Computer Organization and Design
BEG271C0

Year: II **Semester: III**

Teaching Schedule Hours/Week			Examination Schedule				
Theory	Tutorial	Practical	Internal Assessment		Final		Total
			Theory	Practical	Theory	Practical	
3	1	2/2	20	25	80	-	125

Course Objective: To introduce students about the organization of computer structure and the implementation of its architecture.

Course Contents:

1. Overview of Computer Architecture and Organization (3 Hours)
   1.1 Introduction
   1.2 Contrast between computer architecture and organization
   1.3 Fundamentals of computer architecture
   1.4 Organization of Von-Neumann machine

2. Computer Instruction (4 Hours)
   2.1 Instruction format
   2.2 Instruction cycle
   2.3 Instruction types and addressing modes

3. Computer Arithmetic (5 Hours)
   3.1 Representation of integers and real numbers
   3.2 Algorithm of Addition, Subtraction, Multiplication and Division

4. Memory system organization and Architecture (4 Hours)
   4.1 Memory system hierarchy
   4.2 Main memory Organization
   4.3 Cache memory
   4.4 Virtual memory

5. Interfacing and Communication (4 Hours)
   5.1 I/O fundamentals
   5.2 I/O techniques
   5.3 Interrupt
   5.4 Memory system design and interfacing
   5.5 Buses

6. Device subsystem (3 Hours)
   6.1 External storage system
   6.2 RAID architecture

7. Control Unit Design (7 Hours)
   7.1 Instruction sequencing
   7.2 Instruction Interpretation
   7.3 Control memory
   7.4 Hardwired control
   7.5 Micro-programmed control
   7.6 Micro-programmed computer's

8. **Input-Output Organization** **(4 Hours)**
   8.1 Bus control
   8.2 Serial I/O: Asynchronous and synchronous modes, USART and VART

9. **Parallel Data Transfer** **(4 Hours)**
   9.1 Asynchronous and Synchronous program controlled 9.2
   Interrupt Driven and DMA modes
   9.3 interrupt and DMA controller

10. **Trends in Computer architecture** **(3 Hours)**
    10.1 CISC
    10.2 RISC
    10.3 VLIW

11. **lip** **(4 Hours)**
    11.1 Introduction to ILP
    11.2 Pipeline hazards: Structural hazards, Data and control hazards 11.3
    Reducing the effects of hazards

Practicals: Lab implementation of the following algorithms:

1. Addition
2. Subtraction
3. Unsigned and signed multiplication
4. Cache memory mapping

Reference Books:

1. J. P. Hayes, Computer Architecture and Organization, McGraw Hill, 3rd Ed., 1998
2. M. M. Mano, Computer System Architecture, Pearson, 3rd Ed., 2004
3. V. C. Hamacher, Z. G. Veranesic, & S. G. Zaky, "Computer Organisation", Tata McGraw Hill, 5th Ed., 2002
4. W. Stallings, "Computer Organization and Architecture — Designing for Performance", Prentice Hall of India, 7th Ed., 2007
5. D. A. Pattersen and J. L. Hennesy, "Computer Organization and Design: The Hardware Software Interface", Elsevier, 2nd Ed., 2006

Data Structure and Algorithm

BEG273C0

Year: II Semester: III

Teaching Schedule Hours/Week			Examination Scheme				
Theory	Tutorial	Practical	Internal Assessment		Final		Total
			Theory	Practical	Theory	Practical	
3	-	3	20	50	80	-	150

Course Objective: To understand the fundamental concept of data structure. On the completion of this course the student will be able to design data structure and implement it using programming language.

Course Contents:

1.0 Introduction to data structure (2 Hrs)

- 1.1 Concept of data structure and its uses
- 1.2 Abstract data type (ADT): definition and importance
- 1.3 Implementation of data structure
- 1.4 Introduction and application of Big 0 notation

2.0 The Stack (2 Hrs)

- 2.1 Stack as ADT
- 2.2 Operation in stack and stack implementation
- 2.3 Application: evaluation of infix, postfix, and prefix expression

3.0 Queue (3 hrs)

- 3.1 Queue as an ADT, queue Implementation
- 3.2 Operation in queue: enque and deque
- 3.3 Linear and circular queue, and their application
- 3.4 Priority queue: definition and application

4.0 List (2 hrs)

- 4.1 Definition
- 4.2 Static and Dynamic list structure
- 4.3 Array Implementation of Lists, Stacks, and Queues as continuous list

5.0 Linked Lists (5 Hrs)

- 5.1 Definition
- 5.2 Linked list as an ADT
- 5.3 Implementation
- 5.4 Operation in linked list: node insertion, deletion, insertion and deletion after and before nodes
- 5.5 Applications of linked stack and queue
- 5.6 Doubly linked list and its applications
- 5.7 Circular linked list

6.0 Recursion (4 Hrs)

- 6.1 Recursion and principle of recursion
- 6.2 Need and importance of recursion
- 6.3 Recursion and iteration algorithm, Converting recursion to iteration
- 6.4 TOH and fibonacci sequences and recursion
- 6.5 Applications of recursion

7.0 Trees **(6 Hrs)**

- 7.1 Tree concept
- 71 Basic operation in tree: insertion, deletion and search 7.3 Tree height, depth, and level
- 7.4 Binary tree traversals (pre-order, post-order and in-order) 7.5 AVL balance trees, balancing algorithm
- 7.6 Huffman tree and its application

8.0 Sorting **(6 Hrs)**

- 8.1 Definition
- 8.2 Types of sort: internal and external sort
- 8.3 Insertion and selection sort, exchange sort
- 8.4 Quick sort and merge sort
- 8.5 Shell sort
- 8.6 Binary sort
- 8.7 Heap and heap sort as priority queue
- 8.8 Efficiency of sorting

9.0 Searching **(7 Hrs)**

- 9.1 Definition of searching
- 9.2 Searching technique
- 9.3 Essential of search
- 9.4 Types of search: Sequential, binary, tree, general search tree
- 9.5 Hashing: Hash function and Hash tables, Collision resolution technique 9.6 Efficiency comparisons of different search technique
  - 9.6.1 Big OH, Big Sigma, and Big Omega Notation
  - 9.6.2 Calculation of 0(-) for a simple program

10.0 Graphs **(8 Hrs)**

- 10.1 Definition of graph
- 10.2 Representation and applications
- 10.3 Graphs as an ADT
- 10.4 Transitive closure
- 10.5 Graphs types, graph traversal and spanning forests
- 10.6 Kruskal's and shortest path algorithm

Practicals: The practicals should cover all the above chapters of this course in a high-level programming language.

<u>Reference Books:</u>

1. Y. Langsam, M. J. Augenstein & A. M. Tanenbaum, "Data Structures using C and C++", PHI
2. G. W. Rowe, "Introduction to Data Structure and Algorithms with C and C++", PHI
3. R. L. Kruse, B. P Leung, C. L. Tondo, "Data Structure and Program Design in C", PHI

Electronic Devices and Circuits

BEG230EC

Year: II **Semester: III**

Teaching Schedule Hours/Week			Examination Scheme				
Theory	Tutorial	Practical	Internal Assessment		Final		Total
3	1	2	Theory	Practical	Theory	Practical	125
			20	25	80	-	

Course Objective: To introduce students about the working principles and applications of semi-conductor devices such as diodes, transistors, and FETs.

Course Contents:

1. Semiconductor diode **[8 hrs]**

1.1 Review of p-n junction diode
1.2 Analysis of diode circuits
1.3 Applications of p-n junction diode
1.3.1 Clipping and Clamping circuits
1.3.2 Rectification (half wave, full wave and bridge rectifier)
1.4 Types of diode (Schottky, varactor, tunnel, zener)
1.5 Zener diode as a voltage regulator

2. Bipolar Junction Transistor **[18hrs]**

2.1 Construction of a 13.1T 2.2
Ebers-Molls Equation
2.3 Basic Transistor Equation 2.4
CB, CC, CE Configurations 2.5
Load line analysis
2.6 Transistor as an amplifier 2.7
Types of biasing
2.8 Biasing stabilization and thermal runaway
2.9 Small signal analysis (h-parameter and r_e model)
2.10 High Frequency t-model

3. Applications of BJT **[Ilhrs]**

3.1 Power amplifiers (Class A, B, C, AB and efficiency calculation) 3.2
BJT as a switch
3.3 Cascaded amplifier (Single stage and multistage)
3.4 Untuned amplifier
3.4.1 Frequency and phase response of RC coupled amplifier 3.5
Differential Amplifiers

4. Field Effect Transistors **[8hrs]**

4.1 Junction field effect transistor (JFET)
4.1.1 Construction and characteristics
4.1.2 Biasing of JFET
4.1.3 Small signal analysis of JFET
4.1.4 UJT as an oscillator
4.2 MOSFET
4.2.1 Construction, characteristics and types
4.2.2 Biasing of MOSFET
4.2.3 NMOS (Depletion and enhancement type) 4.2.4
Introduction to CMOS

Practicals (in Trainer kits, Multisim and P-Spice):

1. Measurement of characteristics of diode, zener diode
2. Rectifier circuits
3. Measurement of input and output characteristics of CE configurations
4. Single stage BJT amplifier
5. Measurement **of** input and output characteristics of IFET
6. **Measurement** of input and output characteristics of MOSFET

Reference Books:

1. A. S. Sedra & K. C. Smith, *"Microelectronic Circuit?,* 6th Edition, Oxford University Press
2. Theodorre S. Bogart, *"Electronic Devices and Circuits"*
3. Millman & Halkias, *"Electronic Devices and Circuits",* McGraw Hill
4. Robert Boylestad, *"Electronic Devices and Circuits"*
5. **M. N. Horenstein,** *"Microelectronic Circuits and Devices",* Second Edition, Prentice Hall

Information System Design

BEG270C0

Year: II Semester: **III**

Teaching Schedule Hours/Week			Examination Scheme				
Theory	Tutorial	Practical	Internal Assessment .		Final		Total
			Theory ..	Practical	Theory	Practical	
3	1	-	20	-	80	-	100

Course Objective: To provide the basics of designing the information systems. Course Contents:

I. OVERVIEW OF INFORMATION SYSTEM (4hrs)

a. Types of information: operational, tactical, strategic
b. Why information systems
c. Role of Information system
d. Organizations and Information systems
e. Major types of systems in organizations
f. Managers decision making and information systems
g. System Analysis and Design

2. STRUCTURING SYSTEM REQUIREMENTS :Process Modeling (5hrs)

a. What is Process Modeling
b. Introduction to Data flow diagrams (DFD)
c. Data flow diagramming rules
d. Context Diagrams
e. Using Data Flow Diagrams in the Analysis Process

3. STRUCTURING SYSTEM REQUIREMENTS :Logic Modeling (5hrs)

a. Logic Modeling
b. Decision table
c. Decision tree
d. Structured English
e. Deciding among Structured English, Decision table and Decision tree

4. STRUCTURING SYSTEM REQUIREMENTS: Conceptual Data Modeling (4hrs)

a. Conceptual Model
b. Introduction to ER Model
c. Conceptual data modeling and ER Model
d. Role of CASE in conceptual data modeling

5. OBJECT ORIENTED ANALYSIS AND DESIGN (GOAD) (Stirs)

a. Object Oriented Development Life Cycle
b. Difference between Object Oriented Development Life Cycle and Traditional SDLC
c. Unified Modeling Language (UML)
d. Use Case Modeling
e. Object Modeling: Class Diagrams
f. Dynamic Modeling: State Diagrams, Sequence Diagrams
g. Analysis vs Design

6. DESIGNING DATABASES: Logical Data Modeling (4brs)

a. Logical Database Design
b. Relational Database Model
c. Concept of Normalization (1NF, NF, 3NF)
d. Merging Relations

7. DESIGNING PHYSICAL FILES AND DATABASES (4hrs)

a. Physical File and Database Design
b. Designing Fields
c. Designing Physical Records
d. Designing Physical Files
e. Designing databases

8. STRUCTURE CHART AND MODULAR DESIGN (6hrs)

a. Structure Chart
b. Transaction Centered Designs
c. Transform Central Designs
d. Transform Analysis
e. Modularity, Benefits of Modular Design
f. Coupling
g. Cohesion

9. IMPLEMENTATION AND MAINTENANCE (5hrs)

a. System Implementation
b. Software Application Testing
c. Types of Testing
d. Installation
e. Documenting the System
f. Training and Supporting User
g. Project close down
h. Maintaining Information System

10. DESIGNING DISTRIBUTED SYSTEM (3hrs)

a. Distributed systems for LAN
b. File Server and Client Server Architecture
c. Managing data in Distributed System
d. Alternative Designs for Distributed Systems

e. Distributed systems for LAN
f. File Server and Client Server Architecture
g. Managing data in Distributed System
h. Alternative Designs for Distributed Systems

Referenee Books:

Jeffrey A. Hoffer, Joey F. George, Joseph S. Valarich, "Modern Systems Analysis & Design"; Pearson Education, Second Edition

2. Whitten, Jeffrey L., 3rd Edition, "Systems Analysis and Design Methods"

Mathematics-III

BEG201SH

Year: II **Semester: III**

Teaching Schedule Hours/week			Examination Scheme						Total Marks	Remarks
			Final				Internal Assessments			
			Theory		Practical		Theory Marks	Practical Marks		
L	T	P	Duration	Marks	Duration	Marks				
3	2	-	3	80			20		**100**	

Objectives: The purpose of this course is to round out the student's preparation more sophisticated applications with an introduction of linear algebra, a continuous of the study of ordinary differential equations and an introduction to vector algebra and Fourier series.

1.0 Matrices and Determinant **14 Hrs**

1.1 Matrix and Determinant
1.2 Vector Space (Introduction), Dependent and Independent vectors
1.3 Linear Transformation
1.4 System of Linear Equations, Gauss elimination method only
1.5 Inverse of Matrix (Gauss Jordan Method)
1.6 Rank of the Matrix
1.7 Eigen Values of Matrix, Eigen Vectors and its applications

2.0 Laplace Transformation **10 Hrs**

2.1 Introduction
2.2 Laplace Transform of some Elementary Functions
2.3 Properties of Laplace Transform
2.4 Inverse Laplace Transforms
2.5 Application to differential equations

3.0 Line, Surface and Volume Integrals **9** Hrs

3.1 Definition of Line Integral
3.2 Evaluation of Line Integral
3.3 Evaluation of Surface and Volume Integrals
3.4 Diritchlet Integrals

4.0 Integral Theorems 6 Hrs

4.1 Greens Theorem in the plane
4.2 Stoke's Theorem (without proof)
4.3 Gauss Divergence Theorem (without proof)
4.4 Consequences and Applications of Integral Theorems

5.0 Fourier Series 6 Hrs

5.1 Periodic Function
5.2 Trigonometric Series
5.3 Fourier Series
5.4 Determination of Fourier Coefficients: Euler Formulae (-pi, pi)
5.5 Fourier Series in the Intervals (0, 2pi) and *(-l, l)*

5.6 Even and Odd Functions and their Fourier Series: Fourier Cosine & Sine Series
5.7 Half Range Function
5.8 Parsevals Formula
5.9 Fourier Series in Complex Form (Introduction)

Reference Books:

1. E. Kreyszig, Advanced Engineering Mathematics - 5th Edition, Wiley, New York.
2. **A** Text Book of Engineering Mathematics Vol. **E -** P. R. Pokharel.
3. A Text Book of Engineering Mathematics Vol. Ill-N. B. Khatakho & S. P. Pradhanang.

Project-I

BE G27803

Year: **II** Semester: **III**

Teaching Schedule Hours f			Examination Scheme				
Theory	Tutorial	Practical	Internal Assessment		Final		Total
			Theory	Practical	Theory	Practical .	

Course Objective:

To design and complete a software project in a high-level language (C or C++). On the completion of the project, students will be able to develop a small scale software in C or C-f-+ programming language.

Course Contents:

There should be a total of 60 hours covering important features of a high-level language (C or C++). A software development project will be assigned to students in a group (upto 4). A relevant topic shall be identified and instructed to each group. Students must develop the assigned software, submit written report, and give oral presentation.

General Procedure:

1. Topic Selection
2. Information Gathering
3. System Requirements Specifications
4. Algorithms and Flowcharts
5. Coding
6. Implementation
7. Documentation

The project document shall include the following:

1. Technical description of the project
2. System aspect of the project
3. Project tasks and time-schedule
4. Project team members
5. Project supervisor
6. Implementation of the project