

[Type text]

# Institute for Excellence in Higher Education (IEHE), Bhopal

**B.Sc. (HONOURS): PART-I**
**Semester: I**
**SUBJECT: COMPUTER SCIENCE**
**PAPER: HONOURS-I**

Max. Marks : 100

Time : 03 Hrs

## TITLE: Digital Computer Organization

**Objectives:-** Digital representation of data in a computer system. Understand the general concepts in digital logic design, including logic elements, and their use in combinational and sequential logic circuit design. The student will be able to Identify, understand and apply different number systems and codes.

**Course Learning Outcomes:-** Upon completion of the course, students should possess the following knowledge and skills:

- An understanding of a machine's instruction set architecture (ISA) including basic instruction fetch and execute cycles, instruction formats, control flow, and operand addressing modes.
- An understanding of the design and functioning of a machine's central processing unit (CPU) including the data path components (ALU, register file) and the control unit.
- An understanding of basic input/output functioning including program controlled I/O and interrupt I/O.
- An understanding of organization of memory hierarchies including the basics of cache design and DRAM architectures.
- Analyze the performance of processors and caches

## SYLLABUS

### **Data types and Number systems:**

Binary number system, Octal & Hexa-decimal number system, 1's & 2's complement, Binary Fixed-Point Representation, Arithmetic operation on Binary numbers, Overflow & underflow, Floating Point Representation, Codes, ASCII, EBCDIC codes, Gray code, Excess-3 & BCD, Error detection & correcting codes.

### **Logic Gate:**

AND, OR, NOT gates and their Truth tables; NOR, NAND & XOR gates, Boolean Algebra, Basic Boolean Law's, Demorgan's theorem, MAP Simplification, Minimization techniques, K-Map, Sum of Product & Product of Sum.

### **Combinational & Sequential circuits:**

Half Adder & Full Adder, Full subtractor, Flip-flops - RS, D, JK & T Flip-flops, RAM and ROM, Multiplexer, Demultiplexer, Encoder & Decoder.

[Type text]

**Registers:**

Shift registers, serial-in, serial-out, parallel-in, parallel-out.

**Counters:**

Synchronous, Asynchronous, up, down counters, ring counters and ripple counters.

**I/O Interface:**

Properties of simple I/O devices and their controller, Isolated versus memory-mapped I/O, Modes of Data transfer, Synchronous &amp; Asynchronous Data transfer, Handshaking, Asynchronous serial transfer, I/O Processor.

**Auxiliary memory:**

Magnetic Drum, Disk &amp; Tape, Semi-conductor memories, Memory Hierarchy, Associative Memory, Virtual Memory, Address space &amp; Memory Space, Address Mapping, Page table, Page Replacement, Cache Memory, Hit Ratio, Mapping Techniques, Writing into Cache.

**TEXT BOOKS :**

1. Computer Fundamentals : B. Ram – New Age International Publishers
2. BARTEE : *Digital Computer Fundamentals*

**REFERENCE BOOKS:**

1. MALVINO : *Digital Compute Electronics*
2. MORRIS MANO : *Computer System Architecture*
3. MALVINO : *Digital Principles & Applications*
4. R.P. JAIN : *Modern Digital Electronics*



*Institute for Excellence in Higher Education (IEHE), Bhopal*

B.Sc. (HONOURS): **PART-I**

Semester: **I**

SUBJECT: **COMPUTER SCIENCE**

PAPER: **HONOURS-I**

TITLE: Digital Computer Organization (**PRACTICALS**)

**LIST OF PRACTICALS**

**Practical : 3 hours**

**Maximum Marks :100**

1. Design Circuit diagram for Basic Gate using NOR Gate
2. Design Circuit diagram for Basic Gate using NAND Gate
3. Design Circuit diagram for XOR and XNOR Gate using Basic Gates
4. Design Circuit diagram for XOR and XNOR Gate using NOR Gate
5. Design Circuit diagram for XOR and XNOR Gate using NAND Gate
6. Design Circuit diagram for Implement of Boolean expression
7. Design Circuit diagram for De Morgan's Theorem
8. Design Circuit diagram for Half Adder
9. Design Circuit diagram for Half Adder using Universal Gate
10. Design Circuit diagram for Full Adder
11. Design Circuit diagram for Full Adder using Universal Gate
12. Design Circuit diagram for Half Subtractor
13. Design Circuit diagram for Half Subtractor using Universal Gate
14. Design Circuit diagram for Full Subtractor
15. Design Circuit diagram for Full Subtractor using Universal Gate
16. Design Circuit diagram for Multiplexer
17. Design Circuit diagram for Multiplexer( $8 \times 1$  ,  $16 \times 1$ )
18. Design Circuit diagram for Demultiplexer
19. Design Circuit diagram for Demultiplexer( $1 \times 8$  ,  $1 \times 16$ )
20. Design Circuit diagram for Decoder
21. Design Circuit diagram for 3 to 8 Decoder
22. Design Circuit diagram for Encoder
23. Design Circuit diagram for Octal to Binary Encoder
24. Design Circuit diagram for Decimal to Binary(BCD) Encoder
25. Design Circuit diagram for Hexadecimal to Binary Encoder
26. Design Circuit diagram for Parity Bit Checker
27. Design Circuit diagram for Parity Bit Generator

Design Circuit diagram for any Gate.

20 Marks

Design Circuit diagram for Combinational & Sequential circuits

20 Marks

### Design Circuit diagram for Registers and Counters

20 Marks

### Practical Record

20 Marks

Viva

20 Marks

[illegible]



# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: **PART-I**Semester: **I**

SUBJECT: COMPUTER SCIENCE

PAPER: HONOURS-II/SUBSIDIARY

Maximum Marks: 100

Time : 3:00 Hours

## TITLE: PROGRAMMING FUNDAMENTALS USING 'C' LANGUAGE

**Objective of the course:** The course will enable the students to develop basic understanding of computer programming fundamentals. Clear understanding of the basic terminology required for programming.

**Learning Outcome:** - On completion of this course students will be able to:

- Understand various constructs of the C Language along with proper syntax.
- Develop programs on various topics.

## SYLLABUS

**Introduction:** Preliminary concept of Software, Hardware, High & Low level language; Types of Compiler and Interpreter, C Standards, Various types of IDE (Integrated Development Environment) for Programming Languages, Structured programming, Principles of good programming, Flowcharts & Algorithms.

**Overview of C Language:** History of C, Importance of C, Basic structure of C Programming style, Constant, Variables and Data types, declaration of variables, storage class, defining symbolic constants, declaring a variable as constant, volatile, overflow and underflow of data.

**Operator and expressions:** Unary, Binary and Ternary Operator. arithmetic, relational, logical, assignment operators, increment and decrement operators, conditional operators, bitwise operators, special operators, arithmetic expression, evaluation of expressions, type conversions in expression, operator precedence and associativity, mathematical functions, Managing I/O Operations (printf(), scanf(), getchar(), putchar(), getc(), getch(), gets(), puts()) Expressions. Formatting Input and Output.

**Conditional statements:** if, else if, nested if, else if ladder, switch.

**Iterative statements:** do loop, while loop, for loop, Jump statements (break, continue and goto).

**Arrays:** 1-D Array, 2-D Array & Multi- Dimensional Array, declaration, initialization,

**Strings:** declaration, initialization of string variables, reading and writing string, string manipulations(**strlen()**, **strcpy()**, **strcat()**, **strcmp()**, **strlwr()**, **strupr()**).

**User defined functions:** need, multi-function program, elements of user defined functions, definition, return values and their types, arguments and parameters, command line arguments, function declaration and calling (call by value and call by reference), nesting of functions, recursion, passing arrays, strings to functions, scope visibility and life time of variables, multi file programs.

**structure and unions:** defining a structure, declaring structure variables, accessing structure members, initialization, copying and comparing, operations on individual members, arrays of structures, arrays within structures, structure within structures, structures and functions, unions, size of structures, bit fields.

**Pointers & Memory Allocation in C:** pointers in C, declaring, initialization of pointer variable, accessing the address of variable, accessing a variable through its pointer, chain of pointers, pointer expressions, pointer increment, pointer to pointer, pointers to structures, Problems with pointers, passing pointers as function arguments, returning a pointer from a function, using arrays with pointers, Passing arrays to functions. Static and dynamic memory allocation, use of **malloc()**, **calloc()** and **free()** functions, storage of variables in static and dynamic memory allocation.

**Preprocessor directives and File Handling in C:** Understanding the Preprocessor Directives - **#include**, **#define**, **#error**, **#if**, **#else**, **#elif**, **#endif**, **#ifdef**, **#ifndef** and **#undef**, Macros. File I/O, Opening and closing a file, Reading and writing text files, Using **fput()**, **fget()**, **read()** and **write()** functions, Random access in files.

### TEXT BOOKS:

1. E Balaguruswamy, Computer Concepts and Programming in C, TataMcGraw Hill Publications
2. Brian W. Kernighan, Dennis M. Ritchie, C Programming Language 3rd Edition 2011, Pearson Publications

### REFERENCES BOOKS:

1. The Complete Reference C/C++ BY Herbert Schildt
2. Yashavant P. Kanetkar, Let Us C, BPB
3. Test Your Skills in C, Publisher-TMH by S. Thamarai Selvi & R. Murugesan
4. C Language by Amit Chaddha
5. Behrouz A. Computer Science-A Structured Programming Approach Using C.



# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: **PART-I**Semester: **I****SUBJECT: COMPUTER SCIENCE****PAPER: HONOURS-II/SUBSIDIARY****Maximum Marks: 100****Time : 3:00 Hours**

## C LAB PRACTICALS

**Required IDE for C Programming – Dev C++ / CodeBlock**

### Group - I

#### 1. Programs based on basic mathematics: -

- a) Write a C Program to print the sum and product of digits of an integer.
- b) Write a C Program to reverse a number.
- c) Write a C Program to compute the sum of the first **n** terms of the following series: -  

$$S = 1 + 1/2 + 1/3 + 1/4 + \dots + 1/(n+1)$$
- d) Write a C Program to compute the sum of the first **n** terms of the following series: -  

$$S = 1 - 2 + 3 - 4 + 5 - \dots$$
- e) Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
- f) Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
- g) Write a C Program to compute the factors of a given number.
- h) Write a macro that Write a C Programs two numbers. Write a C Program to use it.

#### 2. Write a C Program to print a triangle of stars as follows (take number of lines from user):

|       |       |           |           |       |       |       |       |
|-------|-------|-----------|-----------|-------|-------|-------|-------|
| *     | *     | 1         | 1         | 1     | 1     | 1     | 1     |
| ***   | ***   | 1 2       | 1 2       | 22    | 22    | 21    | 21    |
| ***** | ***** | 1 2 3     | 1 2 3     | 333   | 333   | 321   | 321   |
| ***** | ***** | 1 2 3 4   | 1 2 3 4   | 4444  | 4444  | 4321  | 4321  |
| ***** | ***** | 1 2 3 4 5 | 1 2 3 4 5 | 55555 | 55555 | 54321 | 54321 |

#### 3. Write a C Program to perform following actions on an array entered by the user:

- a) Print the even-valued elements
- b) Print the odd-valued elements
- c) Calculate and print the sum and average of the elements of array
- d) Print the maximum and minimum element of array
- e) Remove the duplicates from the array
- f) Print the array in reverse order

#### 4. The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.



5. Write a C Program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
6. Write a program that Write a C Programs two numbers using pointers.
7. Write a program in which a function is passed address of two variables and then alter its contents.
8. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the **main()** function.

### Group - II

9. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using **malloc()** / **calloc()** functions or new operator.
10. Write a menu driven program to perform following operations on strings:
  - a) Show address of each character in string
  - b) Concatenate two strings without using **strcat()** function.
  - c) Concatenate two strings using **strcat()** function.
  - d) Compare two strings
  - e) Calculate length of the string (use pointers)
  - f) Convert all lowercase characters to uppercase
  - g) Convert all uppercase characters to lowercase
  - h) Calculate number of vowels
  - i) Reverse the string
11. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
12. Write a C Program to display Fibonacci series (i) using recursion, (ii) using iteration.
13. Write a C Program to calculate Factorial of a number (i) using recursion, (ii) using iteration.
14. Write a C Program to calculate GCD of two numbers (i) with recursion (ii) without recursion.
15. Create Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation): a) Sum b) Difference c) Product d) Transpose.
16. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.

### Group - III

17. Write a program to retrieve the student information from file created in previous question and print it in following format:
 

| Roll No. | Name | Marks |
|----------|------|-------|
|----------|------|-------|
18. Copy the contents of one text file to another file, after removing all whitespaces.
19. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void
20. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.
21. Write a Program to Create a File & Store Information
22. Write a C Program to Illustrate Reading of Data from a File
23. Write a C Program Delete a specific Line from a Text File

24. Write a C Program to Append the Content of File at the end of another.

### Scheme of Marks

|    | Marks Distribution of Programs | MARKS |
|----|--------------------------------|-------|
| 1. | Group I                        | 20    |
| 2. | Group II                       | 20    |
| 3. | Group III                      | 20    |
| 4. | Practical Record               | 20    |
| 5. | Viva                           | 20    |

*[Handwritten signatures and marks]*



# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: **PART-I**Semester: **II****SUBJECT: COMPUTER SCIENCE****PAPER: HONOURS-I**

max. marks: 100

Time : 03 Hrs

**TITLE: COMPUTER HARDWARE MAINTENANCE & NETWORKING CONCEPTS**

**Objective of the Course:** The course will enable students to develop background knowledge as well as core expertise in Computer Hardware Maintenance. The students will learn assembling & de-assembling of a computer system, Installation of software's, Server Installation & Network configurations.

**Course Learning Outcomes:-** Upon completion of the course, students should possess the following knowledge and skills:

- Perform the maintenance of a computer.
- Install software and configure Network of computers.
- Analyze the performance of processors and caches
- Study of Formatting, Scheduled Tasks, Defragmentation, Bootable disk

## SYLLABUS

**Basic concepts of Hardware:** CPU Identification, The INTEL CPU, 8086, 80286, 386, 486, P-I, P-II, P-III, P-IV Microprocessors (types and speed), AMD CPU, Cyrix CPU, Introduction of latest desktop, 7th and 8th generation, CPU concepts, CISC vs RISC CPU, Processor speed,

Versions and steps, Processor Power & Management, System clock, Internal MPU Components & functions jumpers, Cache (L1, L2 & L3), CPU socket, CPU fan, keyboard, Display card, Ethernet cards, NIC cards, Serial & Parallel ports, SMPS connection to Mother Board, connection to peripherals, failure and replacement, Printer overview, types and connections.

**PC System Architecture & Overview:** Motherboards (slots, sockets, & connectors types), BIOS, BIOS features, BIOS setup & configurations, AMIBIOS, AWARD BIOS, MR BIOS, BIOS settings, Flash BIOS, BIOS & Boot Sequences, POST. Buses, ISA, 8 bit & 16 bit ISA, PCI, PCI layout, Bus configurations & signals, AGP, AGP similarities to PCI, AGP layout & signals, concepts of I/O addressing, interrupts & IRQ. Memory, RAM, ROM, SDRAM, DRAM, Cache memory, memory settings, adding and replacing memory modules, configuring the motherboard, Booting process, Boot strap loader program, Booting from, CD & HDD, DOS user interface.

**Mass Storage Devices:** Hard Disk overview, HDD capacity, Platters, Read/Write heads, Choosing a Disk type, SCSI, IDE and EIDE, PATA, SATA, SSD, Planning & Installations, Data backup, Drive configurations, Installation of a new drive, Replacing a new drive, Partition a Drive, Low level and High level formatting.

**Operating System:** OS features, MS-DOS, WIN-XP, WINDOWS-7, booting files, .INI files,

Approved for the session 2019-2020

Installation of Operating Systems, Dual boot; Virus & Computer Services, Antivirus software.

**Video Overview:** Resolutions & Color depth, Video card standards, Monitor components & characteristics, Video memory technologies, Video drivers, refresh rates and graphics & video acceleration, Installation a video card, TV tuner card, sound card, driver installation for hardware, jumper settings, CDROM & DVD and their specifications, Mouse, types of Mouse, MODEM installations and configurations.

**Data communication & Computer Networks:** Data transmission modes & speed, Data Transmission media, Digital & Analog Transmission, Network Topologies, Network types, Communication protocols, Installation of Windows Server NT/2008, using device manager, using hardware wizard, configuring network setting, changing network identity, configuring network components, configuring TCP/IP, Installing Printers, Installing local & network printers, Files share & permissions.

#### Text Books

1. *Bigelow's Troubleshooting Maintaining & Repairing PCs*, Tata McGraw Hill Edition, 5<sup>th</sup> Edition.
2. *Computer Fundamentals*, P.K. Sinha, Preeti Sinha, BPB Publication, 2003.

#### Reference Books

1. *Upgrading and Repairing PC*, Scott Mueller's, 22<sup>nd</sup> Edition
2. *Microsoft Windows Server 2003 Administrators Companion*, Russel, Crawford, Gerend (PHI)
3. *Microsoft Windows Server 2003 Administrators Companion*, Russel, Crawford, Gerend (PHI)

*[Signature]*

*[Signature]*

Approved for the session 2019-2020

*[Signature]*

*[Signature]*

*[Signature]*

*[Signature]*

*[Signature]*

*[Signature]*

*[Signature]*

*[Signature]*



# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: **PART-I**Semester: **II**SUBJECT: **COMPUTER SCIENCE**PAPER: **HONOURS-I**TITLE: **COMPUTER HARDWARE MAINTENANCE & NETWORKING CONCEPTS****LIST OF PRACTICAL****Maximum Marks: 100****Practical: 3 Hrs.**

1. Preparing the case.
2. Installation and troubleshooting the Motherboard.
3. Installation and troubleshooting the CPU.
4. Installation and troubleshooting the heat sink and cooling fan.
5. Installation and troubleshooting the RAM.
6. Installation and troubleshooting the SMPS to different devices.
7. Installation and troubleshooting the hard-drive and its cables.
8. Installation and troubleshooting optical drives.
9. Installation and troubleshooting the video card, sound cards, and other cards.
10. Installation and troubleshooting PCI.
11. Installation and troubleshooting Expansion cards.
12. Operating system Installation i.e Windows and dual boot.
13. Device Driver Installation.
14. Straight through and Crossover Wiring, Connector.
15. Installation and troubleshooting the Printer.
16. Files share & permissions
17. Installation of device drivers for Mouse / printer / scanner / pen drive / CD- Combo and Checking hardware and software components and working with multimedia.
18. Study of Power Supply : Types, Concepts of Fuse, MCB, Stabilizer, UPS, SMPS and checking of Different voltages, Power Good Signal
19. Study of Formatting, Scheduled Tasks, Defragmentation, Bootable disk

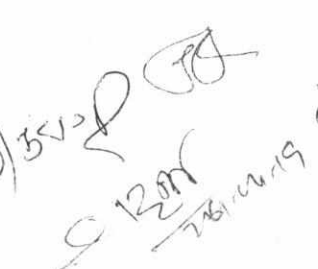
**Schema of Marks:**

- |                                                      |          |
|------------------------------------------------------|----------|
| 1. Preparing the case.                               | 20 Marks |
| 2. Operating system Installation                     | 20 Marks |
| 3. Straight through and Crossover Wiring, Connector. | 20 Marks |
| 4. Practical Record                                  | 20 Marks |
| 5. Viva                                              | 20 Marks |

Approved for the session 2019-2020








# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours : **PART-I**Semester: **II**

SUBJECT: COMPUTER SCIENCE

PAPER: HONOURS-II/SUBSIDIARY

Max. marks: 100

Time: 03 Hrs

## TITLE: Object Oriented Programming using C++

**Objective of the Course:** The course will enable students to develop basic understanding of Object Oriented Programming concepts and develop C++ language programs using OOPs.

- To understand how C++ improves C with object-oriented features.
- To learn how to write inline functions for efficiency and performance.
- To learn the syntax and semantics of the C++ programming language.
- To learn how to design C++ classes for code reuse.
- To learn how to implement copy constructors and class member functions.
- To understand the concept of data abstraction and encapsulation.
- To learn how to overload functions and operators in C++.
- To learn how containment and inheritance promote code reuse in C++.
- To learn how inheritance and virtual functions implement dynamic binding with polymorphism.
- To learn how to design and implement generic classes with C++ templates.
- To learn how to use exception handling in C++ programs.

**Learning Outcomes:** After the completion of this course, a successful student will be able to do the following:

- a) Use the characteristics of an object-oriented programming language in a program.
- b) Use the basic object-oriented design principles in computer problem solving.
- c) Use the basic principles of software engineering in managing complex software project.
- d) Program with advanced features of the C++ programming language.

Maximum Marks: 100

### SYLLABUS

1. **Principle of Object Oriented Programming:** Software Evolution, The Traditional Approach (Structured Methodology Overview, A Structured Methodology Solution, Shortcoming of Procedure Oriented Language), Object Oriented Paradigm, Basic Concepts of Object Oriented Programming, Benefits of OOP, Object oriented Language.
2. **C++ Programming Basics:** Basic Program Construction, Input/Output using cin/cout, Basic and user Defined data types, Manipulators, type Conversions, Arithmetic Operators and Library Functions.

Approved for the session 2019-20



3. **Loops, Decisions and structures:** Relational Operators, Loops: *for*, *while* and *do-while*, Decisions - *if*, *if...else*, *switch* Statements, Logical Operators, Control Statements: *break*, *continue* and *goto*, Structured and Enumerated Variables.
4. **Functions:** Simple Functions, Function Prototyping, Overloaded Functions, Inline Functions, Variable and Storage classes.
5. **Classes and Objects:** Specifying a Class, Defining Member Functions, Making an Outside Function Inline, Private Member Functions, Arrays within a Class, Array of Objects, Objects as Function Arguments, Returning Objects.
6. **Constructors and Destructors:** Constructors, Parameterized Constructors, Multiple Constructors in a Class, Constructors with Default Arguments, Dynamic Initialization of Object, Destructors.
7. **Operator Overloading:** Overloading Unary Operators (The Operator, Keyword, Operators Arguments, Operators Return Value), Overloading Binary Operators (Arithmetic's Operators, Concatenating Strings, Multiple Overloading, Comparison Operators, Arithmetic Assignments Operators), Type Conversions (Conversions between Basic Types, Conversions between Objects of different Classes).
8. **Inheritance - Extending Classes:** Derived Class and Base Class, Public and Private Inheritance, Levels of Inheritance, Multiple Inheritance (Multiple Functions in Multiple Inheritance, Constructors in Multiple Inheritance).
9. **Pointers:** Address and Pointers (The Address of Operators and Pointer Variables. Accessing the Variables Pointed to, Pointer to void), Pointers and Arrays (Pointer Constants and Pointer Variables), Pointers and Functions, Memory Management: New and Delete Operators, Pointers to Objects.
10. **Virtual Functions and Polymorphism:** Virtual Functions (Accessing of Normal Members Function with Pointers, Accessing of virtual Members Functions with Pointers, Pure Virtual Functions, Abstract Classes, Virtual Base Classes)
11. **Files and Streams:** Streams (Streams classes, Header files), String I/O, Character I/O, Object I/O (Writing/Reading an object to/from Disk, Binary Versus Character files, Compatible Data Structures).

**TEXT BOOK**

1. Object-Oriented programming with C++ by E. Balagurusamy, Tata Mc-Graw Hill

**REFERENCES BOOKS**

1. Object-Oriented programming in Turbo C++ Robert Lafore, Galgotia Publications Pvt. Limited
2. C++ Primer by Stanley B. Lippman, Addison Wesley Professional
3. Object-Oriented programming in C++ by Nabajyoti Barkakati
4. The Complete Reference C/C++ by Herbert Schildt,
5. C++ by Schumn's Series, Tata Mc-Graw Hill



# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours : **PART-I**Semester: **II**

SUBJECT: COMPUTER SCIENCE

PAPER: HONOURS-II/SUBSIDIARY

Max. Marks: 100

Time : 03 Hrs

**TITLE: Object Oriented Programming using C++ Practical Syllabus**

## 1) Some basic Programs :

- a. Write a program for finding greatest number to any three value Sum of first ten natural numbers
- b. Calculate the marks of five subjects and also find percentage.
- c. Make a program of swapping a variable without using third variable.
- d. Display table of any number.
- e. Program that displays the size of every data type.
- f. Write a program to showing bitwise operators.
- g. Demonstrate the use of assignment operator.
- h. Write a program for finding vowel by using switch statement.
- i. Similar other programs

## 2) C++ Programs (Based on Control statement/looping/ Decisions/Structures):

- a. Program for explaining difference between do...while and while statement.
- b. Entered number is odd or even by using *if ... else*
- c. The greatest among three numbers by using *if* statement.
- d. Calculate sum of any numbers by using *for* loop
- e. Sum of natural numbers by using *while* loop.
- f. Demonstrate the use of nested *if .. else* statement.
- g. Demonstrate the use of switch case statement.
- h. Program to enter a number from 1-7 and display the corresponding day of the week using switch case statement.
- i. Design a program for explaining control statement of "break".
- j. Design a program for explaining control statement of "goto".
- k. Design a program for explaining control statement of "continue".
- l. Similar other programs

## 3) C++ Program based on Functions:

- a. Write a function to Design a calculator program by using operators calculating simple and compound interest.
- b. Design a program of overloading function for addition and subtraction of numbers.
- c. Write a program for explaining inside a function.
- d. Similar other programs

Approved for the session 2019-20

**4) C++ Programs based on Classes and Object.**

- a. Program for explaining outside a function.
- b. Program for explaining inside a function.
- c. Design a program for accessing a private member functions.
- d. Design a program for accessing a private member variables.
- e. Program for explaining Array within a class.
- f. Program for explaining Array of class
- g. Program for returning Objects.
- h. Similar other programs

**5) C++ Programs based on Constructors and Destructors.**

- a. Design a program for defining constructor
- b. Program for Defining Multiple Constructors.
- c. Program for Defining Overloading a Constructor.
- d. Design a program of explaining Destructor
- e. Program for Dynamic initialization of Constructor of Long terms Fixed Deposit System.
- f. Similar other programs.

**6) C++ Programs based on Operator Overloading.**

- a. Program of Overloading Unary Operators.
- b. Design a program of Overloading Binary operator.
- c. Design a program of Overloading Binary operator.
- d. Write a program for explaining type conversion.
- e. Similar other programs

**7) C++ Programs based on Inheritance.**

- a. Design a program of single inheritance.
- b. Design a program of three level inheritances.
- c. Design a program of multiple inheritances.
- d. Design a program of Hybrid inheritances.
- e. Design a program of constructor in multiple inheritances.
- f. Design a program of multiple functions in multiple inheritance inheritances.
- g. Similar other programs

**8) C++ Programs Pointers.**

- a. Design a program for Explaining pointer.
- b. Program to display read and display by using pointer with array.
- c. Program to display read and display by using array of pointers.
- d. Write a program for explaining pointer to function..
- e. Design a program for explaining pointers to objects.
- f. Design a one program of memory management.
- g. Similar other programs

**9) C++ Program based on Virtual Functions and Polymorphism.**

- a. Design a program for explaining Virtual function.



- b. Design a program by using Virtual function of Hybrid inheritance.
- c. Design a program for explaining Run time Polymorphism.
- d. Similar other programs

#### 10) C++ Programs based on Files and Streams :

- a. Write a Program for defining unformatted I/O operations.
- b. Write a Program for reading strings with getline().
- c. Write a program for displaying string with write().
- d. Design a program on Working with some file operations.

#### Scheme of Marks :

|   |                                                                                                                                             |          |
|---|---------------------------------------------------------------------------------------------------------------------------------------------|----------|
| 1 | Some basic Programs/ C++ Programs (Based on Control statement/looping/ Decisions/Structures)/ C++ Program based on Functions                | 20 Marks |
| 2 | C++ Programs based on Classes and Object./ C++ Programs based on Constructors and Destructors./ C++ Programs based on Operator Overloading. | 20 Marks |
| 3 | C++ Programs based on Inheritance./ C++ Programs Pointers./ C++ Program based on Virtual Functions and Polymorphism/ Files and Streams      | 20 Marks |
| 4 | Practical Record                                                                                                                            | 20 Marks |
| 5 | Viva                                                                                                                                        | 20 Marks |

# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: **PART-II**Semester: **III**SUBJECT: **COMPUTER SCIENCE**PAPER: **HONOURS - I**

## TITLE: DATA STRUCTURE THROUGH C++

Maximum Marks: **100**Time: **3:00 Hours**

### Course Objective:

1. The objective of this course is to make the student learn fundamental data structures algorithms.
2. The course describes and implements algorithms such as stacks, queues, linked lists, trees, searching techniques, sorting techniques, hashing techniques and graphs.
3. Comprehend alternative implementations using the differing logical relationships and appreciate the significance of choosing a particular logical relationship for implementation within real-world setting.
4. Demonstrate the ability to plan, design, execute and document sophisticated technical programs to handle various sorts of data structures.

**Learning Outcome:** Having successfully completed this course, the student will be able to:

1. Apply advance C programming techniques such as pointers, dynamic memory allocation, structures to developing solutions for particular problems.
2. Design and implement abstract data types such as linked list, stack, queue and tree by using C as the programming language using static or dynamic implementations.
3. Analyse, evaluate and choose appropriate abstract data types and algorithms to solve particular problems.

## SYLLABUS

**Introduction:** Introduction to Data Structures: Definition of Data structure and Abstract data type Classification of Data structures: Linear, Non-linear, homogeneous, non-homogeneous, static & dynamic.

**Complexity:** Advantages and Disadvantages. Big O Notation, Big-omega and Big-theta Notations, Growth of Functions.

**Arrays:** Types, Memory Representation, Address Translation, Functions of single and multi-dimensional arrays with examples.

**Traversal Sparse Matrix:** Definition & memory representation.



**Linked Structures:** Singly and doubly linked list (non-circular and circular), List manipulation with pointers: Searching, Insertion and deletion of elements.

**Stack:** Definition, Array implementation of stack (static stack): Operations PUSH, POP, TRAVERSE. Applications of stack: Infix, Prefix, Postfix representation and evaluation using stack, Use of stack in recursive implementation.

**Queue:** Definition, Array implementation of queue (static queue): Operations INSERT, DELETE, TRAVERSE.

Introduction to Circular queue: Definition & implementation, Priority queue, Double ended queue

Applications of queue

**Recursion:** Divide and Conquer, elimination of recursion.

**Trees:** Introduction, Quantitative Properties, Binary Tree, Tree traversals, Internal and external path lengths: Properties, Minimum and maximum path length of a binary tree, Importance.

**Binary Search Trees:** Introduction, Searching, Insertion, Deletion.

**Searching:** Linear and binary search, performance and complexity.

**Hashing:** Concepts, Advantages and Disadvantages, Different types of hash functions, Collision and Collision Resolution Techniques – Open Addressing with probing, Linear Chaining, Coalesced Chaining, Application.

**Sorting:** Terminology, Performance Evaluation, Different Sorting Techniques (Bubble, Insertion, Selection, Quick sort, Merge Sort, Heap, Partition Exchange, Radix with iterative and recursive description), Complexity, advantages and disadvantages.

**Introduction to graphs:** Definition, Terminology, Directed, Undirected & Weighted Graph, Representation of Graphs, Graph Traversal-Depth First & Breadth First Search. Spanning Tree, Minimum Spanning Tree, Shortest Path Algorithm.

### TEXT BOOKS:

1. Yddish Langsam, Moshe J. Augenstein and Aaron M. Tanenbaum, Data Structures Using C and C++, 2<sup>nd</sup> Edition, Prentice Hall of India,
2. M. Tennenbaum, Y. Langsam and M. J. Augenstein, Data Structures using C, PHI.
3. S. Lipschutz, Data Structure, Schaum's Outline Series.
4. G.S.Baluja, "Data Structures Through C", 4th Edition, Dhanpat Rai & Co.

### REFERENCES BOOKS:

1. Sahani, Data Structures, Algorithms and Applications with C++, Mc-GrawHill.

2. Nnarasimha Karumanchi, Data Structures and Algorithms made easy 5<sup>th</sup> Edition, CareerMonk Publications
3. Jean Paul Trembley and Paul G. Sorenson, An Introduction to Data Structures with applicationsI, McGraw Hill.



***Institute for Excellence in Higher Education (IEHE), Bhopal***

B.Sc. Honours: **PART-II**Semester: **III****SUBJECT: COMPUTER SCIENCE****PAPER: HONOURS - I**

**TITLE: COMPUTER SCIENCE LAB: Data Structures Lab**  
**Maximum Marks: 100** **Time: 3:00 Hours**

**Required IDE for Data Structure Programming: Dev C++/ CodeBlock**

**Part I: -**

- Write a program for insertion, deletion and traversal of elements of an array.
- Write a Program to convert the Sparse Matrix into non-zero form and vice-versa
- Write a program to search an element from a list. Give user the option to perform Linear or Binary search.
- Write a Program to implement Lower Triangular Matrix using one-dimensional array.
- Write a Program to implement Upper Triangular Matrix using one-dimensional array.
- Write a Program to implement Symmetric Matrix using one-dimensional array

**Part II: -**

- Implement Linked List. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).
- Implement Doubly Linked List. Include functions for insertion, deletion and search of a number, reverse the list.
- Implement Circular Linked List. Include functions for insertion, deletion and search of a number, reverse the list.
- Perform Stack operations using Linked List implementation.

**Part III: -**

- Write a Program to reverse the order of the elements in the stack using additional stack.
- Perform Stack operations using Array implementation.

- c) Perform Queues operations using Circular Array implementation.
- d) Write a Program to reverse the order of the elements in the stack using additional Queue.
- e) Create and perform different operations on Double-ended Queues using Linked List implementation.

**Part IV: -**

- a) Write a Program to scan a polynomial using linked list and add two polynomials.
- b) Write a Program to calculate factorial and to compute the factors of a given no. (i) using recursion, (ii) using iteration
- c) Write a Program to display Fibonacci series (i) using recursion, (ii) using iteration
- d) Write a Program to create a Binary Search Tree and include following operations in tree:
  - 1) Insertion (Recursive and Iterative Implementation)
  - 2) Deletion by copying
  - 3) Deletion by Merging
  - 4) Search a no. in BST
  - 5) Display its preorder, postorder and inorder traversals Recursively
  - 6) Display its preorder, postorder and inorder traversals Iteratively
  - 7) Display its level-by-level traversals
  - 8) Count the non-leaf nodes and leaf nodes
  - 9) Display height of tree
  - 10) Create a mirror image of tree
  - 11) Check whether two BSTs are equal or not

**Part V: -**

- a) Write a program for Implement bubble sort.
- b) Write a program for Implement selection sort.
- c) Write a program for Implement insertion sort.
- d) Write a program for Implement merge sort.
- e) Write a program for Implement heap sort.
- f) Write a Program to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort, Selection sort, Merge sort, Heap sort.



- g) Write a Program to create a Threaded Binary Tree as per inorder traversal, and implement operations like finding the successor / predecessor of an element, insert an element, inorder traversal

Scheme of Internal Marks Distribution:

| S.N. | Topic            | Marks    |
|------|------------------|----------|
| 1.   | PART - I / V     | 20 marks |
| 2.   | PART - II / III  | 20 marks |
| 3.   | PART - IV        | 20 marks |
| 4.   | PRACTICAL RECORD | 20 marks |
| 5.   | VIVA             | 20 marks |

# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: **PART-II**Semester: **III**SUBJECT: **COMPUTER SCIENCE**PAPER: **HONOURS-II/SUBSIDIARY**

Max. Marks: 100

Time: 03 Hrs

TITLE: **Database Management System & Design****Objective of the course:**

The course will enable the students to develop basic understanding of following things:

- 1 To describe a sound introduction to the discipline of database management systems.
- 2 To give a good formal foundation on the relational model of data and usage of Relational Algebra.
- 3 To introduce the concepts of basic SQL as a universal Database language.
- 4 To enhance knowledge to advanced SQL topics like embedded SQL, procedures connectivity through JDBC.
- 5 To demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization.
- 6 To provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques.

**Learning Outcomes:**

At the end of the course the students will able to:

- 1 Explain the features of database management systems and Relational database.
- 2 Design conceptual models of a database using ER modelling for real life applications and construct queries in Relational Algebra.
- 3 Create and populate a RDBMS for a real-life application, with constraints and keys, using SQL.
- 4 Retrieve any type of information from a data base by formulating complex queries in SQL.
- 5 Analyze the existing design of a database schema and apply concepts of normalization to design an optimal database.
- 6 Build indexing mechanisms for efficient retrieval of information from a database.

**Maximum Marks: 100****SYLLABUS****PART-(A): DATABASES & THEIR CONTEXT****1. Database Systems and the Evolution of Database Technology**

The Evolution of Database Technology, File-Oriented Systems, Database Systems: Hierarchical and Network Model Systems, Relational Database Systems; Database Systems: Hardware, Software, Data & People.



## 2. Database Systems in the Organization

Data Sharing - Between functional units, between different levels of users and between different locations. Strategic Database Planning – Need, Database Planning Project, Database Development Life Cycle. Databases and Management Control – Database Design, Database Security & Integrity. Risk and Costs of Databases – Organizational Conflict, Development Project Failure, System Failure, Overhead Cost. Logical & Physical Data Representation – Three-Level Database Architecture. Database Development – Database Design and the traditional System Development Life Cycle (SDLC), The Database Development Life Cycle (DDLCC), Preliminary Planning, Feasibility Study, Requirement Definition, Conceptual Design, Implementation

### PART-(B): DATABASE DESIGN

## 3. Principles of Conceptual Database Design

Reality, Requirements Definition and Conceptual Data Modeling- Reality & Models, Conceptual Data Models. Fundamentals – Objects, Specialization & Generalization, Relationships, Cardinality, Attributes & their examples. Aggregation & their examples, View Integration: An example.

## 4. The Relational Data Model

*Fundamental Concepts* - Relations, Null Values, Keys, Foreign Keys; Integrity Constraints: Entity Integrity & Relational Integrity.

*Normalization Process* – First Normal Form, Functional Dependencies, Second Normal Form, Third Normal Form, Fourth Normal Form; Other Normal Form: Fifth Normal Form & Domain/Key Normal Form.

*Transforming a Conceptual Model to a Relational Model* – Transforming Objects Sets and Attributes, Transforming Models without External Keys, Transforming Specialization and Generalization, Object Sets; Transforming Relationships: One-One Relationships, One-Many Relationships, Many-Many Relationships; Transforming Aggregated Object Sets, Transforming Recursive Relationships, Transformation examples.

*Comparison of Conceptual and Relational Data Modeling.*

### PART-(C): RELATIONAL DATABASE IMPLEMENTATION

## 5. Relational Algebra and Calculus

*Relational Algebra*: Union, Intersection, Difference, Product, Select, Project, Join - Natural, Theta & Outer Join, Divide, Assignment.

*Relational Calculus*: Target list & Qualifying Statement, The Existential Quantifier, The Universal Quantifier.

## 6. Relational Implementation with SQL

*Relational Implementations an Overview.*

*Schema and table Definition* – Schema and Table Definition, Data Types & Domain, Defining Tables, Column Definition.

*Data Manipulation*: Simple Queries (SELECT, FROM, WHERE); Multiple-Table Queries, Subqueries, Correlated Subqueries; EXISTS and NOT EXISTS, Built-In Functions, GROUP BY and HAVING clause Built-In Functions with Subqueries.

*Relational Algebra Operations* – UNION, INTERSECT, EXCEPT, JOIN.

*Database Change Operations* – INSERT, UPDATE, DELETE.

*Using SQL with Data Processing Languages*; View Definition, Information Schema.

## 7. Physical Database Systems

Introduction, Physical Access of the Database.

*Physical Storage Media*: Secondary Storage, Physical Storage Blocks.

*Disk Performance Factors*: Access Motion Time, Head Activation Time, Rotational Delay, Data Transfer Rate, Data Transfer Time.

*Data Storage Formats on Disk*: Track Format, Record Format, Input/output Management.

*File Organizing and Addressing Methods*: Sequential File Organization, Indexed-Sequential File Organization, Direct File Organization, Hashing- Static Hash Functions and Dynamic Hash Functions;

**Indexing: Basic concept of Indexing, Ordered Indexing- Primary Index and Secondary Index, Dense Index, Sparse Index.** Implementing Logical Relationships: Linked Lists, Inverted Lists, Balanced-Tree Index (B<sup>+</sup>-Tree Index Files).

## TEXT BOOK

1. Database Management and Design 2<sup>nd</sup> edition by Gary W. Hansen & James V. Hansen, Prentice Hall.

## REFERENCE BOOKS

1. An Introduction to Database System by Bipin C. Desai, Galgotia publications
2. An Introduction to Database System 8<sup>th</sup> Edition by C.J. Date
3. Principle of Database Management System by James Martin, Prentice Hall
4. Database System Concepts 6<sup>th</sup> edition by Henry Korth Abraham. Silbersetz, Tata McGraw Hills
5. Database Management Systems by ISRD Group, Tata McGraw Hills



## *Institute for Excellence in Higher Education (IEHE), Bhopal*

B.Sc. Honours: **PART-II**Semester: **III**

Max. Marks: 100

Time: 03 Hours

### TITLE: Database Management System & Design Practical Syllabus

1. Create a table using table wizard.
2. Create table using CREATE command.
3. Write a query for the use of USE command.
4. Write a query for the use of BROWSE command
5. Write a query for the use of DISPLAY command
6. Write a query for the use of LIST command
7. Write a query for the use of INSERT command
8. Write a query for the use of EDIT/CHANGE command
9. Write a query for the use of CLEAR command
10. Write a query for the use of APPEND command
11. Write a query for the use of DELETE command
12. Write a query for the use of PACK command
13. Write a query for the use of REPLACE command
14. Write a query for the use of GOTO and SKIP command
15. Write a query for the use of MODIFY STRUCTURE command
16. Write a query for the use of COPY command
17. Write a query for the use of COPY STRUCTURE command
18. Write a query for the use of SORT command
19. Write a query for the use of TRIM, LTRIM, RTRIM and ALLTRIM command.
20. Creating tables Worker.dbf, Assignment.dbf, Building.dbf, and Skill.dbf
21. Using '?' operator to perform arithmetic operation.
22. Simple queries using SELECT, FROM and WHERE clause
23. Multiple table queries using SELECT command.
24. Write Sub queries using SELECT command.
25. Use of built in functions in queries (SUM, MAX, MIN, COUNT, AVG)
26. Write queries using GROUP BY and HAVING clause
27. Create a report in FoxPro using report wizard.
28. Create a form in FoxPro using form wizard.
29. Create indexes in FoxPro.
30. Create a relationship using SET RELATION command between two tables.



**Scheme of Marks:**

|   |                                                                       |          |
|---|-----------------------------------------------------------------------|----------|
| 1 | Query based on simple commands/ multiple table queries                | 20 Marks |
| 2 | Report creation using report wizard and Index creation                | 20 Marks |
| 3 | Form creation using form wizard/ Relationship creation between tables | 20 Marks |
| 4 | Practical Record                                                      | 20 Marks |
| 5 | Viva                                                                  | 20 Marks |

*Rky...*

*Agg*

*A*

*Prag*

*Ancilla*

*AS*

*AS*

*AS*

*AS*

## Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: **PART-II**Semester: **IV****SUBJECT: COMPUTER SCIENCE****PAPER: HONOURS-I**

max. marks: 100

Time: 03 Hrs

### TITLE: OPERATING SYSTEMS

**Objective of the course:** The course will enable the students to develop basic understanding of operating systems such as:

1. To understand the main components of an OS & their functions.
2. To study the process management and scheduling.
3. To understand various issues in Inter Process Communication (IPC) and the role of OS in IPC.
4. To understand the concepts and implementation Memory management policies and virtual memory.
5. To understand the working of an OS as a resource manager, file system manager, process manager, memory manager and I/O manager and methods used to implement the different parts of OS
6. To study the need for special purpose operating system with the advent of new emerging technologies

#### Outcomes:

At the end of the course students will able to:

1. Describe the important computer system resources and the role of operating system in their management policies and algorithms.
2. Understand the process management policies and scheduling of processes by CPU
3. Evaluate the requirement for process synchronization and coordination handled by operating system
4. Describe and analyze the memory management and its allocation policies.
5. Identify use and evaluate the storage management policies with respect to different storage management technologies.
6. Identify the need to create the special purpose operating system.

**Maximum Marks: 100**

### SYLLABUS

**An Overview:** Definition, Operating system Concepts, Evolution of operating system, Goals of an Operating System, Types of Operating Systems, Functions of OS

**Processes:** Process Concept, Process Scheduling, Scheduling Criteria, Scheduling (Preemptive, non preemptive scheduling), Scheduling Algorithms – First Come First Serve (FCFS), Shortest Job First (SJF), Round Robin (RR), Shortest remaining time next (SRTN), Priority Based Scheduling, Multilevel Queue Scheduling (MLQ), Interprocess communication, Interprocess Synchronization, Mutual exclusion, Critical Section Problem, Semaphores.

**Deadlocks:** Deadlocks definition, Methods for handling Deadlocks, Deadlock Prevention, Deadlocks Avoidance, Deadlock Detection and Recovery from deadlocks.



**Memory Management:** Introduction Logical versus Physical Address Space, Swapping, Memory management, Contiguous Allocation Methods, Non-contiguous allocation methods, partitioned memory allocation, Fragmentation, Paging, Segmentation, Virtual Memory, Demand Paging, Page Replacement and Algorithms – FIFO, Optimal and LRU.

## LINUX

**Introduction:** Basic Features of Linux Operating System, Advantage of using Linux, Installing requirement & Installation of Linux System, System Shutdown.

Basic Architecture of Linux, Linux System, Kernel, Shell, Linux File System, Managing the file System, Understanding the File and Directory System, Linux Standard Directories.

**Linux Commands, Utilities and Editors:** Linux commands for files and directories cd, ls, cp, md, mkdir, more, less, creating and viewing files using cat, file comparisons. Understanding shells, understanding processes, connecting processes with pipes, Redirecting input output. Background processing, Managing multiple processes, kill, who, sleep, printing commands, grep, fgrep, find, sort, vi editor, user to user communication, write, finger etc.

**Shell Programming in Linux:** Basics of Shell Programming, Various types of shell, Conditional & looping statements, case statements, parameter passing and arguments.

## WINDOWS 2008

**Introduction:** Windows 2008 Operating System Structure, Peer to Peer to Network, Domains, Network Protocols, File Services, Print Services, Using Mapped Drive, Few Important facts about Windows 2008 usages.

**Managing Windows 2008 Server:** Using Windows 2008 Server and Client, Logging on to the Network, Browsing Network Resources, Accessing Network Resources using MY Network Places, Mapping a folder.

**Windows 2008 Networking:** Windows 2008 domains, workgroups & trusted relationships, concepts of Domains, trust relationships, building domains, user administration and Remote access.

**TEXT BOOKS:**

1. Operating System Concepts by Silberchatz Galwin Gagne, Publisher: Wiley, July-2008
2. Operating Systems: Concepts and Design by Milan Milenkovic, McGraw Hill Higher Education

**REFERENCES BOOKS:**

1. Sams Teach Yourself Linux in 24 Hours by William Ball, Sams; 1st Edition
2. Red Hat Linux: Study Guide by Vijay Shekhar, Laxmi Publications; First Edition
3. Windows Server 2008: The Complete Reference

*Raj...*



***Institute for Excellence in Higher Education (IEHE), Bhopal***B.Sc. Honours: **PART-II**Semester: **IV****SUBJECT: COMPUTER SCIENCE****PAPER: HONOURS-I**

Max. Marks: 100

Time : 03 Hrs

**TITLE: OPERATING SYSTEMS Practical Syllabus****1 Linux File Handling Utilities and Process Utilities commands:**

- Mkdir, Rmdir/ Rm, Cat ,Ls ,Mv, Cp
- Touch, Chmod, Chown, Ps, Kill, Sleep
- Who/ Who am i

**2 Text Processing and Backup Utilities Commands:**

- Head, Tail, Cut, Paste, Sort
- Wc, Uniq, Cmp, Diff, Comm
- Grep, Egrep, Fgrep

Similar other useful utilities commands like:

- i. Ln
- ii. Date
- iii. Cal
- iv. Alias
- v. which

**3 Shell Script Programs:**

- a. Write a shell script program to print a message "Hello World"
- b. Write a shell script program to display a date in "MM/DD/YYYY" format.
- c. Write a shell script program to see current date, time, username, and current directory.
- d. Write a shell script program for addition of two numbers.
- e. Write a shell script program for multiplication of two numbers.
- f. Write a shell script program for swapping of two numbers using third variable.
- g. Write a shell script program to check whether a number is even or not
- h. Write a shell script program to for printing a table.
- i. Write a shell script program for creating calculator using switch case write a shell script program to
- j. Write a shell script program to check which number is greater.
- k. Write a shell script program to calculate a factorial of a given number.

**Scheme of Marks:**

|                                                                |          |
|----------------------------------------------------------------|----------|
| 1 Linux file handling utilities and process utilities commands | 20 Marks |
| 2 Text Processing and Backup Utilities Commands:               | 20 Marks |
| 3 Shell script programs                                        | 20 Marks |
| 4 Practical Record                                             | 20 Marks |
| 5 Viva                                                         | 20 Marks |

Rity...

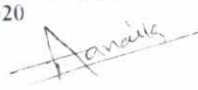


Approved for the session 2019-2020

Page 5 of 5

A















# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. (Honours): **PART-II**Semester: **IV**SUBJECT: **COMPUTER SCIENCE**PAPER: **HONOURS-II/Subsidiary**

Max. Marks: 100

Time: 03 Hrs.

**TITLE: ORACLE**

## Objectives

The objective of this course is to provide a strong formal foundation in database concepts implementation, technology and practice to become a well informed database application developer.

## Learning Outcomes

- Understand the underlying concepts of database technologies.
- Populate and query a database using SQL DDL/DML commands.
- Implement Security by applying User Management.
- Declare and enforce integrity constraints on a database.
- Enable to perform PL/SQL programming.

## SYLLABUS

**Maximum Marks: 100**

### Introduction:

**Interactive SQL:** DBMS components (overview): Normalization (1,2,3<sup>rd</sup>): EF CODD'S Rules, ER diagrams Data Integrity: Security Backup Recovery. Concept of SQL \* Loader. Invoking SQL \* plus.

**The Oracle Data Type,** Two Dimension Matrix creation. Insertion of Data into Tables. Updating the contents of a table. Deletion operations. Modifying the structure of Tables. Removing / Deleting / Dropping Tables. Data Constraints.

**The many faces of the select command.** computations in Expressions. List used to select Data, Logical operators. Range Searching, Pattern matching. Oracle Functions, Grouping, Data from Tables in SQL. Manipulating Dates in SQL, Joins.

**Constructing an English Sentence with Data from Table columns.**

**Sub queries,** using the union, intersect and minus clause indexes, views, sequences Granting Permissions, Revoking the Permissions given.



**PL/SQL:** Introduction to the PL/SQL Execution Environment the PL/SQL syntax, understanding the PL/SQL Block Structure, Error Handling in PL/SQL.

**Oracle Transactions, ACID properties,** Concurrency control in Oracle, locks, cursors,

**Stored Procedures:** Procedures: Declarative part, Executable part, and Exception Handling part. Where do procedures reside. Creating a Procedure. Executing procedures, Advantages of procedures. Syntax for creating a stored procedure.

**Stored Functions:** Functions: where do functions reside, creating a function, executing a function, Advantages of functions. Syntax for creating a stored function. Deleting a stored function.

**DataBase Triggers:** Introduction, use of Triggers, database Triggers, Types of Triggers. Syntax for creating and deleting a trigger.

**DataBase Packages:** Introduction, use of Packages, Syntax for creating and deleting a Packages.

**Object oriented databases:** concept and queries of Nested Table, Varying Arrays.

### TEXT BOOK

1. The Programming Language of Oracle 6<sup>th</sup> Revised Edition by Ivan Bayross, BPB Publication
2. Oracle Database 12c –PL/SQL Programming by Michael McLaughlin, Oracle Publication

### REFERENCE BOOK

3. Oracle Database 12C: The Complete Reference 1<sup>st</sup> Edition by Bob Bryla, McGraw Hill Publication
4. Easy SQL Programming & DB Management for Beginners by Bill Pribye , BPB Publication

***Institute for Excellence in Higher Education (IEHE), Bhopal***

**B.Sc. (Honours): PART-II**

**Semester: IV**

**SUBJECT: COMPUTER SCIENCE**

**PAPER: HONOURS-II/Subsidiary**

**TITLE: ORACLE Practical Syllabus**

**1. Create following tables using DDL - CREATE command:**

- i) CLIENT\_MASTER(Clientno , Name, City, Pincode, State, Baldue)
- ii) PRODUCT\_MASTER(Productno, Description, ProfitPercent, Unitmeasure, Qtyonhold, Reorderlvl, Sellprice, Costprice)
- iii) SALESMAN\_MASTER(Salesmanno, Salesmanname, Address1, Address2, City, Pincode, State, Salamt, Tgtoget, Ytdsales, Remarks)
- iv) SALES\_ORDER(Orderno, Clientno, Orderdate, Delyaddr, Salesmanno, Delydate, Billyn, Delydate, Orderstatus)
- v) SALES\_ORDER\_DETAILS(Orderno, Productno, Qtyordered, Qtydisp, Productrate)

**2. Populate data into above mentioned tables using DML command.**

**3. Creating tables using constraints- PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK CONSTRAINT**

- 4. Queries based on retrieving records from tables -using SELECT command.
- 5. Queries based on updating records from tables – using UPDATE command.
- 6. Queries based on deleting records from tables- using DELETE command.
- 7. Queries based on deleting table structure- using DROP command.
- 8. Queries based on altering table structure - using ALTER TABLE command.
- 9. Queries based on adding constraint, deleting constraint via ALTER TABLE command.
- 10. Queries based on renaming table- using RENAME command.

11. Queries based on arithmetic and logical operators.
  12. Queries based on LIKE, BETWEEN, IN/ NOT IN PREDICATES.
  13. Queries based on use of DUAL table.
  14. Queries based on oracle Aggregate Functions- AVG, MIN, COUNT, MAX, SUM.
  15. Queries based on use of Numeric Functions – ABS, POWER, ROUND, SQRT, EXP, GREATEST, LEAST, MOD, TRUNC, FLOOR, CEIL.
  16. Queries based on use of String Functions – LOWER, INITCAP, UPPER, SUBSTR, ASCII, COMPOSE, DECOMPOSE, INSTR, LENGTH, LTRIM, RTRIM, TRIM, VSIZE.
  17. Queries based on use of Conversion Functions – TO\_NUMBER, TO\_CHAR.
  18. Queries based on use of Date Functions – TO\_DATE, ADD\_MONTHS, LAST\_DAY, MONTHS\_BETWEEN, NEXT\_DAY, ROUND.
  19. Queries based on GROUP BY, HAVING, ORDERBY clause.
  20. Queries based on SUBQUERIES- Correlated/ Non Correlated sub query.
  21. Queries based on JOINS and Nested tables.
  22. Write SQL statement for creating different types of indexes.
  23. Write SQL statement for creating views.
  24. Write SQL statement for creating sequences.
- Program based on PL/SQL:**
25. Using GRANT and REVOKE command for granting permission.
  26. Using transaction control commands COMMIT, ROLLBACK, SAVEPOINT.
  27. Write PL/SQL program to find sum of two numbers.
  28. Write PL/SQL program using IF- THEN- ELSEIF- ELSE- ENDIF statement.
  29. Write PL/SQL program using SIMPLE LOOP, WHILE LOOP, FOR LOOP Statements.
  30. Write PL/SQL program for creating explicit cursor.
  31. Write PL/SQL program for creating triggers.



32. Write PL/SQL program for creating Procedures.
33. Write PL/SQL program for creating Functions.

**Schema of Practical marks:**

|                                   |          |
|-----------------------------------|----------|
| I. Queries based on DDL commands. | 20 Marks |
| II. Queries based on DML commands | 20 Marks |
| III. Creating PL/SQL program      | 20 Marks |
| IV. Practical Record              | 20 Marks |
| V. Viva                           | 20Marks  |

# Institute for Excellence in Higher Education (IEHE), Bhopal

**B.Sc. Honours: PART-III**

**Semester: V**

**SUBJECT: COMPUTER SCIENCE**  
**Maximum Marks: 100**

**PAPER: HONOURS-I**  
 Time: 03 hrs

## TITLE: ASP.NET WITH VC#.NET

**Objective of the Course:** This course is designed to provide the knowledge of Dot Net Frameworks along with ASP.NET and C#. The objectives of this Course are:

- To learn .Net Framework & understand .Net with C#.
- To develop web application with ASP.NET
- To develop database applications using ADO.NET
- Use web services to create distributed applications
- Develop simple and complex applications using .Net Framework

### Learning Outcomes:

At the end of the course the student will be able to -

- Create a Website with server controls.
- Separate page code from content by using code-behind pages, page controls, and components
- Display dynamic data from a data source by using Microsoft ADO.NET and data binding.

## SYLLABUS

**Visual C#.NET:** Working with variables, operators and expressions, Writing Methods and applying Scope, Using Decision Statements, Iteration statements, Managing Errors and Exceptions, managing classes and objects, working with inheritance, delegates and events, Introduction to generics, introduction to windows forms, menu and dialog boxes.

**Introduction to Microsoft Visual Studio.NET:** Overview of .NET framework, understanding ASP.NET application and controls, concept of Internet Information services, ASP.NET coding model, steps to deployment, ASP.NET event model, auto post back, view states, details about Page class, **Server Controls:** HTML server control, web controls, list controls, input validation control, state management: view, session and application state.

Approved for session 2019-2020

Page 1 of 3

**ADO.NET Architecture:** Connection, Command and Data Reader classes, Transactions, connected and disconnected data, Data Set and Data Adapter classes, Data binding, Grid View: Formatting, row selection, sorting, paging, templates, Details view, Form View.

**Introduction to XML:** reading and writing datasets to XML, working with file system, reading and writing with file streams, uploading files and compression, user controls, themes, menu, tree view and master pages.

**ASP.NET Security Model:** level of security, IIS level security, security architecture, form authentication and authorization, windows authentication, roles, concept of profiles, core JavaScript, overview of web services.

**Text Book:**

1. ASP.NET 4.0 Projects Black Book Dreamtech Press : ASP.NET 4.0

**Reference Books:**

- |                                              |                          |
|----------------------------------------------|--------------------------|
| 1. Matthew MacDonald and Mario Szpuszta :    | ASP.NET 2.0 in C# 2005   |
| 2. J.G.R Sathiaselalan and N. Sasikaladevi : | Programming with C# .Net |
| 3. Ramesh Bangia :                           | ASP.NET and C#.Net       |

Approved for session 2019-2020

Page 2 of 3



**Institute for Excellence in Higher Education (IEHE), Bhopal**

**B.Sc. Honours: PART-III**

**Semester: V**

**TITLE: ASP.NET WITH VC# .NET PRACTICAL SYLLABUS**

**Time of practical: 3:00 hrs**

**Max. Marks: 100**

- 1. C# program on Fundamental Mathematical Operations:**
  - a. Program to check whether the entered number is Even or Odd.
  - b. Program to swap two numbers.
  - c. Program to get a number and display the sum of the digits.
  - d. Program to get a number and display the number with its reverse.
- 2. C# program on Date Formats and Years:**
  - a. Program to check whether the entered year is a leap year or not.
  - b. Program to display the date in various formats.
  - c. Program to compare two dates.
  - d. Program to display the ATM Transaction.
- 3. C# program on Fundamental Mathematical Operations:**
  - a. Program to illustrate Single Inheritance.
  - b. Program to illustrate Hierarchical Inheritance.
  - c. Program to demonstrate Multilevel Inheritance.
- 4. C# program on Delegates:**
  - a. Program to combine two Delegates.
  - b. Program to display results using Delegates.
- 5. C# program on Events:**
  - a. Program to perform addition with MouseUp Event.
  - b. Program to add two TimeSpan.
  - c. Program to perform subtraction with KeyUp Event.

**Scheme of Marks:**

|                                                      |   |          |
|------------------------------------------------------|---|----------|
| 1. C# program on Fundamental Mathematical Operations | : | 10 Marks |
| 2. C# program on Date Formats and Years              | : | 10 Marks |
| 3. C# program on Fundamental Mathematical Operations | : | 10 Marks |
| 4. C# program on Delegates                           | : | 10 Marks |
| 5. C# program on Events                              | : | 10 Marks |
| 6. Practical Record                                  | : | 20 Marks |
| 7. Viva                                              | : | 20 Marks |

Approved for session 2019-2020

*Institute for Excellence in Higher Education (IEHE), Bhopal*

B.Sc. (HONOURS): **PART-III**

Semester: **V**

SUBJECT: **COMPUTER SCIENCE**

PAPER: **HONOURS-II/SUBSIDIARY**

max. marks: **100**

Time: **03 Hrs**

**TITLE: Internet Technology & Cyber Security**

**Course objective:**

- Build an understanding of the fundamental concept of computer network.
- Familiarize the students with the basic concept of web and cyber security.
- Introduced the student in the field of advanced Internet technology and cyber Security.

**Expected outcomes:**

- A Student will have good knowledge of HTML PROGRAMMING.
- A Student will learn use of forms, tables and hyperlinks to design web pages.
- A Student will be able to design static web pages.

**SYLLABUS**

**Networking Protocols and OSI Model:**

Protocols in computer communications, The OSI model and OSI layer function.

**Internetworking Concepts:**

Internetworking, Problem in internetworking, dealing with Hardware & Software issues, Virtual network. Internetworking Devices: Repeaters, Bridges, Routers, Gateways. Internet Basics: History and Architecture.

**TCP/IP:**

Introduction to TCP/IP Model and Layers. Addressing: Physical, Logical, Port, Specific addresses.

**IP (Internet Protocol) Addresses:** IP datagram, Concept of IP Address, Classes of IP Addresses. ARP (Address Resolution Protocol), RARP (Reverse Address Resolution Protocol), ICMP (Internet Control Message Protocol).

**TCP (Transmission Control Protocol):** Features of TCP, Relationship between TCP/IP, Ports & Sockets, TCP Connections, TCP Segment Format. UDP (User Datagram Protocol), UDP Datagram, Differences between UDP & TCP.

**Protocol and Services on Internet:**

DNS (Domain Name System), Email (Electronic Mail), FTP (File Transfer Protocol), WWW (World Wide Web), HTTP (Hyper Text Transfer Protocol), CGI (Common Gateway Interface), TELNET (Remote login), Search Engines, Web Browser.

**Web Security:**

Principles of Security, Cryptography, Plain Text and Cipher Text, Symmetric Key Algorithms, Public Key Algorithms, Digital Certificates, Digital Signature and Secure Socket Layer (SSL).

**Network Security:** Firewalls, IP Security, Virtual Private Networks (VPN).

**Cyber Security:**



Introduction of Cyber Security and Cyber Laws in India.

**Email Security:** Importance of E-mail Security, Threats through E-mails – Attachments, Fake E-Mails, Spam E-Mails. **Social Networking:** Introduction to Social Networking, Risks in Social Networking, a. Illegal content, b. Spam

**HTML (Hyper Text Markup Language):** Introduction to HTML, Structure of HTML 5 code, Basic Tags and Attributes of HTML working with Image, Background, colouring Text & Images. **Lists:** List item, unordered list, ordered list, Definition list.

**Links:** Creating Hyper links, Using Graphical Hyperlinks, Creating Anchors and Linking to section in a page, E-mailing with Hyper Linking. **Tables:** TABLE, TR & TD Tags, Cell Spacing & Cell Spacing & Cell Padding and Colspan & Rowspan, CSS (Cascading Style Sheet). **Forms:** FORM and INPUT Tag, Adding Controls, Controls-Button, Checkbox, Image, Radio Button, Text Box, Password, Text are, Select, Label, Submit & Reset.

**Text books:**

1. Web Technologies: TCP/IP to Internet Application Architectures by Achyut S. Godbole and Atul Kahate, McGraw Hill Education, 3<sup>rd</sup> Edition.
2. TCP/IP Protocol Suite by Behrouz A. Forouzan, McGraw Hill Education; 4<sup>th</sup> Edition.
3. Fundamentals of Cyber Security by Mayank Bhushan, BPB Publications; 1<sup>st</sup> edition (1 August 2017)

**Reference Books:**

1. O Level Internet Technology & Web Design Edition by Satish Jain, Shashank Jain, BPB Publications (M2-R4) 2007
2. Introduction to Computer Networks and Cybersecurity by Chwan-Hwa (John) Wu, J. David Irwin, CRC Press; 1 edition (1 March 2013).

*[Handwritten signature]*

*[Handwritten signature]*

*[Handwritten signature]*

*[Handwritten signature]*

*[Handwritten signature]*

*[Handwritten signature]*

*[Handwritten signature]*

*[Handwritten signature]*

*[Handwritten signature]*

*[Handwritten signature]*



# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. (HONOURS): **PART-III**Semester: **V**SUBJECT: **COMPUTER SCIENCE**PAPER: **HONOURS-II/SUBSIDIARY**TITLE: **Internet Technology & Cyber Security**Maximum Marks: **100**

Time :03 HRS

## PRACTIAL LIST

1. Introduction to HTML create a Basic HTML FILE.
2. Write the HTML Page to Print Hello World.
3. Create a web page illustrating text formatting tags available in HTML.(<h1>,<b>,<u>,<i>).
4. Create file using Basic tags in HTML.
5. Create a web page to illustrate three types of lists in HTML.
6. Create a static web pages using table tags of HTML.
7. Create a static web page which defines all text formatting tags HTML in tabular format.
8. Create a web page using frames for division.
9. Create HTML pages which display 3 images at LEFT,RIGHT AND CENTER respectively
10. Create web pages using list tags of HTML.

### HTML List: Ordered, Unordered and Definition List

1. Develop a program related to unit wise in computer laboratory
2. Develop any module to be useful in real life application

### SCHEME OF MARKS

Implementing Basic concept of HTML Programming.  
 Creating Static web Pages with all formatting text.  
 Creating Module for real life applications.  
 Practical Record  
 VIVA

20 Marks  
 20 Marks  
 20 Marks  
 20 Marks  
 20 Marks

***Institute for Excellence in Higher Education (IEHE), Bhopal***

**B.Sc.(HONOURS) : PART-III**

**Semester: VI**

**SUBJECT: Computer Science**

**PAPER: Honours-I**

max. marks: 100

Time : 03 Hrs

**TITLE: Object Oriented Programming with Java**

**SYLLABUS**

Object: Oriented Paradigm, Benefits of OOP, Applications of OOP.

Java History, Java Features, How Java Differs from C and C++, Java and internet, Java and World Wide Web, Web Browsers, Hardware and Software Requirements, Java Supports Systems, Java Environment.

Java Program Structure, Java Tokens, Java Statements, Implementing a Java Program, Java Virtual Machine, Command Line Arguments, and Programming Style.

Constants, Variables, Data Types, Declaration of Variables, Giving Values to Variables, Scope of Variable, Symbolic Constants, Type Casting, Getting Values of Variables, Standard Default Values.

Arithmetic Operator, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operators, Bitwise Operators, Special Operators, Arithmetic Expressions, Evaluation of Expressions, Precedence of Arithmetic Operators, Type Conversions in Expressions, Operator Precedence and Associativity, Mathematical Functions.

Decision Making with if Statement, Simple if Statement, The if.....Else Statement, Nesting of if ...Else Statement, The if else Ladder, The Switch Statement, The? Operator. The While Statement, The do Statement, The for Statement, Jump in Loops, Labeled Loops.

Defining a Class, Adding Variables, Adding Methods, Creating Objects, Accessing Class Members, Constructors, Methods Overloading, Static Members, Nesting of Methods, Inheritance; Extending a Class, Overloading Methods, Final Variables and Methods, Final Classes, Finalize Methods Abstract Methods and Classes, Visibility Control Arrays, One Dimensional Array, Strings, Vectors, Wrapper Classes.

Defining Interfaces, Extending Interfaces, Implementing Interfaces, Accessing Interface Variables.

Java API Packages, Using System Packages, Naming Conventions, Creating Packages, Accessing a Package, Using a Package, Adding a Class to a Package, and Hiding Classes.

Approved for the session 2019-20



Creating Threads, Extending the Thread Class, Stopping and Blocking a Threads, Life Cycle of a Thread, Using Threads Methods, Threads Exceptions, Threads Priority, Synchronization, Implementing the 'Runnable interface.

Types of Errors, Exceptions, Syntax of Exception Handling Code, Multiple Catch Statements, Using Finally Statements, Throwing Our Own Exceptions, Using Exceptions for Debugging.

Preparing to Write Applets, Building Applet Code Applet Life Cycle, Creating an Executable Applet, Designing a Web Page, Applet Tag, Adding Applet to HTML File, Running the Applet, More About the Applet tag, Passing Parameters to Applets, Aligning the Display, More About HTML Tags, Displaying Numbering Values, Getting Input from the user.

The Graphics Class, Lines and Rectangles, Circles and Ellipses, Drawing Arcs, Drawing Polygons, Line Graphs, Using Control Loops in Applets, Drawing Bar Charts.

Concept of Stream, Stream Classes, Byte Stream Classes, Character Stream Classes, Using Streams, Other Useful I/O Classes, Using the File Class, Input / Output Exceptions, Creation of Files, Reading / Writing Characters, Reading / Writing Bytes, Handling Primitive Data Types, Concatenating and Buffering Files, Random Access, Files, Interactive Input and Output, other Stream Classes

### TEXT BOOK

1. Programming with Java, E Balguruswami, Tata McGraw-Hill Publication, 2<sup>nd</sup> Edition

### REFERENCE BOOKS

1. The Complete Reference, Herbert Schildt, Seventh Edition
2. Java Projects, BPB, BPB Publication, 2004
3. Programming in Java, Dr. S.S. Kandare, S Chand Publication

*Anand*

*1*  
*guy*

*h-*  
*b*

*Blag*

*26/3/19*

*11/Jan*

*huka*

*26/4/19*

Approved for the session 2019-20



# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: **PART-III**Semester: **VI**

SUBJECT: COMPUTER SCIENCE

PAPER: HONOURS-I

## Title : Object Oriented Programming with Java

Max. Marks: 100

Time: 3 hrs

### PRACTICAL SYLLABUS

#### 1) Java Program based on Mathematical Problems :

- a. Write a program to find the average of three numbers using command line argument.
- b. Write a program to find the average of three numbers using scanner class.
- c. Write a program to illustrate the narrowing and widening type casting.
- d. Write a program of swapping a variable without using third variable.
- e. Write a program to determine the sum of following harmonic series for a given value of n:  
 $1 + 1/2 + 1/3 + 1/4 + \dots + 1/n$
- f. Write a program to calculate the volume of right circular cone and cuboid.
- g. Write a program to illustrate the increment and decrement operator.
- h. Similar other programs.

#### 2) Specific Java Programs (Based on Control statement/looping/ Strings):

- a. Write a program to find the greatest among three numbers by using nesting *if.....else* statement.
- b. Write a program to illustrate the switch case statements.
- c. Write a program to find the number of and sum of all integers greater than 100 and less than 200 that are divisible by 7.
- d. Write a program to compute the sum of the digits of a given integer number.
- e. Write a program to reverse the digits of the number by using *while* loop.
- f. Write a program to illustrate the Fibonacci numbers by using *do.....while* loop.
- g. Write a program to print the following some pattern:-

| (1)   | (2)   | (3)       |
|-------|-------|-----------|
| *     | 1     | 1         |
| ***   | 01    | 2 2       |
| ***** | 101   | 3 3 3     |
| ***** | 0101  | 4 4 4 4   |
| ***** | 10101 | 5 5 5 5 5 |

*Handwritten signature*

*Handwritten signature*

*Handwritten signature*

*Handwritten signature*

*Handwritten signature*

*Handwritten signature*

*Handwritten signature*

*Handwritten signature*

- h. Write a program to illustrate the control statements of "break", "goto" and "continue".
- i. Write a program to illustrate the String functions strcpy, strlen, strcmp and strcat.
- j. Similar other programs.

### 3) Java Program based on Methods, Arrays, Inheritance and Interfaces:

- Write a program to illustrate the default constructor and parameterized constructor.
- Write a program to illustrate the method overloading.
- Write a program to illustrate the static variables and methods.
- Write a program to illustrate the nesting of methods.
- Write a program to illustrate the single inheritance.
- Write a program to illustrate the method overriding.
- Write a program to illustrate the final variables and final methods.
- Write a program to illustrate the constructor overloading.
- Write a program to illustrate the abstract method.
- Write a program to sort a list of given numbers in 1-D array.
- Write a program to multiply two 2-D arrays.
- Write a program to implement the interface.
- Write a program to create a package and use its methods in another class.
- Similar other programs.

#### 4) Java Program based on Multithreading and Exception Handling

- Write a program to illustrate the multi threading.
- Write a program to illustrate the multi threading with thread priorities.
- Write a program to illustrate the “Runnable Interface”.
- Write a program to illustrate the thread methods.
- Write a program to illustrate the exception handling.
- Write a program to illustrate the multiple catch statements.
- Write a program to illustrate the finally statements.
- Write a program to illustrate throwing our own exceptions.
- Similar other programs.

### 5) Java Program based on Java Applet and Files in Java :

- Write a program to add two numbers using applet programming.
- Write a program to draw a circle and rectangle and filled both with color blue.
- Write a program to draw a bar chart and line chart by using applet.
- Write a program to display numerical values in applet.
- Write a program to illustrate the interactive input to an applet.
- Write a program to draw a human face by using applet.
- Write a program to illustrate the copying characters from one file into another.
- Write a program to illustrate the writing, reading and copying bytes from a file.
- Similar other programs.

Am








**Schemes of Marks :**

|                                                                         |          |
|-------------------------------------------------------------------------|----------|
| 1 Java Program based on Mathematical Problems :                         | 10 Marks |
| 2 Specific Java Programs (Based on Control statement/looping/ Strings): | 10 Marks |
| 3 Java Program based on Methods, Arrays, Inheritance and Interfaces:    | 20 Marks |
| 4 Java Program based on Multithreading and Exception Handling           | 10 Marks |
| 5 Java Program based on Java Applet and Files in Java :                 | 10 Marks |
| 6 Practical Record                                                      | 20 Marks |
| 7 Viva                                                                  | 20 Marks |

*Anand*

*20/3/2019*

*[Signature]*

*[Signature]*

*[Signature]*

*[Signature]*

*[Signature]*

*20/3/2019*

*[Signature]*

*[Signature]*

*[Signature]*



# Institute for Excellence in Higher Education (IEHE), Bhopal

Semester: VI

B.Sc. Honours: PART-III

SUBJECT: COMPUTER SCIENCE

PAPER: HONOURS – II/Subsidiary

Time: 03 Hrs

Max. Marks: 100

## TITLE: SOFTWARE ENGINEERING

### Course Objective:

In this course, students will gain a broad understanding of the discipline of software engineering and its application to the development of and management of software systems.

### Learning Outcome:

- knowledge of basic SW engineering methods and practices, and their appropriate application;
- A general understanding of software process models such as the waterfall and evolutionary models.
- An understanding of the role of project management including planning, scheduling, risk management, etc.
- An understanding of software requirements and the SRS document.
- An understanding of different software architectural styles.
- An understanding of implementation issues such as modularity and coding standards.
- An understanding of approaches to verification and validation including static analysis, and reviews.
- An understanding on quality control and how to ensure good quality software.
- An understanding of some ethical and professional issues that are important for software engineers.
- development of significant teamwork and project based experience

Maximum Marks 100

## SYLLABUS

**INTRODUCTION:** Characteristics of Software, Software Engineering, EVOLVING role of software, the changing nature of software, System software, Application software, Engineering/Scientific software, Embedded software, Product-line software, Web-applications Software Engineering: A Layered Technology, process, Methods and Tools, A Generic view of software Engineering, The Software Process, Software Process Models – The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Software Process Model – The Incremental Model, The Spiral Model, The Component Assembly Model, The Concurrent Development Model. The Formal Methods Model Fourth Generation Techniques, Process Technology, Product and Process.

*[Handwritten signatures and initials at the bottom of the page]*

**PROJECT MANAGEMENT CONCEPT:**

The Management Spectrum – The People, The Product, The Process, The Project. People- The Players, Team Leaders, The Software Team

**Software Process and Project Metrics:** Introduction to measures, metrics and indicators. Process Metrics and Software process Improvement, Project Metrics, Software Measurement – Size Oriented Metrics, Function Oriented Metrics.

**SOFTWARE PROJECT PLANNING AND QUALITY ASSURANCE:** Observations on Estimating, Project Planning Objectives, Software Scope- Obtaining Information Necessary for Scope, Feasibility. Resources – Human Resources, Reusable Software Resources, Environment Resources.

**Software Quality Assurance:** Quality Concepts, Quality, Quality Control, Quality Assurance, Cost of Quality, The Quality Movement, Software Quality Assurance- Background Issues, SQA Activities. The review meeting, Review Reporting and Record Keeping.

**ANALYSIS CONCEPTS AND PRINCIPLES**

Requirements Analysis, Requirements, Elicitation for software Initiating the Process, Facilitated Application specification Techniques, Quality function Deployment, Analysis Principles – the Information Domain, Modeling, Portioning, Essential and Implementation Views. Specification – Specification Principles, Representation, the Software Requirements Specification

**FUNCTIONAL MODELING AND INFORMATION FLOW:**

Data Flow Diagram (DFD), Extensions for real- Time systems, Behavioral Modeling, The Mechanics of Structural analysis –Creating an Entity/Relationship Diagram, Creating a Dataflow Model, Creating a Control Flow Model, The Control Specification, The Control Specification, The Process Specification. The Data Dictionary..

**DESIGN CONCEPT AND PRINCIPLES:**

Software Design and Software Engineering, The Design Process- Design and Software Quality, The evaluation of software Design. Design Principles, Design Concepts- Abstraction, Refinement, Modularity, Software Architecture, Control Hierarchy, Structural Portioning, Data Structure, Software Procedure and Information Hiding, Effective Modular Design- Fundamental Independence, Cohesion, Coupling. Design Documentation.

**SOFTWARE TESTING TECHNIQUES**

Software testing fundamental – testing objectives, testing principles, testability. Test case design, white box testing, basic path testing- flow graph notation, cyclomatic complexity, deriving test cases, graph matrices. Control structure testing- condition testing, data flow testing, loop testing. Black box testing- graph based testing methods, equivalence portioning, boundary value analysis, comparison testing, orthogonal array testing.

**MAINTENANCE:**

Software Maintenance-Software Supportability- Reengineering Business Process Reengineering- Software Reengineering- Reverse Engineering Restructuring- Forward Engineering- Economics of Reengineering.



### TEXT BOOKS

1. Software Requirements, Karl E. Wiegers, Word Power Publishers, 2000.
2. Software Requirements and Estimation, Rajesh Naik, Swapna Kishore, TMH

### REFERENCES:

1. Requirements Engineering: A Good practice Guide, Ian Sommerville, Pete Sawyer, Pearson, 2004
2. Managing Software Requirements A Use Case Approach, 2/e, Dean, Don, Addison-Wesley, 2003
3. Requirements Engineering and Rapid Development, Ian Graham, Addison-Wesley, 1998
4. Mastering the Requirements Process. 2/e, S.Robertson, J.Robertson, Pearson, 2006



# Institute for Excellence in Higher Education (IEHE), Bhopal

B.Sc. Honours: PART-III

Semester: VI

SUBJECT: COMPUTER SCIENCE

PAPER: HONOURS – II/SUBSIDIARY

MAX MARKS: 100

## TITLE: SOFTWARE ENGINEERING PRACTICAL SYLLABUS

**Pre-requisites:** Basic programming knowledge of Asp.net and c#.and dbms and oracle and.java

### Introduction

In this lab you will practice the software development life cycle (project management, requirements engineering, systems modeling, software design, prototyping, and testing) using CASE tools within a team work environment. •UML notation is covered in this lab as the modeling language for analysis and design.

### Requirement of the lab

#### Hardware requirements:

Pentium 4 processor(2.4GHz)128 Mb RAM standard keyboard and mouse, coloured monitor.

#### Software requirements:

Microsoft visual studio 2017.

**List of practical:** Following type of projects has been developed by the students which include the analysis and design of the system. A documentation covering all phases of SDLC (SRS, HLD, LLD, Software Development, Testing, Implementation/Deployment).

- Inventory control system
- Online shopping system
- Railway reservation system
- Fast food billing system
- blood bank system.
- Online doctor appointment system
- Hotel management system
- Movie ticket booking system.
- Online book shop system
- Online quiz etc.
- Cab booking system
- Online recruitment system.
- Orkut using Java
- Traffic Monitoring
- Personal Health Monitoring
- E-fashion
- Examination branch system

o Information portal.

### Scheme of Marks:

- |                          |          |
|--------------------------|----------|
| 1. Knowledge of Language | 10 Marks |
| 2. Project Reviews       | 20 Marks |
| 3. Project Report        | 20 Marks |
| 4. Viva                  | 50 Marks |