**Scheme of Examination**

**for**

**Bridge Course(s)**

**for**

**MCA 2-Year Programme**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Course No./** **Course Name** | **Course Type** | **Workload Hours/ Week** | **Internal Marks** | **External Marks** | **Total** **Marks** | **Credits** |
| **MCA-BC-01** MCA Bridge Course – 01 |  Theory | 4 | 30  | 70  | 100 | 4 |
| Practical | 4 | - | 50 | 50  | 2  |
| **MCA-BC-02** MCA Bridge Course – 02 |  Theory | 4 | 30  | 70  | 100 | 4 |
| Practical | 4 | - | 50  | 50  | 2  |

**Note:** Candidates admitted under category **3a(ii)[[1]](#footnote-1)** of the MCA ordinance shall have to pass bridge courses designed and offered by the University during the duration of MCA programme failing which their MCA degree shall get delayed till the time they pass these courses.

Bridge Courses’ examination will be conducted by the University. The student has to secure 40% marks in both theory and practical examination in order to pass the bridge course(s). Examination fee shall be made applicable as per University rules. The respective University Teaching Department/Affiliated College shall arrange for the contact sessions for completing the bridge course(s). The University/College/Institute shall not charge any fee for the conduct of bridge course(s), however, the contact classes for bridge course(s) shall count towards teaching workload.

Candidates belonging to category **3a(ii)** of the MCA ordinance and having earned a diploma or a degree of at least 6-months duration in computer science, or information technology, or computer science & engineering, or software engineering shall be exempted from the requirement of studying and passing the bridge courses.

|  |
| --- |
| **MCA-BC-01: MCA Bridge Course 01** |
| Course Type | Course Credit | Contact Hours/ Week | Delivery Mode | Maximum Marks | Exam Duration | Assessment Methods |
| External | Internal |
| Bridge Course  | 4 Theory | 4 Theory | Lecture | 70 | 30 | TEE 3 Hours | MTE 2 Hours | TEE/MTE/ Assignment/ Attendance |
| 20 | 5 | 5 |
| 2 Practical | 4 Practical | Practical | 50 | - | TEE3 Hours | TEE/ Practical File |
| **Instructions for Mid-Term examination:** The mid-term examination shall cover the first two units of the course content. The question paper will be set by the internal teacher.**Instructions to paper setter for Term-End Examination:** The term-end examination shall cover the whole content of the course. The total number of questions shall be nine. Question number one will be compulsory and will be consisting of short/objective-type questions from the complete syllabus. In addition to the compulsory first question, there shall be four units in the question paper each consisting of two questions. The student will attempt one question from each unit in addition to the compulsory question. All questions will carry equal marks. |
| **Course Objectives:** The objective of the course is to bridge the gap between subjects studied by the BCA/Bachelor of CSE students and the BA/BSc/BCom students. The students taking this bridge course shall be taught in foundational mathematics, fundamental concepts of computers and C programming language  |
|

|  |  |
| --- | --- |
| **Course Outcomes**  | At the end of this course, the student will be able to:  |
| CO1 | name and enlist various programming constructs of C language; enumerate fundamental concepts of computer system and computer networks; and concepts of numbers, set theory, group theory and graph theory.  |
| CO2 | understand and describe various programming constructs of C language; describe fundamental concepts of computer system and computer networks; and concepts of numbers, set theory, group theory and graph theory. |
| CO3 | use various programming constructs of C language and mathematical concepts to solve simple problems through C programs;  |
| CO4 | differentiate between various data types, flow-control constructs, structures and unions, computer types, storage types, problem solving tools, types of networks, numbers, sets, groups and graphs. |
| CO5 | choose between various flow-control constructs, standard/user defined data type of C language, design representation tools, transmission media, network topologies, mathematical structures that suits better in a given situation. |
| CO6 | design and represent the solutions of simple problems and create C programs therefor.  |

|  |
| --- |
| **CO-PEO Mapping Matrix for Course MCA-BC-01** |
| COs | PEO1 | PEO2 | PEO3 | PEO4 | PEO5 |
| CO1 |  1 | 3 | 3 | 3 | 3 |
| CO2 | 2 | 3 | 3 | 3 | 3 |
| CO3 | 3 | 3 | 3 | 3 | 3 |
| CO4 | 3 | 3 | 3 | 3 | 3 |
| CO5 | 3 | 3 | 3 | 3 | 3 |
| CO6 | 3 | 3 | 3 | 3 | 3 |
| Average | 2.5 | 3 | 3 | 3 | 3 |
| **CO-PO Mapping Matrix for Course MCA-BC-01** |
| COs | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  1 | 3 | 1 | 1 | 1 | - | 3 | 1 | - | - | - | - |
| CO2 | 2 | 1 | 1 | 3 | 1 | - | 3 | 2 | - | - | - | - |
| CO3 | 3 | 1 | 1 | 3 | 3 | - | 3 | 3 | - | - | - | - |
| CO4 | 3 | 3 | 1 | 3 | 1 | - | 3 | 3 | - | - | - | - |
| CO5 | 3 | 1 | 3 | 1 | 3 | - | 3 | 3 | - | - | - | - |
| CO6 | 3 | 3 | 3 | 3 | 3 | - | 3 | 3 | - | - | - | - |
| Average | 2.5 | 2 | 1.66 | 2.33 | 2 | - | 3 | 2.5 | - | - | - | - |
| **CO-PSO Mapping Matrix for Course MCA-BC-01** |  |  |  |  | 3 |
| COs | PSO1 | PSO2 | PSO3 | PSO4 | PSO5 |
| CO1 | 3 | 1 | 1 | 1 | - |
| CO2 | 3 | 2 | 2 | 2 | - |
| CO3 | 3 | 3 | 3 | 3 | - |
| CO4 | 3 | 3 | 3 | 3 | - |
| CO5 | 3 | 3 | 3 | 3 | - |
| CO6 | 3 | 3 | 3 | 3 | - |
| Average | 3 | 2.5 | 2.5 | 2.5 | - |

 |
| **Course Content****MCA-BC-01: MCA Bridge Course – 01** |
| Unit - I | **Elements of C language**: C character set, identifiers & keywords, data types: declaration & definition.Operators: Arithmetic relational, logical, bitwise, unary, assignment and conditional operators & their hierarchy & associativity, Data input/output.Control statements: Sequencing, Selection: if and switch statement; iteration, repetition: for, while, and do-while loop; break, continue, go to statement. |
| Unit - II | **Functions in C language**: Definition, prototype, passing parameters, recursion, Data structure: arrays, structures, union, string, data files.Pointers: Delectation, operations on pointers, array of pointers, pointers to arrays.C preprocessors |
| Unit - III | **Computer Fundamentals**: introduction, characteristics of computers, the evolution of computers, the computer’s generations, basic computer organization, storage hierarchy, primary & secondary storage, input-output devices. Computer Software: introduction, relationship between hardware and software, types of software, planning the computer program: purpose of program planning, algorithm, flowcharts, decision tables, pseudocodes, application software packages.Data Communications and Computer Networks: Introduction, data transmission modes, data transmission speed, transmission media, digital and analog transmission, the internet, multimedia. |
| Unit - IV | **Foundational Mathematics**: Types of numbers and their properties, natural numbers, whole numbers, integers, real numbers, rational numbers, irrational numbers, complex numbers, imaginary numbers.Set theory: Basic concept, set types, set operations, cardinality, and notation. Group theory: Basic concept, subgroups, group axioms, subgroups, co-sets, normal subgroups, semigroups.Graph theory: Directed and undirected graphs, chains, circuits, paths, cycles, connectivity, adjacency matrix. |
| **Text/Reference Books** |
| Text Books | 1. Pradeep K. Sinha & Priti Sinha, Computer Fundamentals, BPB Publications
2. Rajaraman V, Fundamentals of Computers, Prentice Hall of India.
3. Seymour Lipschutz, Marc Lars Lipson, Discrete Mathematics, McGraw-Hill International Editions, Schaum's Outline Series.
4. Yashwant Kanetkar, Let us C, BPB Publications.
5. Behrouz, Frozen, Introduction to Data Communications and Networking, Tata McGraw Hill.
 |
| Reference Book | 1. Thomas C. Bartee. 1985. Digital computer fundamentals (6th ed.). McGraw-Hill, Inc., USA.
 |

|  |
| --- |
| **MCA-BC-02: MCA Bridge Course 02** |
| Course Type | Course Credit | Contact Hours/ Week | Delivery Mode | Maximum Marks | Exam Duration | Assessment Methods |
| External | Internal |
| Bridge Course  | 4 Theory | 4 Theory | Lecture | 70 | 30 | TEE 3 Hours | MTE 2 Hours | TEE/MTE/ Assignment/ Attendance |
| 20 | 5 | 5 |
| 2 Practical | 4 Practical | Practical | 50 | - | TEE3 Hours | TEE/ Practical File |
| **Instructions for Mid-Term examination:** The mid-term examination shall cover the first two units of the course content. The question paper will be set by the internal teacher.**Instructions to paper setter for Term-End Examination:** The term-end examination shall cover the whole content of the course. The total number of questions shall be nine. Question number one will be compulsory and will be consisting of short/objective-type questions from the complete syllabus. In addition to the compulsory first question, there shall be four units in the question paper each consisting of two questions. The student will attempt one question from each unit in addition to the compulsory question. All questions will carry equal marks. |
| **Course Objectives:** The objective of the course is to bridge the gap between subjects studied by the BCA/Bachelor of CSE students and the BA/BSc/BCom students. The students taking this bridge course shall be taught in foundational mathematics, fundamental concepts of computers and C programming language  |
|

|  |  |
| --- | --- |
| **Course Outcomes**  | At the end of this course, the student will be able to:  |
| CO1 | list/name various topics/terms/concepts in object orientation, C++ language, digital fundamentals, and computer organisation. |
| CO2 | understand and describe various topics/terms/concepts in object orientation, C++ language and relate the features of C++ to concepts of object orientation as also the working of digital systems and organisation of a digital computer. |
| CO3 | apply/use various constructs of C++ language and digital principles to write simple object oriented programs. |
| CO4 | classify and illustrate various programming constructs of C++, computer codes, binary operations, logic gates, and digital circuits. |
| CO5 | choose and justify a particular C++ feature, a Boolean simplification method, a particular digital circuit, an instruction format or an addressing mode. |
| CO6 | use the knowledge of C++, digital fundamentals and computer organisation to create solutions for simple problems in an object oriented manner. |

|  |
| --- |
| **CO-PEO Mapping Matrix for Course MCA-BC-02** |
| COs | PEO1 | PEO2 | PEO3 | PEO4 | PEO5 |
| CO1 |  1 | 3 | 3 | 3 | 3 |
| CO2 | 2 | 3 | 3 | 3 | 3 |
| CO3 | 3 | 3 | 3 | 3 | 3 |
| CO4 | 3 | 3 | 3 | 3 | 3 |
| CO5 | 3 | 3 | 3 | 3 | 3 |
| CO6 | 3 | 3 | 3 | 3 | 3 |
| Average | 2.5 | 3 | 3 | 3 | 3 |
| **CO-PO Mapping Matrix for Course MCA-BC-02** |
| COs | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  1 | 3 | 1 | 1 | 1 | - | 3 | 1 | - | - | - | - |
| CO2 | 2 | 1 | 1 | 3 | 1 | - | 3 | 2 | - | - | - | - |
| CO3 | 3 | 1 | 1 | 3 | 3 | - | 3 | 3 | - | - | - | - |
| CO4 | 3 | 3 | 1 | 3 | 1 | - | 3 | 3 | - | - | - | - |
| CO5 | 3 | 1 | 3 | 1 | 3 | - | 3 | 3 | - | - | - | - |
| CO6 | 3 | 3 | 3 | 3 | 3 | - | 3 | 3 | - | - | - | - |
| Average | 2.5 | 2 | 1.66 | 2.33 | 2 | - | 3 | 2.5 | - | - | - | - |
| **CO-PSO Mapping Matrix for Course MCA-BC-02** |
| COs | PSO1 | PSO2 | PSO3 | PSO4 | PSO5 |
| CO1 | 3 | 1 | 1 | 1 | - |
| CO2 | 3 | 2 | 2 | 2 | - |
| CO3 | 3 | 3 | 3 | 3 | - |
| CO4 | 3 | 3 | 3 | 3 | - |
| CO5 | 3 | 3 | 3 | 3 | - |
| CO6 | 3 | 3 | 3 | 3 | - |
| Average | 3 | 2.5 | 2.5 | 2.5 | - |

 |
| **Course Content****MCA-BC-02: MCA Bridge Course II** |
| Unit - I | **Object oriented concept:** Data abstraction, encapsulation, classes and objects modularity, hierarchy, typing, concurrency, object-oriented methodology: advantages and disadvantages of OO methodologies. aggregation, generalization and inheritance, abstract class, meta data, object diagram, dynamic model – events, states, scenarios, event traces, state diagram; functional model- data flow diagram, analysis, system design and object design. |
| Unit - II | **C++ Programming:** Data types, structures vs classes, static data and member function, constant parameters and destruction, dynamic objects, operator overloading, function overloading, abstract class, virtual class, inheritance, virtual functions, template functions & template classes, exception handling, I/O streams. |
| Unit - III | **Digital Fundamentals:** Information representation - number systems, codes, binary arithmetic operations; number systems - non positional number system, positional number system, number system conversion, fractional number conversion; computer codes - BCD code, EBCDIC code, ASCII, binary arithmetic - addition, subtraction, multiplication, division; binary logic - Boolean algebra, Boolean functions, truth table, simplification of Boolean functions, digital logic gates. |
| Unit - IV | **Computer Organisation:** Combinational logic - adders, subtractors, encoder, decoder, multiplexer, demultiplexer and comparators; processor organisation - machine instructions, instruction cycles, instruction formats and addressing modes, microprogramming concepts, microprocessor sequence; sequential logic - flip flops, shift registers and counters; I/O organisation - I/O interface, interrupt structure, transfer of information between CPU, memory and I/O devices. |
| **Text/Reference Books** |
| Text Books | 1. Rumbaugh. J. et. al., Object Oriented Modeling and Design, Prentice Hall of India 1998.
2. Balaguruswami, Object Oriented Programming in C++ : Tata McGraw
3. Pradeep K. Sinha & Preeti Sinha, Computer Fundamentals, BPB Publications
4. Rajaraman V, Fundamentals of Computers, Prentice Hall of India
5. Mano. M. Morris Digital Logic & Computer Systems Design, Prentice Hall of India Pvt. Ltd., 2000.
 |
| Reference Book | 1. Stroustrup, B., The C++ programming language, Addison –Wesley1993.
 |

1. 3. A person who has passed one of the following examinations of this University or an examination recognized as equivalent thereto shall be eligible to join the First Semester of the programme:

a(i) Passed BCA/Bachelor Degree in Computer Science Engineering or equivalent Degree.

OR

a(ii) Passed BSc/BCom/BA with Mathematics at 10+2 Level or at Graduation level (with additional bridge courses as per the norms of Ch. Devi Lal University, Sirsa). [↑](#footnote-ref-1)