Semester-I

Major/DS Course (Core)
COMP 1011: Computer Fundamentals & Digital Logic
Credit: 03 **45 Hours**

Course Objective

This course introduces the students to the fundamental concepts of digital computer organization and design. It aims to develop a basic understanding of the building blocks of the computer system and highlights how these blocks are organized together to architect a digital computer system. The course teaches the fundamentals of digital systems, applying the logic design and development techniques. This course forms the basis for the study of advanced subjects like Computer Architecture and Organization, Microprocessor through Interfacing, VLSI Designing etc.

Course Learning Outcomes

On successful completion of the course, students will be able to:

i. Acquire the basic knowledge of digital logic to understand digital electronics circuits.

ii. Prepare students to perform the analysis and design of various digital electronic circuits.

iii. Design and simplify combinational and sequential circuits using basic building blocks.

iv. Represent data in binary form, convert numeric data between different number systems and perform arithmetic operations in binary.

v. Simulate the design of a basic computer using a software tool/ digital trainer kit.

Syllabus

Computer Fundamentals (15 Hours)

Introduction to Computer and Problem Solving: Information and Data. Hardware: CPU, Primary and Secondary storage, I/O devices, Bus structure Software: Systems and Application. Generation of Computers: Super, Mainframe, Mini and Personal Computer. Introduction to Programming Languages: Machine Language, Assembly Language, High Level Language. Problem Solving: Flow Charts, Decision Tables and Pseudo codes. Number Systems and Codes: Number representation: Weighted Codes, Non-weighted codes, Positional, Binary, Octal, Hexadecimal, Binary Coded Decimal (BCD), Conversion of bases. Complement notions. Binary Arithmetic, Binary Codes: Gray, Alphanumeric, ASCII, EBCDIC; Single Error-Detecting and Correcting Codes, Hamming Codes, IEEE 754 floating point representation. Boolean algebra: Fundamentals of Boolean algebra, Switches and Inverters, Functionally Complete Gates (AND, OR, NOT), NAND, NOR, Switching function and Boolean Function. De Morgan's Theorem, Minterms, Maxterms, Truth table and minimization of switching function up to four variables, Algebraic and K-map method of Logic circuit synthesis: Two-level and Multi-level

Digital Logic(30 Hours)

Combinational Circuits: Realization of AND , OR Gates using diodes and NOT Gate using transistors, Standard Gate Assemblies, IC chips packaging nomenclature, Half and Full Adder(3 & bit), Multi-bit adders – Ripple carry and Carry Look Ahead Adder, Adder/subtractor, BCD-Adder, Data selectors/multiplexers – expansions, reductions, function realization, universal function 5 realization, multi-function realization, Decoders: function realization, De-multiplexer and function realization, Encoder, Priority Encoder, Parity bit Generator/checker, Gray Code Generator, Code Converters, Keyboard encoder, Seven segment display unit, Comparators. Sequential Circuits: Model of Sequential computing, Difference between Combinational and Sequential circuit, RS-Latch: using NAND and NOR Gates, RS Latch as a Static RAM Cell, Problems of Basic Latch circuits, Digital Clock – Duty Cycle, Rising time, Falling time, Clocked Flip Flops - SR, JK, D, T, Level Trigger and Edge Trigger, Excitation Functions of each flip-flops, Flip-flops with Preset and Clear, Application of Flip-flops: Asynchronous Counter(UP/DOWN) up to 4-bit counter, Decade Counter, Mod – n Counter, Finite State machine Model – State Transition Diagram and Table, Synchronous Counters – different mod counters, Ring counter, Johnson’s Counter, Registers, Registers with parallel load, Shift Registers.

Practical :: Digital Circuit Design

Credit: 01 **30 Hours**

Combinational Circuits:

1) Implement Half Adder/Half Subtractor/Full Adder/Full Subtractor using Logic Gates. Realize a logic function using basic/universal gates in SOP and POS form. Study the functionalities of 7483 and design a BCD adder using 7483 or equivalent.
2) Design of two level AND – OR, NAND –NAND, NOR-NOR circuits to realize any truth table. Realize XOR in two level and multilevel.
3) Design a 4 bit 2’s complement adder – subtractor unit using 7483 or equivalent and XOR gates.
4) Design a circuit to convert BCD numbers to corresponding gray codes.
5) Design a 4:1 MUX using NAND gates. Study of 74153 and 74151. Design Full Adder/Subtractor using MUX.
6) Design a 2:4 decoder using NAND gates. Study of 74155 and 74138. Design Full Adder/Subtractor using decoders.
7) Design a parity generator/checker using basic gates.
8) Design magnitude comparator using basic/universal gates. Study of 7485.
9) Design a seven-segment display unit.

Sequential Circuits:

1) Realize S-R, D, J-K and T flip-flop using basic gates. (Study the undefined state in S-R flip-flop).
2) Design a shift register (shift left and shift right) using flip-flops. (Study the functional characteristic of IC 74194 with emphasis on timing diagram).
3) Design Asynchronous and Synchronous counters. Study of IC 74193.
4) Study the functional characteristics of RAM IC chip. Study of open collector and tri-state output. Horizontal and vertical expansion of RAM chips by cascading. Use 74189, 7489, 2114 or any available chip.

Reference Books:

1. Digital Logic and Computer Design by M.Morris Mano, PHI
2. Digital Fundamentals by Floyd, Pearson Education
3. Digital Principle and Applications by Malvino & Leach, TMH

4 P. K. Sinha & Priti Sinha , Computer Fundamentals‖, BPB Publications, 2007.

5 Dr. Anita Goel, Computer Fundamentals, Pearson Education, 2010.

Skill Enhancement Course (SEC)

Credit: 03 **90 Hours**

COMP 1051: Programming in Python (Practical)

Course Objective

The course is designed to introduce programming concepts using Python to students. The course aims to develop structured as well as object-oriented programming skills using Python. The course also aims to achieve competence amongst its students to develop correct and efficient Python programs to solve real life problems.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Develop, document, and debug modular Python programs of reasonable complexity.
2. Implement arrays and user defined functions in Python.
3. Solve real life problems of reasonable complexity using suitable and efficient programming constructs in Python.
4. Solve real life problems of reasonable complexity using the concepts of object-oriented programming in Python.

Syllabus

Planning the Computer Program: Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation. (20 Hrs)

Techniques of Problem Solving: Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming.(20 Hrs)

Overview of Programming: Structure of a Python Program, Elements of Python (10 Hrs)

Introduction to Python: Python Interpreter, Using Python as calculator, Python shell, Indentation. Atoms, Identifiers and keywords, Literals, Strings, Operators(Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment, Operator, Ternary operator, Bit wise operator, Increment or Decrement operator, List , Tuple , Set and Dictionary.(20 Hrs)

Creating Python Programs: Input and Output Statements, Control statements (Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.), Defining Functions, default arguments, function , structures (20Hrs)

Reference Books

1. T. Budd, Exploring Python, TMH, 1st Ed, 2011
2. Python Tutorial/Documentation www.python.or 2015
3. Allen Downey, Jeffrey Elkner, Chris Meyers , How to think like a computer scientist : learning with Python , Freely available online.2012
4. http://docs.python.org/3/tutorial/index.html
5. http://interactivepython.org/courselib/static/pythonds
6. http://www.ibiblio.org/g2swap/byteofpython/read/

Lab Work:

Section: A (Simple programs)

- Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice.
- WAP to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria :

  Grade A: Percentage >=80

  Grade B: Percentage>=70 and <80Grade C:
  Percentage>=60 and <70Grade D:
  Percentage>=40 and <60 Grade E:
  Percentage<40

Section: B (Visual Python):

All the programs should be written using user defined functions, wherever possible.

1. Write a menu-driven program to create mathematical 3D
   objects I. curve
   II. sphere
   III. cone
   IV. arrow
   V. ring
   VI. cylinder.
2. WAP to read n integers and display them as a histogram.
3. WAP to display sine, cosine, polynomial and exponential curves.
4. WAP to plot a graph of people with pulse rate p vs. height h. The values of p and h are to be entered by the user.
5. WAP to calculate the mass m in a chemical reaction. The mass m (in gms) disintegrates according to the formula m=60/(t+2), where t is the time in hours. Sketch a graph for t vs. m, where t>=0.
6. A population of 1000 bacteria is introduced into a nutrient medium. The population p grows as follows:

   P(t) = (15000(1+t))/(15+ *e)*

   where the time t is measured in hours. WAP to determine the size of the population at given time t and plot a graph for P vs t for the specified time interval.
7. Input initial velocity and acceleration, and plot the following graphs depicting equations of motion:
   I. velocity wrt time (v=u+at)
   II. distance wrt time (s=u*t+0.5*a*t*t)
   III. distance wrt velocity (s=(v*v-u*u)/2*a)

Minor Courses (For other allied discipline):

Semester-I
COMP 1021: Computer Fundamentals & Digital Logic
Credit: 03 **45 Hours**

Course Objective

This course introduces the students to the fundamental concepts of digital computer organization and design. It aims to develop a basic understanding of the building blocks of the computer system and highlights how these blocks are organized together to architect a digital computer system. The course teaches the fundamentals of digital systems, applying the logic design and development techniques. This course forms the basis for the study of advanced subjects like Computer Architecture and Organization, Microprocessor through Interfacing, VLSI Designing etc.

Course Learning Outcomes

On successful completion of the course, students will be able to:

i. Acquire the basic knowledge of digital logic to understand digital electronics circuits.

ii. Prepare students to perform the analysis and design of various digital electronic circuits.

iii. Design and simplify combinational and sequential circuits using basic building blocks.

iv. Represent data in binary form, convert numeric data between different number systems and perform arithmetic operations in binary.

v. Simulate the design of a basic computer using a software tool/ digital trainer kit.

Syllabus

Computer Fundamentals (15 Hours)

Introduction to Computer and Problem Solving: Information and Data. Hardware: CPU, Primary and Secondary storage, I/O devices, Bus structure Software: Systems and Application. Generation of Computers: Super, Mainframe, Mini and Personal Computer. Introduction to Programming Languages: Machine Language, Assembly Language, High Level Language. Problem Solving: Flow Charts, Decision Tables and Pseudo codes. Number Systems and Codes: Number representation: Weighted Codes, Non-weighted codes, Positional, Binary, Octal, Hexadecimal, Binary Coded Decimal (BCD), Conversion of bases. Complement notions. Binary Arithmetic, Binary Codes: Gray, Alphanumeric, ASCII, EBCDIC; Single Error-Detecting and Correcting Codes, Hamming Codes, IEEE 754 floating point representation. Boolean algebra: Fundamentals of Boolean algebra, Switches and Inverters, Functionally Complete Gates (AND, OR, NOT), NAND, NOR, Switching function and Boolean Function. De Morgan’s Theorem, Minterms, Maxterms, Truth table and minimization of switching function up to four variables, Algebraic and K-map method of Logic circuit synthesis: Two-level and Multi-level.

Digital Logic(30 Hours)

Combinational Circuits: Realization of AND ,OR Gates using diodes and NOT Gate using transistors, Standard Gate Assemblies, IC chips packaging nomenclature, Half and Full Adder(3 & bit), Multi-bit adders – Ripple carry and Carry Look Ahead Adder, Adder/subtractor, BCD-Adder, Data selectors/multiplexers – expansions, reductions, function realization, universal function 5 realization, multi-function realization, Decoders: function realization, De-multiplexer and function realization, Encoder, Priority Encoder, Parity bit Generator/checker, Gray Code Generator, Code Converters, Keyboard encoder, Seven segment display unit, Comparators. Sequential Circuits: Model of Sequential computing, Difference between Combinational and Sequential circuit, RS-Latch: using NAND and NOR Gates, RS Latch as a Static RAM Cell, Problems of Basic Latch circuits, Digital Clock – Duty Cycle, Rising time, Falling time, Clocked Flip Flops - SR, JK, D, T, Level Trigger and Edge Trigger, Excitation Functions of each flip-flops, Flip-flops with Preset and Clear, Application of Flip-flops: Asynchronous Counter(UP/DOWN) up to 4-bit counter, Decade Counter, Mod – n Counter, Finite State machine Model – State Transition Diagram and Table, Synchronous Counters – different mod counters, Ring counter, Johnson's Counter, Registers, Registers with parallel load, Shift Registers.

Practical :: Digital Circuit Design

Credit: 01 **30 Hours**

Combinational Circuits:

1) Implement Half Adder/Half Subtractor/Full Adder/Full Subtractor using Logic Gates. Realize a logic function using basic/universal gates in SOP and POS form. Study the functionalities of 7483 and design a BCD adder using 7483 or equivalent.
2) Design of two level AND – OR, NAND –NAND, NOR-NOR circuits to realize any truth table. Realize XOR in two level and multilevel.
3) Design a 4 bit 2's complement adder – subtractor unit using 7483 or equivalent and XOR gates.
4) Design a circuit to convert BCD numbers to corresponding gray codes.
5) Design a 4:1 MUX using NAND gates. Study of 74153 and 74151. Design Full Adder/Subtractor using MUX.
6) Design a 2:4 decoder using NAND gates. Study of 74155 and 74138. Design Full Adder/Subtractor using decoders.
7) Design a parity generator/checker using basic gates.
8) Design magnitude comparator using basic/universal gates. Study of 7485.
9) Design a seven-segment display unit.

Sequential Circuits:

i. Realize S-R, D, J-K and T flip-flop using basic gates. (Study the undefined state in S-R flip-flop).
ii. Design a shift register (shift left and shift right) using flip-flops. (Study the functional characteristic of IC 74194 with emphasis on timing diagram).
iii. Design Asynchronous and Synchronous counters. Study of IC 74193.
iv. Study the functional characteristics of RAM IC chip. Study of open collector and tri-state output. Horizontal and vertical expansion of RAM chips by cascading. Use 74189, 7489, 2114 or any available chip.

Reference Books

a. Digital Logic and Computer Design by M.Morris Mano, PHI
b. Digital Fundamentals by Floyd, Pearson Education
c. Digital Principle and Applications by Malvino & Leach, TMH
d. P. K. Sinha & Priti Sinha , Computer Fundamentals‖, BPB Publications, 2007.
e. Dr. Anita Goel, Computer Fundamentals, Pearson Education, 2010.

Multi/Interdisciplinary courses

(For other discipline)

Credit: 03 **45 Hours**

COMP 1031: Basic IT Tools(Theory)

Course Objective

The goal of this course is to present overview of IT tools used in day to day use of computers and data base operations. The Course has been designed to provide knowledge on various hardware and software components of computer, operating system, various packages used for different applications, data base concepts & operations and various issues related to IT and application of IT.

Course Learning Outcomes:

On successful completion of the Course, a student will :

i. Acquire the foundation level knowledge required to understand computer and its operations.
ii. Understand the hardware and software components of the computer.
iii. Understand the basic concept of operating system and get knowledge about various different operating systems.
iv. Understand to use the packages of word processing, spread sheet and presentation in detail.
v. Understand various data base concepts and operations.
vi. Understand the issues related to IT and IT applications
vii. Prepare research and academic related presentations.

Syllabus

Introduction – Introduction to computers – Evolution – Generation of Computers – Computers Hierarchy – Applications of Computers. (5 Hrs)

Windows Basics – Introduction to word – Editing a document - Move and Copy text - Formatting text & Paragraph – Enhancing document – Columns, Tables and Other features.(10 Hrs)

Introduction to worksheet and shell – getting started with Excel – Editing cell & using Commands and functions – Moving & Copying , Inserting & Deleting Rows & Columns - Printing work sheet.(5 Hrs)

Creating charts – Naming ranges and using statistical, math and financial functions, database in a worksheet – Additional formatting commands and drawing toolbar – other commands & functions – multiple worksheet and macros.(10 Hrs)

Introduction to Database Development: Database Terminology, Objects, Creating Tables, working with fields, understanding Data types, Changing table design, Assigning Field Properties, Setting Primary Keys, Select data with queries: Creating simple Query by design & by wizard (10 Hrs)

Overview of Power point – presenting shows for corporate and commercial using Power point –Introduction to Desktop publishing – Computer viruses – Introduction to Internet – Web features.(5 hrs)

Reference Books:

i. Swinford, E., Dodge, M., Couch, A., Melton, B. A. (2013). Microsoft OfficeProfessional 2013. United States: O'Reilly Media.

ii. Wang, W. (2018). Office 2019 For Dummies. United States: Wiley. Microsoft Lambert, J. (2019). Microsoft Word 2019 Step by Step. United States: Pearson Education.

iii. Jelen, B. (2013). Excel 2013 Charts and Graphs. United Kingdom: Que.

iv. Alexander, M., Jelen, B. (2013). Excel 2013 Pivot Table Data Crunching. UnitedKingdom: Pearson Education.

v. Alexander, M., Kusleika, R. (2018). Access 2019 Bible. United Kingdom: Wiley.