X AC- 3 (Special)
01-03-2021

GOA UNIVERSITY
Taleigao Plateau, Goa 403 206

A G E N D A

For the 3rd Special Meeting of the

X ACADEMIC COUNCIL

Day & Date

1st March, 2021

Time

10.30 a.m.

Online Via Google Meet

Third Meeting (Special) of the X Academic Council

Date: 01-03-2021

Time: 10.30 a.m.

Venue: Online via Google Meet

D	**DISCUSSIONS**
D 3	**BOARDS OF STUDIES**
D 3.1	**Minutes of Meeting of the Board of Studies in Computer Science & Technology (PG) held on 12.02.2021.** **Part A** i. Recommendations regarding courses of study in the subject or group of subjects at the undergraduate level: NA ii. Recommendations regarding courses or group of subjects **at postgraduate level:** The BoS was called to recommend necessary changes from the 3 year MCA degree programme to the 2 year MCA, in compliance with the AICTE recommendation dated 3/7/2020. The **AICTE recommendation** dated 3/7/2020 is at **Annexure I (refer page no 1)** A) The BoS in Computer Science recommended the compliance to the 2 year MCA degree program from AY 2021-22 with the following **eligibility criteria**: To be eligible for admission to the 2 year program leading to the degree of "Master of Computer Applications" (MCA), a candidate shall have: a. (i) Passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent degree. OR (ii) Passed graduation in a non Computer Science discipline with Mathematics at 10+2 level or at Graduation level (with additional bridge courses). b. Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying examination c. The admission to MCA program will be strictly based on the merit list prepared via the **Aptitude Entrance Test** which will be notified from time to time. B) i) The **syllabus for the Aptitude Entrance Test** was accordingly revised taking into consideration the changes to the current eligibility criteria and the requirement of the basic computer science prerequisite knowledge for admission to the 2 year program. **This Aptitude Entrance Test would be in effect from AY 2022-23.** ii) For the current AY 2021-22, candidates would be required to answer the Aptitude Test based on the existing syllabus.

iii) The BoS also recommended **a cut off of 30%** (percentage obtained by candidate to be rounded up to the next integer) at the Aptitude Entrance Test which will have negative marking. The test would have an equal weightage for the Aptitude and Computer Science concept components.

The **Aptitude Entrance** Test syllabus is at **Annexure II** **(refer page no 2).**

C) In conformance to the clause of additional bridge courses for non Computer Science candidates, the BoS proposed that the content required for the bridge course "Fundamental Concepts in Computer Science" to be done via self study using identified content from existing MOOCs courses. The evaluation of the bridge course would consist of a written component and a skill based practical component which will be conducted by the programme. Candidates would be required to obtain a minimum of 40% marks in each individual component, to be considered as "passed". The assessment of the bridge course would be done in Sem1 and, although not credited, a candidate would require to have a pass in this course to obtain the MCA degree.

Syllabus for the **bridge course** "Fundamental Concepts in Computer Science" witcorresponding MOOC urls is at **Annexure III** **(refer page no 3)**

D) i) The basic objective of the MCA degree program is to prepare learners with the Fundamental knowledge and skills required for Application Development. Hence the Laboratory and Internship become an integral part of the Curriculum.

ii) The motivation for the 2 year MCA degree is that fundamental concepts in Computer Science are already covered at graduation level and courses in the 2 year MCA degree program would need to take this into consideration while designing the revised structure and curriculum.

iii) Keeping in view the objective and the motivation, the 2 year MCA curriculum is restructured so as to give learners a foundation in topics to be able to further venture into domains of (Data Science/Application Development/Ambient Computing/ Systems) via choice of electives

The structure with associated credits for the 2-year MCA degree program was discussed and finalized.
The structure (course codes conform to OB 10.22) approved by the BoS is attached at **Annexure IV** **(refer page no 5).**
The syllabus of the proposed courses for Semester 1 is at **Annexure V** **(refer page no 7)**.
The list of proposed Electives is at **Annexure VI** **(refer page no 23)**.

E) The **Proposed Ordinance** changes reflecting the revised MCA program are is submitted in three column format at **Annexure VII** **(refer page no 24)**

Part B

i. Scheme of examination at undergraduate level: NA
ii. Panel of examiners for different examinations at Undergraduate Level: NA
iii. Scheme of examinations at postgraduate level : Unchanged
iv. Panel of examiners for different examinations at post-graduate Level: Unchanged

Part C

i. Recommendation regarding preparation and publication of selection of reading material in the subject or group of subjects and the names of the persons recommended for appointment to make the selection: Nil

Part D

i. Recommendations regarding general academic requirements in the Departments of University or affiliated colleges: Nil

Part E

i. Recommendations of the text books for the course of study at undergraduate level-included in the syllabus : NA
ii. Recommendations of the text books for the course of study at post graduate level: Text and Reference Books are indicated below each subject of the syllabus.

Part F

Important points for consideration/approval of Academic Council

1) Eligibility criteria changes
2) Entrance Test Syllabus
3) Bridge Course Syllabus
4) New structure and syllabi for MCA degree
5) Ordinance changes – revised ordinances
6) Ordinance changes – proposed ordinance

The declaration by the Chairman that the minutes were read by the Chairman at the meeting itself.

The minutes were circulated to the members and special invitees on 13th February 2021 via email

Sd/-

Date: 12th February 2021
Place: Goa University

Signature of the Chairperson, BOS Computer Science

Part G The Remarks of the Dean of the Faculty

i. The minutes are in order
ii. The minutes may be placed before the Academic Council with remarks if any
iii. May be recommended for approval of Academic council
iv. Special remarks if any.

Sd/-

Date: 12th February 2021
Place: Goa University

Signature of the Dean

(Back to Index)

D 3.1 Minutes of Meeting of the Board of Studies in Computer Science & Technology (PG) held on 12.02.2021.

Annexure I

Phone : 011-26131577 - 78, 80
011-29581000
Website : www.aicte-india.org

सत्यमेव जयते

अखिल भारतीय तकनीकी शिक्षा परिषद्
(भारत सरकार का एक सांविधिक निकाय)
(मानव संसाधन विकास मंत्रालय, भारत सरकार)
नेल्सन मंडेला मार्ग, वसंत कुंज, नई दिल्ली-110070

ALL INDIA COUNCIL FOR TECHNICAL EDUCATION
(A Statutory Body of the Govt. of India)
(Ministry of Human Resource Development, Govt. of India)
Nelson Mandela Marg, Vasant Kunj, New Delhi-110070

F. No. AICTE/AB/MCA/2020-21 03.07.2020

To

All AICTE Approved institutes / Universities

Subject:- **Change in the duration of MCA Program from 3 Years to 2 Years w.e.f. 2020-21-reg.**

Sir / Madam

As you may be aware that the issue regarding change in the duration of MCA program from 3 years to 2 years has been engaging the attention of UGC / AICTE for quite some time in the past.

It is informed that the above issue was placed before 545th Meeting of University Grant Commission held on 19.12.2019 and the same has been approved. Hence the MCA course shall be of 2 Years duration from 2020-21.

It is also brought to your notice that the above change in the duration of MCA program from 03 Years to 02 Years has also been incorporated in the AICTE APH 2020-21 and the eligibility qualification is as below:-

"Passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent Degree. OR Passed B.Sc./ B.Com./ B.A. with Mathematics at 10+2 Level or at Graduation Level (with additional bridge Courses as per the norms of the concerned University). Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying Examination

In view of the above, necessary modification in the course duration of MCA program may please be made in conformity with the UGC decision and provisions contained in AICTE APH for A. Y. 2020-21. A line of confirmation may please be sent to the Council in this regard immediately.

3.7.20

Member Secretary, AICTE

(Back to Index) (Back to Agenda)

Annexure II

Master of Computer Applications
Syllabus for Entrance Examination from the academic year 2022 onwards

Analytical Ability and Logical Reasoning:
The questions in this section will cover analytical and logical reasoning and are based on Series, Relationships, Classification, Coding, Permutations and Combinations and Inference, Numerical problems.

Mathematics:
Set Theory, Probability and Statistics, Logarithms, Geometric and Harmonic progressions, Determinants and Matrices, Coordinate Geometry & Applications. Basic Calculus: Limit of functions, continuous function, differentiation of function, Integration and their applications. Trigonometry & applications. Vectors: Concepts of vectors & vector algebra, applications of Vectors.

Discrete Mathematics:
Fundamentals of logic, Relations and Functions, Counting Techniques: Basics of Counting, Pigeonhole Principle, Recurrence relations, Graphs: Basic concepts of Graph and its applications. Introduction to trees, Applications of trees, Boolean Algebra and Circuits.

Programming and Basic Data Structures:
Introduction to Algorithms, Flow charts, Assembly language and high-level language, Programming in C: Tokens, Identifiers, Data Types, Sequence Control, Subprogram Control, Arrays, Structures,Functions. Data Structures: Abstract data types, stacks, queues, Singly Linked Lists.
Basic sorting algorithms: bubble sort, selection sort, insertion sort.

Computer Organization & Architecture and Operating Systems:
Basic functional blocks of a computer, Number Systems, Conversion & Arithmetic, Complements.
Introduction to operating systems, Structure and Basic functions, types of OS, Operating System Services.

Application development:
Internet and WWW Architecture, The Web browsers, HTML, Structural & formatting tags, Page elements, Tables, forms.

(Back to Index) (Back to Agenda)

Annexure III

Syllabus Content for MCA Bridge Course "Fundamentals in Computer Science"
(For Students with Qualifying Degree as non-Computer Science)

Self Study via MOOCs (urls listed below)

To be qualified for the MCA degree, students are required to pass the test in the individual theory and laboratory components of the bridge course (40% marks to be obtained in theory and lab separately) which will be conducted by the programme. However, the marks obtained, although shown on the final year grade sheet, will not be added to the CPI/SPI.

The content of the bridge course(s) will consist of the fundamentals in the following topics (percentages indicate weightage assigned to the topic for the purpose of evaluation)

Written Exam **(100 marks)**

Programming and Simple Linear Data Structures: **(40%)**

Introduction to Algorithms, Flow charts, Assembly language and high-level language Programming in C: Tokens, Identifiers, Data Types, Sequence Control, Subprogram Control, Arrays, Structures, Union, String, Pointers, Functions, File Handling, Command Line Arguments, Preprocessor directives.

Data Structures: Abstract data types, Linear Data Structures: stacks, queues, and their applications. Linked Lists: singly linked list.

Basic sorting algorithms: bubble sort, selection sort, insertion sort,

Computer Organization and Architecture & Fundamentals of Operating Systems: **(30%)**

Data Representation: Data Types, Number Systems and Conversion, Complements, Fixed Point Representation, Floating Point Representation, Binary Arithmetic - Addition and Subtraction. Computer System: Computer Components and Functions, interconnection structures, Bus Interconnections

Processor Organization: Instruction Formats, Addressing modes, Processor Organization, Register Organization, Instruction Cycle, Instruction Pipelining.

Memory System Organization: Memory Hierarchy, Internal Memory, Cache Memory.

Input/output Organisation: Peripheral devices. I/O interface, Asynchronous Data Transfer, I/O Processor.

Introduction to operating systems, Structures and Basic functions of monolithic OS, System services.

Discrete Mathematics: **(30%)**

Set Theory: Concepts of sets – Union, Intersection, Cardinality, Elementary counting; permutations and combinations.

Fundamentals of logic: Propositional and Predicate Logic, Propositional Equivalences, Predicates and Quantifiers, Rules of Inference.

Relations and Functions: Cartesian Product, Relations and their types, Properties of Relations Functions, Types of Functions, Operations on Functions

Counting Techniques: Basics of Counting, Pigeonhole Principle, Recurrence relations.

Boolean Algebras, Boolean Expression, Boolean Functions.

Lab examination shall be based on following subjects: **(100 marks)**

Programming and Simple Linear Data Structures: **(60%)**

Refer syllabus as mentioned above

Web Basics(HTML,CSS) **(40%)**

The Web browsers, HTML: HTML Overview, Structural HTML Tags, formatting text, creating links, adding image and other page elements,Tables, Frames, forms, Image mapping, cascading style sheets- inline, internal and external CSS, selectors and identifiers, css colors, borders, margin, padding, text and fonts.

Course name	Organised by	Link
Computer Organization	Prof. S. Raman, Department of Computer Science and Engineering, IIT Madras.	http://www.nptelvideos.in/2012/11/computer-organization.html
Programming and data structure	Dr. P.P. Chakraborty, Department of Computer Science and Engineering, IIT Kharagpur.	http://www.nptelvideos.in/2012/11/programming-and-data-structure.html
Operating system	PROF.SANTANU CHATTOPADHYAY Department of Computer Science Engineering IIT Kharagpur	https://nptel.ac.in/courses/106/105/106105214/ First two weeks
Discrete Mathematical Structure	Prof. Kamala Krithivasan, Department of Computer Science and Engineering, IIT Madras	http://www.nptelvideos.in/2012/11/discrete-mathematical-structures.html
Web basics		https://www.youtube.com/watch?v=mU6anWqZJcc
UNIX fundamentals		https://nptel.ac.in/courses/117/106/117106113/ first 4 Modules

(Back to Index) **(Back to Agenda)**

Annexure IV

MCA Programme Structure (2 year)

SEMESTER - 1

Course_Name		Contact_Hours/week			Credits
		L	T	P	
CSC-101	Data Structures & Algorithms	3	0	0	3
CSC-102	Object Oriented Concepts	2	0	0	2
CSC-103	Operating Systems	4	0	0	4
CSC-104	Internet Technologies	3	0	0	3
CSC-105	Data Structures & Algorithms LAB	1	0	4	3
CSC-106	Object Oriented Programming LAB	1	0	4	3
CSC-107	LINUX LAB	2	0	4	4
CSC-108	Communication Skills	2	0	0	2
		Total_Credits			**24**

SEMESTER - 2

Course_Name		Contact_Hours/week			Credits
		L	T	P	
CSC-201	Web Development	3	0	0	3
CSC-202	Database Management Systems	3	0	0	3
CSC-203	Mathematics for Computer Science	4	0	0	4
CSC-204	Software Design & Engineering LAB	0	2	4	4
CSC-205	Web Development LAB	1	0	4	3
CSC-206	Database Management Systems LAB	1	0	4	3
	Elective - 1	4	0	0	4
		Total Credits			**24**

SEMESTER - 3					
Course_Name		**Contact_Hours/week**			**Credits**
		L	**T**	**P**	
CSC-301	Machine Learning	3	0	0	3
CSC-302	Modern Development Platforms	3	0	0	3
CSC-303	Machine Learning LAB	1	0	4	3
CSC-304	Modern Development Platforms LAB	1	0	4	3
	Elective - 2	4	0	0	4
	Elective - 3	4	0	0	4
	Elective - 4	4	0	0	4
		Total_Credits			**24**

SEMESTER - 4		
Course_Name		**Credits**
Industry Internship/ Project		24
	Total_Credits	**24**

Total Credits **72+24 = 96**

(Back to Index) (Back to Agenda)

Annexure V

MCA SEMESTER 1 SYLLABUS

1. CS101 Data Structures and Algorithms
2. CS102 Object Oriented Concepts
3. CS103 Operating Systems
4. CS104 Internet Technologies
5. CS105 Data Structures and Algorithms Lab
6. CS106 Object Oriented Programming Lab
7. CS107 Linux Lab
8. CS108 Communication Skills../../../../CST/Desktop/BoS2year/AfterBOS/CSC-106 Object Oriented Programming Lab.pdf

(Back to Index) **(Back to Agenda)**

Programme: MCA
Course Code: CSC-101 **Title of Course:** Data Structures & Algorithms
Number of Credits: 3 (3L-0T-0P) **Contact Hours:** 36 hours (36L-0T-0P)
Effective from AY: 2021-22

Prerequisites for the course	Program Prerequisites	
Objectives	The aim of the course is to emphasize the importance of data structures in implementing efficient algorithms. It provides an exposure to various algorithm design techniques and an introduction to algorithm analysis.	
Content	**Revision of Programming & Data Structures** Problem solving, Data Types: Primitive and User Defined Selection Constructs, Repetition Constructs, Recursion Pointers Algorithm Representation: - Pseudocode and flowcharts Three level Approach Abstract Data Types (ADTs) Basic Linear Data Structures (LinkedList, Stack, Queue)	6 hours
	Algorithm Analysis Analysis of Algorithms Algorithm Complexity: Space and Time Cases of Complexity: Best, Worst and Average Growth of Functions: Asymptotic Notation	3 hours
	Advanced Linear Data Structures Variants of Linked List and its applications (e.g. Polynomial addition, Sparse matrices) Applications of stacks (e.g. Infix-to-Postfix conversion, Evaluating Postfix Expressions, Bracket Matching) Variants of Queue and Applications	5 hours
	Nonlinear Data Structures: Trees: Binary Search Trees, AVL Trees, B-trees & variants. Tree Traversal Algorithms Heaps and its applications (e.g. implementation of Priority Queue) Graph: Adjacency Matrix and Adjacency List Representations Graph Traversal Algorithms: Breadth First Search and Depth First Search	12 hours
	Divide & Conquer Strategy Algorithms based on Divide and Conquer Strategy: Sorting Algorithms (QuickSort, MergeSort) Binary Search	3 hours
	Greedy Algorithms 1. Huffman Coding Algorithm	4 hours

	2. Minimum Cost Spanning Tree (Prim's, Kruskal's) 3. Single Source Shortest Path (Dijkstra's)	
	Dynamic Programming Coin Change Problem Longest Common Subsequence All-pair shortest Path (floyd-warshall)	3 hours
Pedagogy	● Lectures/Tutorials/Assignments/Quizzes ● Each data structure should be explained along with implementation of its ADT, its applications and complexity	
References/ Readings	1. Horowitz, Ellis, Sartaj Sahni, and Susan Anderson-Freed. "Fundamentals of data structures in C" WH Freeman & Co., Latest Edition. 2. Thomas H. Cormen, Charles E. Leiserson, et al "Introduction to Algorithms", Latest Edition 3. Allen, Weiss Mark. Data structures and algorithm analysis in C. Pearson Education India, Latest Edition. 4. Dasgupta, Papadimitriou, and Vazirani,Algorithms, by McGraw-Hill. 5. Jeri R. Hanly and Eliot B. Koffman "Problem Solving and Program Design in C" Pearson Education, VII Edition, 2012 6. R.G.Dromey "How to Solve it by Computer ", PHI , Latest Edition	
Learning Outcomes	Upon successful completion of the course, a student will be able to ● Implement common data structures such as lists, stacks, queues, graphs, and binary trees for solving programming problems. ● Identify and use appropriate data structures in the context of solution to a given problem. ● Be able to analyze the complexity of a given algorithm	

(Back to Index) (Back to Agenda)

Programme: MCA
Course code: CSC-102 **Title of course:** Object Oriented Concepts
Number of credits: 2 (2L-0T-0P) **Total contact hours:** 24 hours (24L-0T-0P)
Effective from AY: 2021-22

Prerequisites for the course	Program Pre Requisites	
Objectives	Aim of this course is to introduce the learner to the object oriented paradigm.	
Content	**Classes and objects** Programming paradigm; procedural to object oriented Class; attributes & methods; classes as modules & types; uniform type system, wrapper type classes Object; object references; objects instantiation & interaction; constructor & destructor; pass-by-reference & pass-by-value Object copying & cloning; composite objects Static & non-static members Enumeration & Annotations	7 hours
	Object oriented principles Encapsulation Inheritance; types of inheritance; diamond problem Abstraction; virtual methods Polymorphism; overloading and overriding	6 hours
	Object oriented features Interfaces Access modifiers Errors & Exceptions; user-defined exceptions Collections Anonymous & Inner classes Type parametric polymorphism (e.g. Generics in Java & Templates in C++)	6 hours
	Advanced features Persistence & Serialization; JSON User packages & custom libraries; reflection Predicates & streams Lambda functions	5 hours
Pedagogy	Hands-on assignments / tutorials / peer-teaching / flip classroom. Concepts can be explained using UML class diagrams.	
References/ Readings	**Main Reading** 1. Timothy Budd, "An Introduction to Object Oriented Programming", Pearson Education, 3rd Edition	

	2. Brett D. McLaughlin, Gary Pollice & David West, "Head First Object-Oriented Analysis Design", O'Reilly 3. Ken Arnold, James Gosling, David Holmes, "The Java Programming Language", Addison-Wesley Professional 4. Stanley Lippman, "C++ Primer", Addison Wesley 5. Cay S. Horstmann, "Core Java Volume I—Fundamentals", Pearson 6. Herbert Schildt, "Java: The Complete Reference", Oracle Press 7. Joshua Bloch, "Effective Java", Addison Wesley 8. Kathy Sierra & Bert Bates, "Head First Java", O'Reilly 9. Bjarne Stroustroup, "The C++ Programming Language", Addison Wesley 10. https://www.tutorialspoint.com/java/index.htm	
Learning Outcomes	1. Learner will appreciate mapping real-world scenarios in the object-oriented world 2. Learner will understand object-oriented principles 3. Learner will be able to design object oriented softwares 4. Learner will be able to analyse	

(Back to Index) (Back to Agenda)

Programme: MCA
Course code: CSC-103 **Title of course:** Operating Systems
Number of credits: 4(4L-0T-0P) **Total contact hours:** 48 (48L-0T-0P)
Effective from AY: 2021-22

Prerequisites for the course	Programme Prerequisites	
Objectives	This course focuses on the principles and understanding of the functionality of an operating system and evaluates their trade-off in various environments.	
Content	**Introduction and Systems Structures** Computing Environments, Operating-systems Services, System Calls, System Programs, Virtual Machines, monolithic and micro kernel architectures	4 hours
	Process Management Process-Concept and states, Process Creation and Control, Scheduling Criteria, Scheduling Algorithms, MultiLevel Queues., Multiple-processor scheduling, real time CPU scheduling	6 hours
	Threads Motivation and Challenges , Multithreading Models, Threading Issues, Thread libraries, Thread scheduling	4 hours
	Process Synchronization Cooperating processes and Race Conditions, The critical-section problem, Peterson's solution, mutex locks, Synchronization Hardware, Semaphores and their Implementation, Classic problems of synchronization	6 hours
	Inter process Communication, Overview of IPC, Examples of IPC Systems, Communication in Client Server Systems.	3 hours
	Deadlocks System Model, Deadlock characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery From Deadlock	4 hours
	Memory Management Hardware Support, Address Binding, Swapping ,Contiguous Memory Allocation, Fragmentation, memory Protection, Paging , Structure of the page table , Segmentation, Example: Intel architecture	6 hours
	Virtual-Memory Management	6 hours

	Background, Demand Paging, Copy-on-write, Page Replacement algorithms, Allocation of Frames, Thrashing, Allocating Kernel Memory	
	File System File Concept, Access Methods, Directory Structure, File-system mounting, File sharing, Protection ; Virtual file systems, Implementing File Systems, Directory implementation, Allocation Methods, Free-space Management, Efficiency and performance, Recovery, Log-structured file systems	6 hours
	Secondary-storage Structure Overview of Mass-storage Structure ,Disk Structure, Disk Attachment ,Disk Scheduling ,Disk Management, Swap-Space Management	3 hours
Pedagogy	lectures/ tutorials/assignments/class presentations and debates/peer reviews/self-study.	
References/ Readings	**Main Reading** 1. Silberschatz ,Galvin and Gagne , Operating systems Principles – 8th edition or Later(Wiley Asia Student Edition) 2. Deitel H.M., "An Introduction to Operating Systems", Addison Wesley Publishers Company, Latest Edition 3. Milenkovic M., "Operating Systems : Concepts and Design", McGraw Hill International Edition Computer Science series ; Latest Edition 4. Tanenbaum A. S., Modern Operating Systems", Prentice Hall of India Pvt. Ltd., Latest Edition 5. Operating Systems – a modern perspective - Gary Nutt , Addison Wesley, Latest Edition	
Learning Outcomes	1. To understand the services provided by and the design of an operating system. 2. To understand the structure and organization of the file system. 3. To understand what a process is and how processes are synchronized and scheduled. 4. To understand different approaches to memory management. 5. Students should be able to understand the implementation and use of system calls for managing processes, memory and the file system. 6. Students should understand the data structures and algorithms used to implement an OS. 7. Evaluate operating system implementations	

(Back to Index) (Back to Agenda)

Programme: M.C.A
Course Code: CSC-104 **Title of the Course:** Internet Technologies
Number of Credits: 3 (3L-0T-0P) **Contact Hours:** 36 hours (36L-0T-0P)
Effective from AY: 2021-22

Prerequisites for the course:	Programme requisites	
Objectives:	The objective to introduce the TCP/IP architecture and allied protocols of the Internet by following a top-down approach.	
Content:	**Computer Networks and the Internet:** Networking and Inter-networks, Internetworking devices, Internet: Network edge and Network core. TCP/IP protocol stack: Protocol stack, Connection oriented, connectionless services, Packet switching, circuit switching, Delay, Loss, and Throughput in Packet-Switched Networks.	6 hours
	Application layer: Principles of Application Layer Protocols, the Web and HTTP, MIME, mail access protocols, DNS, Peer to Peer Applications.	8 hours
	Transport layer: Transport-layer services, Multiplexing and demultiplexing, UDP protocol, Principles of reliable data transfer, Connection oriented transport - TCP protocol, Principles of congestion control, TCP congestion control.	6 hours
	Network layer: Packet switching: virtual circuit & datagram networks, The Internet Protocol (IP): Forwarding and Addressing in the Internet, route aggregation, subnetting, CIDR, IP datagram, fragmentation, NAT, DHCP, ICMP. Routing protocols: shortest path, link state routing algorithm, distance vector routing. Internet routing: Autonomous Systems (AS), RIP, OSPF, BGP. Address Resolution Protocol (ARP) and RARP.	10 hours
	Internet Security protocols Basic cryptography concepts, Secure Socket Layer (SSL), Internet Security Protocol (IPSec), Virtual Private Network (VPN).	6 hours
Pedagogy:	lectures/ tutorials/assignments/self-study	
References/Readings	1. Forouzan, Behrouz A., and Firouz Mosharraf. "Computer networks: a top-down approach". McGraw-Hill, 2012. 2. Andrew S. Tanenbaum., "Computer Networks", (5th Edition) Prentice Hall of India. 3. James F. Kurose, Keith W. Ross, "Computer Networking: A Top-Down Approach" Pearson, Sixth Edition 2017.	
Learning Outcomes	After completion of this course, students will be able to ● Have a good understanding of layered communication architecture (TCP/IP) and knowledge of some of the important networking protocols ● Understand the concepts of reliable data transfer and how TCP implements these concepts.	

	• Basic knowledge of the routing algorithms. (Back to Index) (Back to Agenda)

Programme: MCA
Course Code:CSC-105 **Title of Course:** Data Structures & Algorithms Lab
Number of Credits: 3 (1L-0T-2P) **Contact Hours:** 60 hours (12L-0T-48P)
Effective from AY: 2021-22

Prerequisites for the course	Programme Prerequisites	
Objectives	To develop skills to design and implement linear and nonlinear data structures and to identify the most appropriate data structure for solving a real world problem.	
Content	**Lab Assignments may be based on the following** Advanced Linear Data Structures Infix-to-Postfix conversion, Evaluating Postfix Expressions, Bracket Matching Non-linear data structures Binary Trees Tree Traversal Algorithms Binary Search Trees Heap Priority Queue using Heap Heap Sort Graph implementation using Adjacency list and matrix Graph Traversal Algorithms Divide & Conquer Strategy MergeSort QuickSort Binary Search Algorithm Greedy Algorithms Huffman Coding Algorithm Prims' and Kruskal's Algorithm Dijkstra's Algorithm Dynamic Programming Coin Change Problem Longest Common Subsequence Floyd-Warshall Algorithm A Mini Project	2L + 8P 4L+16P 2L+8P 2L+8P 2L+8P
Pedagogy	Programming assignments/ discussions/ self-review/ peer-review/ testing of code/ debugging of code/ projects	

References/ Readings	1. Horowitz, Ellis, Sartaj Sahni, and Susan Anderson-Freed. "Fundamentals of data structures in C" WH Freeman & Co., Latest edition. 2. Thomas H. Cormen, Charles E. Leiserson, et al "Introduction to Algorithms", Latest Edition 3. Allen, Weiss Mark. "Data structures and algorithm analysis in C." Pearson Education India, Latest Edition. 4. Dasgupta, Papadimitriou, and Vazirani, "Algorithms" McGraw-Hill. 2017	
Learning Outcomes	Upon successful completion of the course, a student will be able to ● Implement common data structures such as lists, stacks, queues, graphs, and binary trees for solving programming problems. ● Identify and use appropriate data structures in the context of solution to a given problem.	

(Back to Index) (Back to Agenda)

Programme: MCA
Course code: CSC-106 **Title of course:** Object Oriented Programming Lab
Number of credits: 3 (1L-0T-4P) **Total contact hours:** 60 hours (12L-0T-48P)
Effective from AY: 2021-22

Prerequisites for the course	Program Prerequisites	
Objectives	To impart programming skills using object oriented paradigms.	
Content	**Lab assignments using Java/C++/C#** Classes and objects Class, object, attributes & methods; classes as modules & types; uniform type system, wrapper type classes Object; object references; objects instantiation & interaction; constructor & destructor; pass-by-reference & pass-by-value Object copying & cloning; composite objects Static & non-static members Enumeration & Annotations Object oriented principles Encapsulation Inheritance; types of inheritance; diamond problem Abstraction; virtual methods Polymorphism; overloading and overriding Object oriented features Interfaces Access modifiers Errors & Exceptions; user-defined exceptions Collections Anonymous & Inner classes Type parametric polymorphism Advanced features Persistence & Serialization; JSON User packages & custom libraries; reflection Predicates & streams Lambda functions Mini-Project	3L+12P 3L+12P 3L+12P 3L+12P
Pedagogy	Hands-on assignments / pair programming / group project/ git project management.	
References/ Readings	**Main Reading** 1. Timothy Budd, "An Introduction to Object Oriented Programming", Pearson Education, Latest Edition. 2. Brett D. McLaughlin, Gary Pollice & David West, "Head First Object-Oriented Analysis Design", O'Reilly, Latest	

	Edition. 3. Ken Arnold, James Gosling, David Holmes, "The Java Programming Language", Addison-Wesley Professional, Latest Edition 4. Stanley Lippman, "C++ Primer", Addison Wesley, 2012 5. Cay S. Horstmann, "Core Java Volume I—Fundamentals", Pearson, 2018 6. Herbert Schildt, "Java: The Complete Reference", Oracle Press, latest edition 7. Joshua Bloch, "Effective Java", Addison Wesley 8. Kathy Sierra & Bert Bates, "Head First Java", O'Reilly, 2012 9. Bjarne Stroustroup, "The C++ Programming Language", Addison Wesley, Latest Edition 10. https://www.tutorialspoint.com/java/index.htm	
Learning Outcomes	1. Learner will be able to write good object oriented code 2. Learner will understand object-oriented principles 3. Learner will be able to design object oriented softwares	

(Back to Index) (Back to Agenda)

Programme: M.C.A
Course Code: CSC-107 **Title of the Course:** LINUX Lab
Number of Credits: 4 (2L-0T-2P) **Contact hours:** 72 hours (24L-0T-48P)
Effective from AY: 2021-22

Prerequisites for the course:	Program Prerequisites	
Objectives:	The objective is to introduce students to the Linux operating system environment and provide a knowledge of basic Linux commands and shell scripting and system call API.	
Content:	**LINUX Environment** Linux Installation and disk partitioning. Shell, Linux commands, Internal and External Commands, using the documentation/manual, users in Linux: user id, effective user id, use of commands su, sudo, id Basic commands: echo, who, whoami, date, cal, ls, passwd, history, shutdown. Input and output redirection operators (<,<<, >, >>)	3L + 3P
	The Linux File System, File and Directory management Structure of LINUX file system. Parent-child relationship. Concept of Home directory, current working directory and referring to home directory. Special Files: . and .. Absolute and relative pathnames. Use of PATH variable, Use of command: mkdir, rmdir, pwd, ls and cd. Use of file management commands: nano, touch, cat, cp, mv and rm. FIND command: Searching for a file using find, Finding List of files and directories. Concept of hard disk partitions, file system, Superblock and Inodes, General structure of Linux inode. use of stat command. Analysing the output of ls -l command. File type and permission. Use of chmod command. File ownership: Changing ownership using chown and chgrp commands. Modification and access times. Default file and directory permissions. Use of umask command. Concept of symbolic links. Hard and soft links. Use of ln command to create hard and soft links. Use of commands du, df, tar, zip, gzip, type, which	4L + 8P
	Filters: File commands- sort, wc, uniq, comm, cmp, diff, pg, tail, head, less, and more , Cut and Paste command Shells' sequence of interpretation of a command; Connecting commands with pipes Regular expressions: grep & sed command AWK script: Selection criteria and action- The BEGIN and END sections,	6L + 8P

	Splitting a line into fields and using printf. Getline function and reading input from files. Writing output to file and pipes. Awk system variables. Using regular expressions. Relational and Boolean operations. Command line parameters and environment variables. Programming constructs: if, for, while.	
	Process Management Concept of UNIX process. Role of init in process creation. Process ID and exit status of a process. Displaying process attributes using ps command, Killing processes, foreground and background processes. Use of commands job, fg, bg	1L + 2P
	Package management: Installing & removing packages	1L+1P
	Shell Script Shell scripts and execution methods. The dot command, Interactive and Non Interactive execution. Use of export command, Aliases and command history. Shell variables, Special variables, Built-in shell parameters. Command line arguments. Escaping and quoting. Difference between single and double quotes. Command substitution, brace and tilde expansion, I/O using read and echo. Escape sequences, 'test' command, arithmetic expressions, operators, Control flow: For, If, While, Case. Shell functions, error handling, debugging.	4L + 8P
	System programming Introduction to system programming, System calls and library functions. **Files and Directory system calls** List of sample programs 1. Write a program to implement the functionality of Linux command *touch* 2. Write a program to implement the functionality of Linux command *cat* *3.* Write a program to implement the functionality of Linux command *ls* 4. Write a program to redirect the output of all the printf statements to a user file using dup system call. 5. Write a program to read the standard input from a user file using dup system call. 6. Write a program to implement the functionality of Linux command *chmod* 7. Write two programs : one called parent.c, the other called child.c. The parent program reads two integers from the keyboard and arithmetic operator (+, -, *, /). The read	5L + 18P

	information is transmitted to a child process. After the child process finishes the operation, it transmits the result to the parent process. The parent process prints the result on the screen. 8. Write a c program namely "parent.c", which reads the processes along with their burst time (bt) and saves it in a file. Using fork, create a child process namely fcfs.c, which takes the filename containing process information as a parameter from the parent. The child process task is to calculate the average waiting time using the FCFS scheduling algorithm.	
Pedagogy:	Practical/ tutorials/assignments/self-study	
References/Readings	1. Unix Concepts and Applications – Sumitaba Das, Tata MacGraw Hill. 2. Unix and Shell Programming – Graham Glass and King Ables Pearson Education 3. C and Unix Programming – Kerningham and Pike, Prentice Hall 4. UNIX man pages	
Learning Outcomes	Upon completion of this course, the student will be able to: 1. Run various LINUX commands 2. Write shell script on LINUX OS. 3. Use various advanced LINUX tools such as grep, SED and AWK	

(Back to Index) (Back to Agenda)

Programme: M.C.A
Course Code: CSC-108
Title of the Course: Communication Skills
Number of Credits: 2 (2L-0T-0P)
Contact Hours: 24 hours (24L-0T-0P)
Effective from AY: 2021-22

Prerequisites for the course:	Programme requisites	
Objectives:	To introduce essentials of effective communication in different contexts	
Content:	**Oral Communication** Difference between formal and informal communication, importance of non verbal communication, skills required for effective communication, Public Speaking Skills.	4 hours
	Written Communication Writing cover letters, Resumés/CVs/Biodata, Letters of Invitation, Report Writing	4 hours
	Content creation Creating content for the website, creating profiles, creating content for brochures of events.	4 hours
	Multimedia and E-Correspondence Conducting Research before presentation, Making PowerPoint Presentation effective (visual), Communication during PowerPoint Presentation, Email etiquette (components, formats, attachments, content and language), Maintaining social media presence.	6 hours
	Preparing for Interview Types (personal, telephonic, online), Techniques of answering interviews, Participating in group discussions.	4 hours
	Allied Communication Effective Reading techniques, analyzing feedback and giving inputs.	2 hours
Pedagogy:	Lectures/ tutorials/laboratory work/ field work/outreach activities/ project work/ vocational training/viva/ seminars/ term papers/assignments/presentations/ self-study/ Case Studies etc. or a combination of some of these. Sessions shall be interactive in nature to enable peer group learning.	
References/Readings	1. Kelly M. Quintanilla and Shawn T. Wahl, "Business and Professional Communication " Sage Publications, 2018, 2. Anjanee Sethi ,Bhavna Adhikari, "Effective Business Communication " Tata MacGraw Hill Education, India. 2009; 3. Nido Qubein, "How to be a Great Communicator in Person, On Paper, and on Podium" Viva Books, India. 2008; 4. Stanton, Nicky. "Mastering Communication", (5th Edition), Macmillan, 2009. 5. Dalmar, Fisher. "Communication in Organisation", West Pub, 1993. 6. Kilian, Crawford. "Writing for the Web. Self-Counsel Press", Fifth edition, 2015. 7. Kallos, Judith. "Email Etiquette Made Easy", Lulu.com. 2007.	
Learning Outcomes	The participant will be able to facilitate interpersonal Communication, participate in group discussions, and to write effectively.	

(Back to Index) (Back to Agenda)

Annexure VI

List of Electives (4 credits each) for the MCA programme

Python Programming
Ethical Hacking
Cryptography
Cyber Security
Android App Development
Internet of Things
System Administration
Data Mining & Warehousing
Data Analytics
Natural Language Processing
High Performance Computing
Programming Paradigms
UNIX Programming

(Back to Index) (Back to Agenda)

Annexure VII

Ordinance	Existing Ordinance	Revised Ordinance	Justification
OB-19A.2	Eligibility and Admission	Eligibility and Admission	Eligibility and Admission
	1. To be eligible for admission to the three year, six semester, full time course leading to the degree of "Master of Computer Applications" (MCA), a candidate shall have: (i) A Graduate Degree in any discipline with at least 55% aggregate marks or equivalent grade at the first degree examination, and 50% aggregate marks or equivalent grade for OBC, ST, SC, candidates. (ii) Mathematics as one of the subjects at HSSCE (10+2) Science, or its equivalence in syllanus content. 2. The admission to MCA program will be strictly based on the Ranking obtained in the Entrance Test as decided and notified from time to time.\ 3. Eligibility and Admission to Semester II, III and IV Students shall be eligible for admission to the next higher Semester and be allowed to keep terms, irrespective of the backlog of earlier Semester(s). 4. Eligibility and Admission to Semester V Students shall be eligible for admission to Semester V, only if they do not have a backlog of more than four	To be eligible for admission to the 2 year program leading to the degree of "Master of Computer Applications" (MCA), a candidate shall have: a. (i) Passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent degree. OR (ii) Passed graduation in a non Computer Science discipline with Mathematics at 10+2 level or at Graduation level (with additional bridge courses). b. Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying examination c. The admission to MCA program will be strictly based on the merit list, prepared via the Aptitude Entrance Test, which will be notified from time to time.	In compliance with the AICTE (F. No. AICTE/AB/MCA/2020-21) dated 3.07.2020 The MCA program at Goa University have, in the past, received applications from candidates from B.E Electronics and Mechanical. These candidates would not be eligible if the clause for non Computer Science graduates was followed in toto. The recommended clause allows applicants from other streams too like BBA/BE – Electronics,Civil,Mechanical etc. All non Computer Science graduate applicants will need to undergo the Bridge course after qualifying at the Aptitude Test.

	Courses from the preceding Semester(s). 5. Eligibility and Admission to Semester VI students shall be eligible to take internship in the Semester VI, only on passing in all papers till Semester III.		
OB-19A.4	Instructional Scheme	Instructional Scheme	Instructional Scheme
	Instructional scheme for the MCA programme is based on a system of integrated units called Courses. Semester-I to Semester-V shall have 4 Theory Courses and 2 Lab Courses. Semester VI shall be exclusively dedicated to project / training. tudents to be allowed to take internship in the Sem-VI, only on passing in all papers till Sem-III.	Instructional scheme for the MCA programme is based on a system of integrated units called Courses. Semester-I to Semester-III shall have Theory Courses and Lab Courses. Semester IV shall be exclusively dedicated to Industry Internship/ Project.	For the 2 year program
OB-19A.4.1	Course Credits	Course Credits	Course Credits
	Each course will be of 100 marks and will have credits depending upon number of contact hours per week. The project will have no credits associated with it. The candidate may be permitted to opt for any Course from any Post Graduate department of the University as an elective. Candidate may earn a maximum of 16 credits by choosing electives from other departments.	Each credit of the course will be of 25 marks and will have credits depending upon number of contact hours per week. The Sem IV industry internship/ project shall have 24 credits. However the grade obtained by the student for the Internship/project shall not be considered for computing the CPI. However, the grade obtained will be considered in determining the lass obtained by the student (OB 19A.8.1) The candidate may be permitted to opt for any Course from any Post Graduate department of the University as an elective. Candidate may earn a maximum of 16 credits by choosing electives from	As papers have varying credits, the associated marks for the paper will differ. Marks per credit will determine the total marks assigned to the paper. The credits associated with internship are indicative of the number of hours required to be spent by a candidate in the industry. Industry requires a minimum of 40hrs/week. Students need to spend at least 14weeks in the Industry 24credits*24hours =576/40 =14.4weeks. Work assigned to students in the Industry are varied and cannot be monitored by the programme. Grades obtained by the student thus cannot be standardized and

		other departments.	should not be considered towards the CPI(which is the cumulative academic performance of candidates within a common environment)
OB19A.4.3	Contact Hours	Contact Hours	Contact Hours
	The total number of Lectures hours, Tutorials hours and Practical hours. Minimum of 45 contact hours are recommended for a 4 credit course, with 4 contact hours per week. One credit is equivalent to 15 contact hours in a semester	One Credit of a Theory Course shall be equivalent to 12 contact hours of learning activities including lecture, group discussion, seminar, problem solving, tutorial, assessment and such others. One Credit of a Practical Course shall be equivalent to 24 clock hours of laboratory or its equivalent.	In accordance with OA-18 A,2
OB19A.4.5	Course Coordinator	Course Coordinator	Course Coordinator
	In case of courses taught by Visiting Faculty, one faculty member from the department/College shall be associated with the course as course-co-ordinator	In case of courses taught by Visiting Faculty, one faculty member from the school shall be associated with the course as course-coordinator	The MCA program is now under the Goa Business School
OB19A.4.8 (c)	Service Learning	Service Learning	Service Learning
	The candidate may earn additional credits by undertaking community-based projects during the semester or during summer vacation. These projects would be defined in consultation with faculty and approved by Departmental Council. A project shall not carry more than 4 credits	The candidate may earn additional credits by undertaking community-based projects during the semester or during summer vacation. These projects would be defined in consultation with faculty and approved by Discipline Level Committee. A project shall not carry more than 4 credits.	Department Council is changed to Discipline Level Committee
OB19A.5.2	Scheme of Evaluation	Scheme of Evaluation	Scheme of Evaluation
	For a theory course, in semester evaluation is a continuous assessment worth 60 marks. At least 40 marks of the in-semester evaluation will be graded through one or more class tests. The remaining could be evaluated through quizzes, assignments etc.	For a theory course, in semester evaluation is a continuous assessment worth 60% marks. At least 40% marks of the in-semester evaluation will be graded through one or more class tests. The remaining could be evaluated through quizzes, assignments etc.	As credits associated for each paper vary, the percentage of marks are mentioned.

OB19A.5.3	Scheme of Evaluation	Scheme of Evaluation	Scheme of Evaluation
	For a theory course, the end-semester evaluation consists of an 'end-semester' examination of 40 marks of 2 hour duration conducted by the college/department. A candidate is eligible to appear for the end-semester examination if she/he has a minimum of 75% attendance in the theory course.	For a theory course, the end-semester evaluation consists of an 'end-semester' examination of 40% marks of maximum 2 hour duration. A candidate is eligible to appear for the end-semester examination if she/he has a minimum of 75% attendance in the theory course.	As credits associated for each paper vary, the percentage of marks are mentioned. Examination duration will also change accordingly.
OB19A.5.4	Scheme of Evaluation	Scheme of Evaluation	Scheme of Evaluation
	For a laboratory course, the assessment will be continuous with 60 marks for the in semester evaluation consisting of lab experiments, assignments etc. and 40 marks being reserved for the end-semester examination which includes a viva-voce and an online examination jointly conducted by an internal and external examiner. An external examiner is to be appointed from the panel of examiners approved according to the University ordinance OB-4. A candidate is eligible to appear for the end-semester examination if he has a minimum of 75% attendance in the laboratory course.	For a laboratory course, the assessment will be continuous with 60% marks for the in semester evaluation consisting of lab experiments, assignments etc. and 40% marks being reserved for the end-semester examination which includes a viva-voce and an online examination jointly conducted by an internal and external examiner. An external examiner is to be appointed from the panel of examiners approved according to the University ordinance OB-4. A candidate is eligible to appear for the end-semester examination if he has a minimum of 75% attendance in the laboratory course.	As credits associated for each paper vary, the percentage of marks are mentioned
OB19A.5.6	Scheme of Evaluation	Scheme of Evaluation	Scheme of Evaluation
	Project viva would be jointly conducted by an internal and an external examiner as per the guidelines of the project evaluation. An external examiner is to be appointed from the panel of examiners approved according to the University ordinance. Detailed guideline for project submission and evaluation is to be published in the	Semester IV internship Industry training/Project viva would be jointly conducted by an internal and an external examiner as per the guidelines of the project evaluation. An external examiner is to be appointed from the panel of examiners approved according to the University ordinance. Detailed guidelines for project submission and evaluation will	As there are mini projects in previous semesters, the ordinance refers to the Semester IV internship report.

	prospectus and copy of the same is made available in the department for reference.	be made available in the office for reference.	
OB19A.8.1	Award of Class	Award of Class	Award of Class
	Each semester grade report for the student shall carry his/her SPI. The final semester mark-sheet will indicate the CPI and the project performance. The final class for the MCA degree would be awarded as per the following scheme: ▪ Distinction: CPI equal to or greater than 8.5 and a minimum "Good" performance in the project ▪ First class: CPI equal to or greater than 6.5 but less than 8.5 and a minimum performance of "Satisfactory" in the project ▪ Second Class: CPI equal to or greater than 5.0 but less than 6.5 and a minimum performance of **"Pass"** in the Project ▪ Pass Class: CPI equal to or greater than 4.0 but less than 5.0 and a minimum performance of "Pass" in the project.	Each semester grade report for the student shall carry his/her SPI. The final semester mark-sheet will indicate the CPI and the project performance. The final class for the MCA degree would be awarded as per the following scheme: ▪ Distinction: CPI equal to or greater than 8.5 and a minimum "Very Good" performance in the project ▪ First class: CPI equal to or greater than 6.5 but less than 8.5 and a minimum performance of "Good" in the project ▪ Second Class: CPI equal to or greater than 5.0 but less than 6.5 and a minimum performance of "Satisfactory" in the Project ▪ Pass Class: CPI equal to or greater than 4.0 but less than 5.0 and a minimum performance of "**Satisfactory**" in the project.	The earlier ordinance did not reflect the correct grade scale assigned for the internship project which states (OB-19A.6.12 effective from 20th September, 2014) The final year Project shall carry only qualitative evaluation such as Excellent, Very Good, Good, Satisfactory, and Reject. **There was no "Pass" in the scale**. A student getting a Reject grade shall have to repeat the project
OA20.6.7	Goa Business School	Goa Business School	Goa Business School
	Admission to Semester I of the 3-year, six semester, full time programme leading to the degree of Master of Computer Applications, is open to any graduate with a Degree in any discipline provided that the candidate shall have offered Mathematics as one of the subjects at HSSCE (10+2) Science, or its equivalence in syllabus content	To be eligible for admission to the 2 year program leading to the degree of "Master of Computer Applications" (MCA), a candidate shall have: a. (i) Passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent degree. OR (ii) Passed graduation in a non Computer Science discipline with Mathematics at 10+2 level or at Graduation level (with	As proposed in OB 19A.2

		additional bridge courses). b. Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying examination c. The admission to MCA program will be strictly based on the merit list prepared via the Aptitude Entrance Test which will be notified from time to time.	
OB19A.7.3		**Proposed Ordinance**	
		To be qualified for the MCA degree, students are required to pass the test in the individual theory and laboratory components of the bridge course (40% marks to be obtained in theory and lab separately) which will be conducted by the programme. However, the marks obtained, although shown on the final year grade sheet, will not be added to the CPI/SPI.	The bridge course content are a mandatory requirement for the completion of the course. For graduate students in Computer Sc, the content of these courses would have been covered during the graduate program. However, this is not the case for non Computer Sc graduates. Under normal circumstances, these courses should be a prerequisite to the MCA.

(Back to Index) (Back to Agenda)