorithms; Cutsets; Hamiltonian and Eulerian paths and circuits; Tree Traversals; Spanning Trees; Applications in computer science.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Kenneth H. Rosen, "Discrete Mathematics and its Applications", 7th Edition,McGraw Hill, 2017.
2. Liu C.L., "Elements of Discrete Mathematics", Fourth Edition, McGraw-Hill, 2012.
3. J.P. Tremblay and R. Manohar, "Discrete Mathematical Structures with Applications to Computer Science", First Edition, McGraw-Hill, 2017.

REFERENCE BOOKS:

1. Kolman, Busby, Ross and Rehman, "Discrete Mathematical Structures", Sixth Edition, Pearson Education, 2015.
2. NarsinghDeo, "Graph Theory with Applications to Engineering and Computer Science", First Edition, Dover Publications, 2016.
3. Seymour Lipschutz, Marc Laras Lipson, "Discrete Mathematics", Schaum's Outlines, Third Edition, 2017.

OLD

DEPARTMENT OF COMPUTER ENGINEERING
B.Tech. II YEAR (4YDC)
SEMESTER-B
CO 24507: DATA STRUCTURES

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	-	-	100

PRE-REQUISITES: CO10504: Computer Programming, CO24497: Programming Practices

COURSE OBJECTIVES: To introduce the fundamental concept of data structures and to emphasize the importance of data structures in developing and implementing algorithms.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain the basic concepts of data structures and algorithms.
2. Describe basic concepts about stacks, queues, linked lists, trees, graphs and their implementation.
3. Apply data structures to efficiently organize the data for improving performance of the system.
4. Design and implement algorithms for solving problems with the help of fundamental data Structures.

COURSE CONTENTS:s

THEORY:

UNIT 1 Introduction to Data Structure: Concepts of Data and Information, Classification of Data structures, Abstract Data Types, Implementation aspects: Memory representation, Data structures operations and its cost estimation. Introduction to linear data structures- Arrays, Linked List: Representation of linked list in memory, different implementation of linked list: Circular linked list, doubly linked list, etc. Application of linked list: polynomial manipulation using linked list, etc.

UNIT 2 Stacks: Stacks as ADT, Different implementation of stack, multiple stacks. Application of Stack: Conversion of infix to postfix notation using stack, evaluation of postfix expression, Recursion. Queues: Queues as ADT, Different implementation of queue, Circular queue, Concept of Dqueue and Priority Queue, Queue simulation, Application of queues.

UNIT 3 Tree: Definitions - Height, depth, order, degree etc. Binary Search Tree – Operations, Traversal, Search. AVL Tree, Heap, Applications and comparison of various types of tree; Introduction to forest, multi-way Tree, B tree, B+ tree, B* tree and red-black tree.

UNIT 4 Graphs: Introduction, Classification of graph: Directed and Undirected graphs, etc., Representation, Graph Traversal: Depth First Search (DFS), Breadth First Search (BFS), Graph algorithm: Minimum Spanning Tree (MST)-Kruskal, Prim's algorithms. Dijkstra's shortest path algorithm; Comparison between different graph algorithms. Application of graphs.

UNIT 5 Sorting: Introduction, Sort methods like: Bubble Sort, Quick sort. Selection sort, Heap sort, Insertion sort, Shell sort, Merge sort and Radix sort; comparison of various sorting techniques. Searching: Basic Search Techniques: Sequential search, Binary search, Comparison of search methods. Hashing & Indexing. Case Study: Application of various data structures in operating system, DBMS etc.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

3. Feedback of students on attainment of cos.
4. Feedback of students on classroom learning.
5. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. AM Tanenbaum, Y Langsam& MJ Augustein, "Data structure using C", Prentice Hall India-2007
2. Robert Kruse, Bruse Leung, "Data structures & Program Design in C", Pearson Education, 2007.
3. Richard, Gilberg Behrouz, Forouzan ,"Data structure – A Pseudocode Approach with C", Thomson press, 2005.

REFERENCE BOOKS:

1. Jean – Paul Trembly , Paul Sorenson, "An Introduction to Structure with application", TMH, 2007.
2. Aho, Hopcroft, Ullman, "Data Structures and Algorithms", Pearson Education, 2002.
3. N. Wirth, "Algorithms + Data Structure = Programs", Prentice Hall, 1978.
4. SartajSahni, " DataStructures,Algorithms and Applications in C++" Universities Press.
5. ReemaThareja, "Data Structures Unsing C", Oxford Press 2012

NEW

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. III YEAR (4YDC)
SEMESTER-A
CO 24____ : AGILE SOFTWARE METHODOLOGY

* Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	-	-	100

PRE-REQUISITES: NIL

COURSE OBJECTIVES:

1. Understand the software development terminologies.
2. Understand the differences between conventional and agile approaches.
3. Learn the background and origins of various agile and traditional methodologies.
4. Learn about Scrum development.
5. Learn about testing of developed software projects.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Describe the fundamental principles and practices associated with software development process models.
2. Compare and contrast agile software development models with traditional development models.
3. Apply techniques and skills to design and build agile projects for effective software development using different frameworks.
4. Adapt existing software testing techniques and knowledge to implement projects.

COURSE CONTENTS:

THEORY:

UNIT 1 **Software Development Methodology:** Fundamentals of Software Engineering Concepts and Process: Requirements Elicitation Methods, Functional and Non Functional requirements, Role of SRS, Deliverables, milestones and artifacts, feasibility Study, Efforts and Cost Estimation; Software Development Life Cycle: Important Steps and Effort Distribution; Overview of Software Development Models like Prototype Model, Incremental Model, Spiral Model, RAD.

UNIT 2 **Fundamentals of Agile Development Methodology**: The Genesis of Agile, Introduction and background, Agile Manifesto and principles, Agile development Lifecycle, Agile Development Methods: Adaptive Software Development (ASD), Dynamic Systems Development Methods (DSDM), Scrum, Extreme Programming (XP): XP lifecycle, Feature Driven development, Lean Software Development, Kanban, Agile project management, Test Driven Development, Key Principles.

UNIT 3 **Design and Development:** Design and Architecture, Big Design Upfront, Ivory Tower Architecture, UML Modeling: Structural and Behavioral Diagrams, Design Practices, Role of Design Principles, Automated build tools, Continuous Integration, Continuous Deployment, Refactoring, Team Dynamics and Collaboration, Introduction to Design Patterns.

UNIT 4 **Agile Framework: Scrum:** Introduction to Scrum, Agile Principles - Sprints Introduction, User Stories and Product Backlog, Estimation, Velocity, Burndown chart, Sprint Zero, Roles - Team Management and Structures, Product Owner, ScrumMaster / Team Lead, Implementation Team Members, Planning in Scrum - Planning, Planning Stakeholders, Planning Types (Portfolio, Product and Sprint), Sprint phases/meeting - Sprint Planning, Sprint Review, Sprint Retrospective, Product Demo, Daily Scrum calls.

UNIT 5 **Testing and Review**: Software Testing Principles, Practice and Processes, Difference between Testing in Traditional, Types of Testing: Black Box, White Box, Unit Testing, Integration, System Testing, Agile Metrics and Measurements, Version Controlling, Configuration Management, Quality Assurance, Quality Attributes, Code Review Practices. Case study using any one of the framework.

COURSE ASSESSMENT (Th.):

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%)
2. End semester Theory Exam (70%)

COURSE ASSESSMENT (Pr.):

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, file etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

TEXT BOOKS RECOMMENDED:

1. Robert C. Martin, Agile Software Development- Principles, Patterns and Practices, Prentice Hall, 2013.
2. Kenneth S. Rubin, Essential Scrum: A Practical Guide to the Most Popular Agile Process, Addison Wesley, 2012.
3. Ian Sommerville, Software Engineering, 10th Edition, Pearson, 2015.
4. Roger S. Pressman, Software Engineering: A Practitioner's Approach, 10th Edition, McGraw-Hill Education, 2019.

REFERENCE BOOKS:

1. Sudipta Malakar, Agile Methodologies In-Depth: Delivering Proven Agile, SCRUM and Kanban Practices for High-Quality Business Demands, BPB Publications, 2021.
2. Steve McConnell, Code Complete: A Practical Handbook of Software Construction, 2nd Edition, Microsoft Press, 2004.

NEW

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. II YEAR (4YDC)
SEMESTER-B
CO 24_____: DATA STRUCTURES LAB

* Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	40	60	100

PRE-REQUISITES: Computer Programming

COURSE OUTCOMES (CO)

On completion of this Lab course, students will be able to:

CO1 Evaluating features of different data structures and its applications

CO2 Identifying the improvement in performance like searching time,storage,computation time of a solution by using optimal data structure.

CO3 Implementing different data structures and finding efficient solutions of various problems by using them.

CO4 Implementing different searching and sorting techniques and realizing their impact on the overall performance of a solution.

LAB PRACTICALS:

Unit 1. Problems based on array data structure and its features like storage,searching,indexing etc. Optimizing different solutions on the basis of space and time complexity.

Unit 2. Implementation of singly linked list and problems based on the features of linked list(including usage of standard data structures). And different types of linked list such as doubly linked list, circular linked list, circular doubly linked list and problems related to this data structures(including usage of standard data structures)

Unit 3. Implementation of stack and queues using arrays or using linked lists and problems related to this data structure(including usage of standard data structures). . Implementation of binary trees like binary search trees and solving problems using this data structure(including usage of standard data structures) .

Unit 4. Implementation of different sorting algorithms like selection sort, insertion sort, bubble sort, merge sort, quick sort etc and solving problems with their features.Implementation of traversal algorithms like breadth first search, depth first search etc

Unit 5. Implementation of graphs and solving problems using this data structure(including usage of standard data structures) .

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. II YEAR (4YDC)
SEMESTER-B
CO24_____: SOFTWARE DESIGN LAB

* Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	40	60	100

PRE-REQUISITES:NIL

COURSE OUTCOMES: On completion of this Lab course, students will be able to:

CO1 Develop the skills to design and document requirements, software architectures using appropriate modeling tools (e.g., UML diagrams).

CO2 Demonstrate proficiency in using prototyping techniques.

CO3 Learn to create and execute comprehensive testing plans, including unit tests, integration tests, and system tests, to ensure the quality and reliability of software products.

CO4 Engage in collaborative software development, utilizing version control systems (e.g., Git), agile methodologies, and project management tools to work effectively as a part of a development team.

LAB CONTENTS:

UNIT I: Requirements Analysis: Gather and analyze user requirements for a software project through surveys, create user stories, and develop a requirements specification document, use softwares like(Jira, Monday, GitScrum).

UNIT II: Designing: To create UML diagrams to visualize software design through use case diagrams, class diagrams, sequence diagrams, and activity diagrams for a given software project..

UNIT III:Prototyping: To create a prototype for a software application, develop a low-fidelity or high-fidelity prototype using wireframing tools or prototyping software, practice CI/CD.

UNIT IV: Agile Methodologies: Participate in agile practices such as sprint planning, daily stand-ups, and retrospectives while developing a software project.

UNIT V: Testing: Perform unit testing, integration testing, system testing, version control(use Git for branching, merging, code versions), technical documentation(user manual, API documentation etc..).

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. II YEAR (4YDC)
SEMESTER-B
CO24_____: MOBILE APPLICATION DEVELOPMENT LAB

* Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	2	2	-	3	-	-	40	60	100

PRE-REQUISITES: Knowledge of CoreJava, Object Oriented Programming Systems (OOPS), DBMS and SQL

COURSEOBJECTIVES: This course enables student in proficiency of creating mobile applications.

COURSE OUTCOMES: After completing the course student should be able to:

- **CO1** Describe the basic building blocks of Android and iOS architecture and framework and Apply basic UI design, concept of Views and ViewGroups and different Layouts and widgets in Android studio.
- **CO2** Use styles and themes and create mobile apps with databases using SQLite.
- **CO3** Implement and use various services in mobile applications.
- **CO4** Successfully create, test and deploy working android apps on Google Play Store and iOS apps on Apple Playstore.

COURSE CONTENTS: (To be instructed in Tutorials &Lab)

Unit 1: Android Installation, Building you First Android application, Understanding Anatomy of Android Application, Android Manifest file. Implementation of an android application that uses Layout Managers and event listeners.

Unit 2: Implementation of permissions for accessing camera, location, and storage, custom menu in various ways and handle menu item selections. Creating Android apps with databases using SQLite.

Unit 3: Creating push Notification while using database like MYSQL etc.,Creating Firebase for JSON database and notifications, Testing app.

Unit 4: Devloping an android application that creates an alert upon receiving a message. And using Android Sensor.

Unit 5: Introduction and usage of to Xcode, UIKit and SwiftUI, ARKit, CoreML and CoreData for iOS.

TEXT BOOKS RECOMMENDED:

1. Android Programming for Beginners: Build in-depth, full-featured Android apps starting from zero programming experience, 3rd Edition, John Horton, Packt Publishing. 2021, ISBN 978-1-80056-343-8

2. Android App Development in Android Studio: Java + Android Edition for Beginners – First Edition, J. Paul Cardle, 2017, ISBN 978-1542885843

REFERENCE BOOKS:

Web Resources:

1. https:// info448-s17.github.io/lecture-notes/services.html

2. https://docs.flutter.dev/reference/tutorials

3. https://developer.android.com/www.freecodecamp.com www.linkedinlearning.com