

CIRCULAR:-

A reference is invited to the Ordinances, Regulations and syllabi relating to the Five Year Integrated M.Sc. Degree Course in Computer Science (Semesters I and II) vide this office Circular No.UG/470 of 2009, dated 12th December, 2009 and the Head, University Department of Computer Science, the Directors/Heads, recognized Science Institutions concerned, the Principals of the affiliated Colleges in Science and the Professor-cum-Director, Institute of Distance and Open Learning are hereby informed that the recommendation made by the Ad-hoc Board of Studies in B.Sc. & M.Sc. Computer Science at its meeting held on 22nd May, 2009 has been accepted by the Academic Council at its meeting held on 14th July, 2009 vide item No.4.6 and subsequently approved by the Management Council at its meeting held on 21st August, 2009 vide item No.18 and that, in accordance therewith, the detailed syllabus for the Five Year Integrated M.Sc. Degree Course in Computer Science (Semesters I and II) and Regulation 8006 relating to the break-up of Fee Structure prescribed is as per Appendix and that the same has been brought into force with effect from the academic year 2009-2010.

MUMBAI-400 032
12th December, 2009

PRIN. K. VENKATARAMANI
REGISTRAR

To,

The Head, University Department of Computer Science, the Directors/Heads, recognized Science Institutions concerned, the Principals of the affiliated Colleges in Science and the Professor-cum-Director, Institute of Distance and Open Learning.

A.C./4.6/14.07.2009
M.C./18/21.08.2009

No.UG/469-A of 2009, MUMBAI-400 032 12th December, 2009

Copy forwarded with compliments for information to :

- 1) The Dean, Faculty of Science.
- 2) The Chairperson, Ad-hoc Board of Studies in B.Sc. & M.Sc. Computer Science.
- 3) The Controller of Examinations.
- 4) The Co-ordinator, University Computerization Center.


12/12/09

UNIVERSITY OF MUMBAI



**The Scheme, Syllabus & Fee Structure
for the First Year (Semester I & II)**

of

**Five Year Integrated
M.Sc degree programme**

in

Computer Science

**(To be introduced with effect from the academic
year 2009-2010)**

APPENDIX - II

COURSE STRUCTURE

(5-YR INTEGRATED M.Sc. COURSE IN COMPUTER SCIENCE)

Semester – I	Semester – II
Programming - I (Functional and Imperative paradigms)	Programming – II (Science of Programming)
Mathematics - I (Logic)	Mathematics - III (Algebra)
Mathematics - II (Calculus and Numerical Methods)	Mathematics - IV (Analysis)
Electronics - I (Analog and Digital)	Electronics - II (Analog and Digital)
Physics - I (Classical theory of electromagnetism)	Physics - II (Semiconductors and Devices)
Elective ARE-I	Elective ARE-II
Semester III	Semester IV
Programming practices – I	Programming practices – II
Programming principles – I	Programming principles – II
Digital signal processing	Embedded systems
Computer organization	Operating systems
Mathematics - V	Mathematics - VI
Elective ARE-III	Elective ARE-IV
Semester V	Semester VI
Algorithms – I	Algorithms – II
Databases – I	Databases – II
Networking – I	Networking – II
Software engineering-I	Software engineering-II
Simulation	Artificial Intelligence
Elective ARE-V	Elective ARE-VI

APPENDIX - III

SYLLABUS

(5-YR INTEGRATED M.Sc. COURSE IN COMPUTER SCIENCE – YEAR-I)

Preamble for: Syllabus of Year I of 5-year integrated M.Sc. in Computer Science

The enclosed syllabus is a non-standard arrangement of mathematics, physics, electronics and computing for the first year after 10+2. The content (not to mention the effort required by each student and teacher) for the integrated M.Sc. degree is quite different from that for the current 3-year B.Sc. + 2-year M.Sc. pattern. Our approach regarding the content as well as expectations from students needs justification. It is given below.

The subject matter is vast. What we teach in the first year becomes the basis for the second and later years' study. The issue before us is: "*what are the significant topics in the foundations of computing and electronics, and how to sequence them through 2 semesters of the first year?*" We have identified a few fundamental topics or areas. These are considered below, subject-wise in three parts.

First and foremost: Electronics is not computing, though it plays an important supporting role. Electronics comes in two parts—*analog* and *digital*—and it has foundations in physics as well as mathematics. (Computing foundations have no connection to physics. They are in mathematics.) We will finish discussing the physics syllabus, and next electronics, before moving to computing.

Two major areas are covered in physics: (1) the classical theory of electromagnetism, starting with Maxwell's equations, and (2) the theory of semiconductors and devices, starting with Bohr's model of the atom and the band gap theory for outer-shell electrons in a crystalline solid. Among the textbooks we recommend are: Jordan, *Electromagnetic Fields*, Jaspri Singh, *Semiconductor Theory* and Millman, *Electronic Devices and Circuits*. (Important note: Only the initial parts of the books will be studied. Hence they can realistically be covered in FY.)

In analog electronics, we cover mostly circuit theory. (Other core areas of electronics have been covered in physics.) The recommended book is M.E. van Valkenburg, *Network Analysis*. Again, only the initial part is covered, systematically, over two semesters. For a few topics we will use Millman, already recommended for physics.

Some readers will recognize that the textbooks we have recommended are used typically in the second year of electronics engineering, or even in postgraduate education. The critical reviewer can rightly ask: is this approach appropriate for FY, immediately after 10+2?

The answer to this doubt is: the enclosed FY syllabus is elementary, i.e., does not need prerequisite knowledge other than 10+2 competence in mathematics and physics. Note: "Elementary" does not always mean "easy". To be sure, the proposed syllabus is difficult. It is up to the students, teachers and administrators to implement it. Just now, the reviewer has to verify that the proposal is realistic. This "reality check" applies not just to physics and electronics, but to mathematics and computing too.

Second: Computing. Our particular approach to computing is much more conventional, starting with Haskell for functional programming, and followed by Python for imperative programming.

Note 1: The Haskell language and style of programming has been used in leading Universities across the world, for over a decade.

Note 2: Python is easy to learn, and is widely popular. It is the language both for Google's large "server farms" and the introductory programming course at Massachusetts Institute of Technology. It is also taught as a first language in hundreds of Universities worldwide.

Note 3: For production-quality programs and for teaching also, C is by far the most popular programming language over the last 2 or 3 decades. But it has serious deficiencies both for teaching and for programming. In 2009, for the student's first language of imperative programming, we submit that Python is a better choice.

Finally: Mathematics. It is divided into two parts: (1) algebra and logic, (2) calculus and analysis. The syllabus of algebra and logic is non-standard. It has been custom-made for the present 5-year study in computer science. Calculus and analysis is more conventional. It consists of a set of topics typically covered in first year Engineering Mathematics.

Syllabus: Programming

Semester 1: Paper 1 Programming - I (functional and imperative paradigms)

Aim: Introduce fundamental concepts: computational problems and their solution (algorithms and programs); functional and imperative paradigms, term-rewriting and state change models of computation; training students to “act like a machine” to compute the results, and to “act like a thinker” to formulate algorithmic solutions and check their correctness.

Unit 1: Lambda calculus: untyped lambda calculus as a programming language, lambda terms, term rewriting (beta reduction), normal vs applicative order reduction, (eager and lazy evaluation); fixed point theorem (Y combinator), conditional constructs and recursion; Church numerals as a case study; equational theory of lambda terms, inductive proofs over recursive structures; types in lambda calculus, type deduction.

Unit 2: Haskell: GHC, data structures tuple and list, list processing, data types

Unit 3: Imperative programming with flowcharts: Flowchart structure (directed graph with nodes and arcs); two kinds of nodes: (1) action, (2) test (for true/false); examples of simple computational problems and their flowchart solutions; goto (branch, jump); well-structured flowcharts and structured programming; semantics of compound imperative statements (if, while, for, etc.) expressed as flowcharts.

Unit 4: Imperative programming (preferably with Python): Interpreter / compiler (introduction to IDE), data types and operations, sets, tuples / records and lists / arrays, control structures, decomposition of an algorithm into procedures.

Unit 5: Imperative programming with the von Neumann machine model: instruction and data memories, assignment as data memory state change, instruction set, example of a simple RISC architecture (MIPS R 3000); instruction fetch-decode-execute cycle, flow of control, i.e., in-sequence and out-of-sequence next instruction; native machine language and assembly language, assembly language programming with the SPIM assembler for the MIPS instruction set.

Books for Programming - I:

1. Doets, Eijck, Haskell Road to Logic, Maths and Programming
2. Hutton, Programming in Haskell, Cambridge University Press
3. Dahl, Hoare and Dijkstra, Structured Programming
4. Convey, Gries and Zimmerman, A Primer on Pascal, Cambridge, Mass. Wintthrop
5. Lutz, Programming Python
6. Hankin, Introduction to Lambda Calculi for Computer Scientists

Books for additional readings

7. Hindley, Lambda-Calculus and Combinators, Cambridge University Press
8. MIPS Corp: MIPS R3000 Manual
9. Thompson, Haskell: The Craft of Functional Programming:
10. Hudak, The Haskell School of Expression

Semester 2 : Paper 1 **Programming - II (Science of programming)**

Aims are two-fold: (1) introduce the science of programming: understanding the semantics of a program and proving its correctness, application of equational reasoning for the verification of functional programs and of natural deduction for imperative programs. (2) More programming with Haskell and Python, with typical content for a second course in algorithms and data structures

Unit 1: Testing vs. proving the correctness of a program, formal vs. informal arguments about correctness; introduction to the correctness of functional programs and imperative programs.

Unit 2: Abstract data types in Haskell and Python.

Unit 3: Program semantics: introduction to operational semantics of functional programs, assertion-based approach to program verification. Proofs of functional programs: equational proofs, proof by induction, use of folds as universal constructs (to simplify and streamline proofs)

Unit 4: Proofs of imperative programs: assertions, Hoare-Floyd logic, semantics of assignment, selection structures and iterative structures, loop invariants and loop termination; weakest precondition approach and refinement calculus

Unit 5: Program derivation: from specification to programs, loop invariants and then the loops, introduction to refinement calculus, formal approach to program derivation.

Books for Programming - II:

1. Dromey, Program Derivation: the development of programs from specifications
2. Doets and van Eijck, The Haskell Road to Logic, Maths and Programming
3. Dahl, Hoare and Dijkstra, Structured Programming
4. Hutton, Programming in Haskell,
5. Back, Refinement Calculus
6. Manes, Predicate Transformer Semantics, Cambridge University Press

Books for additional readings

7. Dromey, Stepwise Program Derivation
8. Thompson, Haskell: The Craft of Functional Programming
9. Hudak, The Haskell School of Expression

Syllabus: Mathematics

Semester 1: Paper 2 Mathematics – I (Logic)

Aim: Give students a theoretical foundation for problem solving

Logic is to be taught in the first semester, in parallel with problem solving in Programming - I.

Unit 1. Fundamentals of arithmetic: introduction to numbers, fundamental laws of addition and multiplication (closure; associative, commutative, monotonic, distributive laws); logical foundations of arithmetic (Kant's approach, Poincare's formalism and proof by induction, Peano's axioms); extension of the notion of number, subtraction and division), negative numbers, fractions (rational numbers).

Unit 2. Fundamentals of logic: concept of proof, direct proof, indirect proofs—proof by contradiction, by counterexample, by equivalence (if and only if); mathematical induction

Unit 3. Propositional logic, predicate logic, equational reasoning: natural deduction for assertion-based program verification of imperative programs, term rewriting model and equational logic for verification of functional programs.

Books for Mathematics I:

1. Huth and Ryan, Logic in Computer Science modeling and reasoning about systems
2. Manin, A Course in Mathematical Logic (tr. from the Russian), Springer
3. Tarski, Introduction to Logic and to the Methodology of the Deductive Sciences, OUP
4. Kelly, The Essence of Logic
5. Gries and Schneider, A Logical Approach to Discrete Math
6. Yalgom, An Unusual Algebra, Mir Publications

Semester 2 : Paper 2 Mathematics – II (Algebra)

Aim: Learn the essentials of classical algebra needed for computer science.

Unit 1. Set theory: sets, relations, operations; semigroup, ring, field, group, lattice.

Unit 2. Polynomials and their roots, quadratic forms, polynomials in several unknowns, polynomials with rational coefficients, linear spaces, Euclidean spaces.

Unit 3. Linear algebra: systems of linear equations, solution by direct and iterative methods, matrices

Unit 4. Graph theory (in view of applications to data structures and other areas of computing)

Books for Mathematics II:

1. van der Waerden, Modern Algebra,
2. Fraleigh, A First Course in Abstract Algebra
3. Vinberg A Course in Algebra (tr. from the Russian), AMS
4. Artin, Galois Theory
5. Hill, Groups and Characters, Chapman and Hall/CRC
6. Kumaresan, Linear Algebra: A Geometric Approach, PHI
7. Singh, Zameeruddin, Modern Algebra, Vikas Publishing House

Semester 1 Paper 3 Mathematics – III (Calculus and Numerical Methods)

Aim: Learn the essentials of calculus and numerical analysis, needed for electronics and CS.

Unit 1. Real and complex numbers; functions of one and several variables; review of differential calculus; differential equations and their solution.

Unit 2. Integral calculus - review, line, surface and volume integrals; Green's, Stokes' and Gauss' thms.

Unit 3. Vector calculus: co-ordinate systems; differential operators gradient, divergence, curl

Unit 4. Finite differences, interpolation, curve fitting with polynomials and splines; errors in approximation.

Unit 5. Numerical differentiation, integration, solution of differential equations; round-off and accumulated errors; numerical linear algebra.

Books for Mathematics III:

1. Stewards, Calculus
2. Hamming, Numerical Methods for Scientists and Engineers
3. Courant and John, Introduction to Calculus and Analysis, vol.1 & 2, Springer
4. Bracewell, The Fourier Transform and Its Applications
5. Wartikar & Wartikar, Textbook of Applied Mathematics, vol. 1-3, Pune Vidyarthi Gruh Prakashan

Books for additional readings

6. Piskunov, Differential and Integral Calculus
7. Apostol, Calculus, vol. 1 & 2
8. Isaacson, Analysis of Numerical Methods
9. Kreyszig, Advanced Engineering Mathematics

Semester 2 Paper 3 Mathematics – IV (Analysis)

Aim: Learn the essentials of calculus and numerical analysis, needed for electronics and CS.

Unit 1. Sequences and series, convergence; mean value theorems.

Unit 2. Complex analysis: complex function theory, Cauchy-Riemann equations, integral representations of functions, analytic functions as power series

Unit 3. Fourier series; periodic functions, time and frequency domains; Fourier transform; convolution;

Unit 4: Laplace transform; continuous and discrete time domains, discrete Fourier transform; Z-transform.

Books for Mathematics IV:

1. Spivak, Calculus, Publish or Perish
2. Palka, Introduction to Complex Function Theory, Undergraduate Texts in Maths, Springer
3. Marsden and Hoffman, Basic Complex Analysis, Freeman
4. Courant and John, Introduction to Calculus and Analysis, vol.1 & 2, Springer

Books for additional readings

5. Lang, Undergraduate Analysis, Springer
6. Greene and Krantz, Function Theory of One Complex Variable, Wiley

Syllabus: Electronics

Semester 1 Paper 4 Electronics - I (Analog and Digital)

1. DC circuits: Resistance, voltage and current sources, resistances in series and parallel; Kirchhoff's laws, Thevenin's and Norton's theorems; capacitance and inductance V-I characteristics; behaviour of a capacitor/inductor in series/parallel with a resistor, transient response of R-C, R-L and R-L-C circuits.
2. AC circuits: Current and voltage sources; average and RMS values of sinusoidal functions; AC behaviour of resistance, capacitance, impedance; R-C, R-L and R-L-C circuits, resonance; bandwidth, Q factor; transient response; active and reactive power; three-phase circuits, star and delta connections and transformations.
3. Two-port networks: Z, Y, ABCD, h-parameters; condition for reciprocity and symmetry; 2 port network connections in series, parallel and cascaded. Small signal models of transistors, derivation of parameters from transistor characteristics.
4. Clocks; digital abstraction; registers and register state; combinational and sequential functions and their implementation. Digital logic design, hazards and races.
5. Number systems and codes: decimal, binary, octal, hexadecimal representation; error-detecting and error-correcting codes, shift registers.
6. Functional units: registers, multiplexers, decoders, ROM and RAM memories, glue logic, interconnection of functional units; using PLDs like FPGAs to implement digital circuits.

Books for Electronics - I:

1. M.E. van Valkenberg, Network Analysis, Pearson. (A classic book with a thorough introduction).
2. C. L Wadhwa, Network Analysis, New Age International
3. O'Malley, Basic Circuit Analysis (Schaum's Outline Series) Tata McGraw Hill.
4. Waverly, Digital Design, Prentice Hall

Semester 2 Paper 4 Electronics - II (Analog and Digital)

- Unit 1: Amplifiers: gain, bandwidth, linearity, efficiency, noise, dynamic range; slew rate, rise and setting time, overshoot; types; feedback and classification of feedback amplifiers; multistage amplifiers;.
- Unit 2: Operational amplifier principles and applications; oscillators, phase shift oscillators; types of harmonic and relaxation oscillators; filters: low-pass, high-pass, band-pass; filter design; active filters.
- Unit 3: Transistors: DC operating point, biasing, loadline, stability; transistor circuits
- Unit 4: Memory elements: flip-flops and latches, types of flip-flops: R-S, J-K, T, D; types of latches; monostable, astable and bistable multivibrators; applications in digital circuits.
- Unit 5: Design of small digital circuits: counters, timers, parity-generator, adder,
- Unit 6: Detailed study of the datasheets of about 10 ICs used in industry, interconnecting ICs to make a digital system.
- Unit 7: Finite state machines: basic theory and implementation; algorithmic state machines.

Books for Electronics - II:

1. Millman and Halkias, Electronic Devices and Circuits, Tata McGraw Hill
2. Lang and Agarwal, Foundations of Analog and Digital Electronic Circuits, Morgan Kaufmann
3. Grob(Dover), Basic Electronics

Syllabus: Physics

The general approach taken is of mathematical physics, i.e., modeling the behaviour of the physical world with equations, then trying to solve them analytically, or if that is not possible, numerically.

Semester 1 Paper 5 **Physics - I (Classical theory of electromagnetism)**

Unit 1: Coulomb's law, electric field; motion of a charged particle in an electric field; electric flux, Gauss' law; current, current density, conductors; work, energy and potential in an electrostatic field; Laplace's equation.

Unit 2: Magnetic field, Ampere's law, forces and torques in magnetic fields; displacement current and induced emf.

Unit 3: Foundations of electromagnetism: Maxwell's equations, boundary conditions, examples, solution by analytic or numerical methods.

Unit 4: Briefly: Electromagnetic waves, transmission lines, antennas: wave equations for perfectly conducting and dielectric media, lossless and lossy wave propagation; Poynting vector and theorem; electromagnetic radiation in free space from an energy source, types of antennas, amplitude and frequency modulation; transmission lines: types, parameters and equations; characteristic impedance, propagation constant, phase and group velocities, distortion; types of loading; impedance matching.

Books for Physics - I:

1. Jordan and Balmain, Electromagnetic Waves and Radiating Systems, Pearson
2. Feynman, Lectures in Physics vol. II (Electromagnetism), Narosa
3. Edminster, Electromagnetics (Schaum Series Outlines), Tata McGraw Hill

Semester 2 Paper 5 **Physics - II (Semiconductors and Devices)**

Unit 1. Bohr's model of the atom; insulators, conductors and semiconductors; intrinsic and extrinsic semiconductors; drift and diffusion currents; mass action law, continuity equation, Hall effect.

Unit 2. P-N diodes, energy band of the P-N junction; open-circuit, forward biased and reverse biased junctions; current components of a P-N diode, temperature dependence; transition and diffusion capacitance. Types of diode: avalanche, Zener, tunnel, LEDs, LCDs.

Unit 3. Bipolar junction transistors, principle of operation; common emitter, common base and common collector configurations; field effect transistors, common-source configuration, fabrication of integrated circuits.

Books for Physics - II:

1. Jasprit Singh, Semiconductor Devices, Wiley.
2. Millman and Halkias, Electronic Devices and Circuits, Tata McGraw Hill

R. 8006 FEE STRUCTURE FOR 5-Year Integrated M.Sc. Degree in Computer Science

FEE	AMOUNT (RUPEES)	Justification
Tuition Fee	11,000/-	Per Semester
Form and Prospectus fee	500/-	Per application form
Other fees/Extracurricular activities	250	Per year
University Examination Fees	600/-	Per paper
Mark Sheet	50/-	Per Semester
Laboratory Fees	11,000/-	---
Library	1000/-	---
Gymkhana	200/-	---
Admission processing fee	200/-	---
Vice Chancellors fund	200/-	---
Magazine	100/-	---
Identity card	50/-	---
Group insurance	40/-	---
Student welfare	50/-	---
University sports and cultural activity	30/-	---
Development fee	500/-	---
Utility	250/-	---
e- suvidha	50/-	---
e- charges	20/-	---
Disaster relief fund	10/-	---
Total	26,100/-	

Document verification	:	Rs 400/- wherever applicable
Refundable deposits	:	Rs 150/-
Caution money	:	Rs 400/-
Library deposit	:	Rs 400/-
Laboratory deposit	:	
Project fee and support for research	:	Rs.2500/- per semester
Publication	:	Rs 1000/-
Registration fee for M.Sc. Part-I	:	Rs.25/-
Registration form fee	:	Rs 250/-
Convocation fee for M.Sc.Part-II	:	

N.B.

Every year there will be a 5% increase in the tuition fess and the laboratory fees.
 Foreign students will have to pay times of the prescribed fees
 The fee collected in different heads be spent only for the specified purpose for which it has been collected.