# iMTech (Integrated M.Tech.) Curriculum

International Institute of Information Technology
Bangalore – 560100

# Contents

# 1 Introduction

This is the curriculum document the 5-year Integrated M.Tech. (iMTech) programme effective for IIITB students who have joined.

## Overall iMTech Programme Structure

| | |
|---|---|
| **Semester 1 (15 weeks)** | **18 credits**<br>• 6 common core courses |
| **Semester 2 (15 weeks)** | **20 credits**<br>• 6 common core courses |
| **Semester 3 (15 weeks)** | **20 credits**<br>• 4 common core courses<br>• 1 CSE / ECE core course |
| **Semester 4 (15 weeks)** | **23 credits ECE / 19 credits CSE**<br>• 2 common core courses<br>• 2 CSE / 3 ECE core course<br>• 1 Elective from Maths and Science Pool |
| **Semester 5 (15 weeks)** | **19 credits ECE / 23 credits CSE**<br>• 2 CSE / 3 ECE core courses<br>• 4 CSE / 2 ECE Elective courses |
| **Semester 6 (15 weeks)** | **20 credits**<br>• 5 Electives |
| **Semester 7 (15 weeks)** | **20 credits**<br>• 5 Electives |
| **Semester 8 (15 weeks)** | **20 credits**<br>• 5 Electives |
| **Semester 9 (15 weeks)** | **20 credits**<br>• Combination of Project and Courses |
| **Semester 10 (15 weeks)** | **20 credits**<br>• Thesis / Internship |
| **Total** | **200 credits CSE / 200 credits ECE** |

- **1 HSS Elective elective course needs to be satisfied by student as graduation requirement**. The elective course can be taken anytime from 5th semester to 9th semester as part of elective courses at student's convenience. Failure to do so, will make the graduation requirements unsatisfactory and student will not be allowed to timely graduate.

The course credits earned over a period 10 semesters are grouped into the following categories:

- Mathematics and Basic Sciences (MBS)
- Humanities and Social Sciences / Management (HSS)
- Engineering Core (EC)
- Systems (Sys)
- Programming (Prog)
- Branch Electives (for CSE and ECE, respectively) -- (BE)
- Electives
- Masters Project / Thesis

.

## 2 Branch and Specialization

The term “branch” as used in this document refers to the separate Integrated M.Tech. (iMTech) degrees that are going to be offered in Computer Science and Engineering (CSE) and Electronics & Communications Engineering (ECE), respectively.

The term “specialization” is used to refer to sub-areas in which a set of related electives are offered giving the student an opportunity to specialize in specific areas. Specializations based on existing elective courses have been suggested in the report. The curriculum offers students to do two specialization and the same will be reflected in students transcripts.

## 3 Categories and Levels

Courses have been classified into levels.

- Level 1 courses are undergraduate level courses.
- Level 2 and Level 3 courses are post graduate level courses, at basic and advanced levels, respectively.

It should be noted that the credit system takes into consideration the course levels also. The credit system for undergraduate level (Level 1) courses and graduate level (Level 2 and higher) courses is as shown in Table 1 below.

| | **Interaction Type** | **Interaction Time** | **# of credits** |
|---|---|---|---|
| Level 1 Courses | Lecture | 1 hour / week for a semester | 1 |
| | Tutorial | 1 hour / week for a semester | 1 |
| | Practical | 2 hours / week for a semester | 1 |
| Level 2 and Level 3 Courses | Lecture | 3 hours / week for a semester | 4 |
| | | 2 hours / week for a semester | 3 |
| | Practical | 2 hours / week for a semester | 1 |

**Table 1: Course Levels and Credits**

## 4 Category-wise Course List

This section contains the course list organized as per the course categories given earlier. Table 2 and Table 3 contain a summary of the overall distribution of courses across the various categories.

| Computer Science and Engineering (CSE) | | |
|---|---|---|
| **Heads** | **Credits** | **Percentages** |
| Programming | 14 | 7% |
| System | 16 | 8% |
| Core Engineering (CSE) | 22 | 11% |
| Branch Elective | 24 | 12% |
| Maths + Basic Sciences | 20 | 10% |
| Humanities | 16 | 8% |
| Other Electives | 48 | 24% |
| Combination of Courses and Masters Project in 9th semester | 20 | 10% |
| Internship/Thesis | 20 | 10% |
| Total | 200 | 100% |

**Table 2: CSE Course Distribution**

| Electronics and Communication Engineering (ECE) | | |
|---|---|---|
| **Heads** | **Credits** | **Percentages** |
| Programming | 14 | 7% |
| System | 16 | 8% |
| Core Engineering (ECE) | 30 | 15% |
| Branch Elective (ECE) | 24 | 12% |
| Maths + Basic Sciences | 20 | 10% |
| Humanities | 16 | 8% |
| Other Electives | 40 | 20% |
| Combination of Courses and Masters Project in 9th semester | 20 | 10% |
| Internship/Thesis | 20 | 10% |
| Total | 200 | 100% |

**Table 3: ECE Course Distribution**

## 4.1 Programming

The list of courses under the Basic Engineering Science and Skills is given in Table 4 below.

| Course Name | Credits | L:T:P:C |
|---|---|---|
| Programming in C | 2 | 1:0:2:2 |
| Programming in Python | 2 | 1:0:2:2 |
| Datastructures and Algorithms | 6 | 3:1:4:6 |
| Programming II | 4 | 3:0:2:4 |

**Table 4: Programming**

## 4.2 Mathematics and Basic Sciences

The list of courses under the Mathematics and Basic Sciences category is listed in Table 5 below. All these courses are Level 1 courses.

| Course Name | Credits | L:T:P:C |
|---|---|---|
| Mathematics - 1 | 4 | 3:1:0:4 |
| Mathematics - 2 | 4 | 3:1:0:4 |
| Mathematics - 3 | 4 | 3:1:0:4 |
| Physics - 1 | 4 | 3:0:2:4 |
| An elective in Basic Science and Maths Pool | 4 | 3:0:0:4 |

**Table 5: Mathematics and Basic Sciences**

## 4.3 Humanities and Social Sciences / Management (HSS/M)

The courses listed in Table 6 below are the core courses under the HSS/M category. All these courses are Level 1 courses.

| Course Name | Credits | L:T:P:C |
|---|---|---|
| Technical Communication | 2 | 2:0:0:2 |
| English | 2 | 2:0:0:2 |
| Economics | 4 | 3:1:0:4 |
| History of Ideas | 4 | 3:0:0:4 |
| An elective in HSS | 4 | 3:0:0:4 |

**Table 6: HSS/M**

## 4.4 Systems

Table 7 below contains other courses that are more general in nature. All these courses are Level 1 courses.

| Course Name | Credits | L:T:P:C |
|---|---|---|
| Digital Systems | 4 | 3:1:0:4 |
| Signals and Systems | 4 | 3:1:0:4 |
| Computer Network | 4 | 3:1:0:4 |
| Computer Architecture | 4 | 3:1:0:4 |
| Operating Systems | 4 | 3:0:2:4 |

**Table 7: Systems**

## 4.5 CSE Core

Table 8 below contains the list of courses that are mandatory for the CSE stream. All these courses are Level 1 courses.

| CSE Core Course Name | Credits | L:T:P:C |
|---|---|---|
| Discrete Mathematics | 4 | 3:1:0:4 |
| Design and Analysis of Algorithms | 3 | 3:0:0:3 |
| Introduction to Automata Theory and computability | 4 | 3:1:0:4 |
| Software Engineering | 4 | 3:0:2:3 |
| Database Systems | 4 | 3:0:2:4 |

**Table 8: CSE Core**

### 4.6 ECE Core

Table  below contains the list of core courses in ECE. All these courses are Level 1 courses.

| ECE Core Course Name | Credits | L:T:P:C |
|---|---|---|
| Basic Electronic Circuits | 4 | 2:0:2:4 |
| Analog Circuits | 4 | 3:0:2:4 |
| Principles of Communication Systems | 4 | 3:0:2:4 |
| Mobile Computing | 4 | 3:1:0:4 |
| Signal Processing | 3 | 3:0:0:0 |
| Control Theory | 3 | 3:0:0:0 |
| Digital Communication | 4 | 3:0:2:4 |

**Table 9: ECE Core Courses**

### 4.7 General

| Course Name | Credits | L:T:P:C |
|---|---|---|
| Physical Education | 0 | 0:4:0:0 |

**Table 8: Systems**

### 4.8 Branch Electives

Students are required to satisfy 6 courses of 4 credits as a part of Branch Elective (BE) requirement. CSE branch students need to take a total of 19 electives (76 credits) and ECE branch students need to take a total of 17 electives (68 credits) till 8 semester. The students can plan their Branch electives in this group of electives starting from 5th semester to 9th semester. The curriculum gives options for students to take additional courses in their 9th semester.  For CSE, the branch electives needs to be satisfied from courses listed in TSCD stream. For ECE, the branch electives needs to be satisfied from courses listed in VLSI and NC stream.

## 5 Specialization

iMTech students can earn specialization in specified areas provided they do atleast 5 electives in those areas. Specialization is optional for the students and is determined at the time of graduation based on the concentration of the electives chosen by the student during the programme. The students' specialization is recorded only in the transcript issued to the student. PE and RE courses will also be counted towards specialization. Dual specialization is encouraged for students. The courses, which fall, in more than one specialization streams will not be double counted for dual specializations. For obtaining dual specialization, the student has to complete 10 different elective courses falling in the two specialization, the student intends to earn. A maximum of 6 Project Elective (PE) and Reading Elective (RE) can be enrolled by the students by identifying and working with regular faculty at IIITB in a semester. The faculty will allocate a letter grade as per IIITB grading policy towards the PE/RE course. A maximum of 3 PE and 3 RE can be taken, with not more than 1 PE and 1 RE in a given semester. PE, RE will be counted towards specialization, however it will not be counted towards satisfaction of 6 Branch Electives. A minimum of 3 regular elective course is required to fill up 5 electives for a specialization. 5 courses in the form of PE/RE will not be counted for specialization.

Based on the electives that are being offered currently, following are the specializations that are available to the students belonging to the CSE and ECE stream:

- Theory and Systems for Computing and data (TSCD)
- Artificial Intelligence and Machine Learning (AIML)
- Networking and Communication (NC)
- VLSI Systems (VLSI)
- Digital Society (DT)

# 6 Masters Project / Thesis

Students have two options for meeting the Masters Project requirement:

1. (Option A) One semester (20 credits) with combination of courses and project during the 9th semester at IIITB followed by 6 month project work in the industry during the 10th semester (20 credits).

   OR
2. (Option B) Two semesters of Thesis work in the 9th and 10th semesters under the supervision of a faculty member

## 7 Course Sequencing for iMTech (CSE)

The course sequencing for the CSE branch is given in Table 9 below.

| Course Name | Credits | Course Category | Level |
|---|---|---|---|
| **SEMESTER 1** | **18** | | |
| Mathematics - 1 | 4 | Mathematics and Basic Sciences | Level 1 |
| Programming in C | 2 | Programming | Level 1 |
| Programming in Python | 2 | Programming | Level 1 |
| Digital Design | 4 | Systems | Level 1 |
| Physical Education 1 | 0 | Others | Level 1 |
| English | 2 | HSS/M Core | Level 1 |
| Economics | 4 | HSS/M Core | Level 1 |
| **SEMESTER 2** | **20** | | |
| Mathematics - 2 | 4 | Mathematics and Basic Sciences | Level 1 |
| Computer Architecture | 4 | Systems | Level 1 |
| Data Structures and Algorithms | 4 | Engineering Core | Level 1 |
| Data Structures Lab | 2 | Engineering Core | Level 1 |
| Computer Networks | 4 | Engineering Core | Level 1 |
| Technical Communication | 2 | HSS/M Core | Level 1 |
| Physical Education 2 | 0 | Others | Level 1 |
| **SEMESTER 3** | **20** | | |
| Mathematics - 3 | 4 | Mathematics and Basic Sciences | Level 1 |
| Programming II | 4 | Programming | Level 1 |
| Physics (Theory) | 3 | Mathematics and Basic Sciences | Level 1 |
| Physics (Lab) | 1 | Mathematics and Basic Sciences | Level 1 |
| Signals and Systems | 4 | Systems | Level 1 |
| Discrete Mathematics | 4 | CSE Core Engineering | Level 1 |
| **SEMESTER 4** | **19** | | |
| Operating Systems (Theory) | 3 | Systems | Level 1 |
| Operating Systems (Lab) | 1 | Systems | Level 1 |
| Design and Analysis of Algorithms | 3 | CSE Core Engineering | Level 1 |
| Database systems (Theory) | 3 | CSE Core Engineering | Level 1 |
| Database systems (Lab) | 1 | CSE Core Engineering | Level 1 |
| 1 Elective in the pool of Maths and Science | 4 | Mathematics and Basic Sciences | Level 1 |
| History of Ideas | 4 | HSS/M Core | Level 1 |
| **SEMESTER 5** | **23** | | |
| Introduction to Automata theory and Computability | 3 | CSE Core | Level 1 |
| Software Engineering (Theory) | 3 | CSE Core | Level 1 |
| Software Engineering (Lab) | 1 | CSE Core | Level 1 |
| Elective-1 | 4 | Elective | Level 1 |
| Elective-2 | 4 | Elective | Level 1 |
| Elective-3 | 4 | Elective | Level 1 |

| Course Name | Credits | Course Category | Level |
|---|---|---|---|
| Elective-4 | 4 | Elective | Level 1 |
| **SEMESTER 6** | **20** | | |
| Elective-5 | 4 | Elective | Level 2 |
| Elective-6 | 4 | Elective | Level 2 |
| Elective-7 | 4 | Elective | Level 2 |
| Elective-8 | 4 | Elective | Level 2 |
| Elective-9 | 4 | Elective | Level 2 |
| **SEMESTER 7** | **20** | | |
| Elective-10 | 4 | Elective | Level 2 / Level 3 |
| Elective – 11 | 4 | Elective | Level 2 / Level 3 |
| Elective – 12 | 4 | Elective | Level 2 / Level 3 |
| Elective – 13 | 4 | Elective | Level 2 / Level 3 |
| Elective – 14 | 4 | Elective | Level 2 / Level 3 |
| **SEMESTER 8** | **20** | | |
| Elective – 15 | 4 | Elective | Level 2 / Level 3 |
| Elective – 16 | 4 | Elective | Level 2 / Level 3 |
| Elective – 17 | 4 | Elective | Level 2 / Level 3 |
| Elective – 18 | 4 | Elective | Level 2 / Level 3 |
| Elective – 19 | 4 | Elective | Level 2 / Level 3 |
| **SEMESTER 9** | **20** | | |
| Combination of Courses and Project | 20 | Masters Project | Masters Project |
| **SEMESTER 10** | **20** | | |
| M.Tech. Project / Thesis | 20 | Masters Project | Masters Project |

**Table 9: Course Sequencing for iMTech (CSE)**

(**1 HSS Elective elective course needs to be completed by student in any semester starting from 5th semester to 9th semester as graduation requirement**.)

# 8 Course Sequencing for iMTech (ECE)

The course sequencing for the ECE branch is given in Table 10 below:

| Course Name | Credits | Course Category | Level |
|---|---|---|---|
| **Semester 1** | **18** | | |
| Mathematics - 1 | 4 | Mathematics and Basic Sciences | Level 1 |
| Programming in C | 2 | Programming | Level 1 |
| Programming in Python | 2 | Programming | Level 1 |
| Digital Design | 4 | Systems | Level 1 |
| Physical Education 1 | 0 | Others | Level 1 |
| English | 2 | HSS/M Core | Level 1 |
| Economics | 4 | HSS/M Core | Level 1 |
| **Semester 2** | **20** | | |
| Mathematics - 2 | 4 | Mathematics and Basic Sciences | Level 1 |
| Computer Architecture | 4 | Systems | Level 1 |
| Data Structures and Algorithms | 4 | Engineering Core | Level 1 |
| Data Structures Lab | 2 | Engineering Core | Level 1 |
| Computer Networks | 4 | Engineering Core | Level 1 |
| Technical Communication | 2 | HSS/M Core | Level 1 |
| Physical Education 2 | 0 | Others | Level 1 |
| **Semester 3** | **20** | | |
| Mathematics - 3 | 4 | Mathematics and Basic Sciences | Level 1 |
| Programming II | 4 | Programming | Level 1 |
| Physics (Theory) | 3 | Mathematics and Basic Sciences | Level 1 |
| Physics (Lab) | 1 | Mathematics and Basic Sciences | Level 1 |
| Signals and Systems | 4 | Systems | Level 1 |
| Electronic Circuits | 2 | ECE Core | Level 1 |
| Electronics Lab | 2 | ECE Core | Level 1 |
| **Semester 4** | **23** | | |
| Operating Systems (Theory) | 3 | Systems | Level 1 |
| Operating Systems (Lab) | 1 | Systems | Level 1 |
| Analog Circuits | 3 | ECE Core Engineering | Level 1 |
| Analog Circuits lab | 1 | ECE Core Engineering | Level 1 |
| Signal Processing | 3 | ECE Core Engineering | Level 1 |
| 1 Elective in the pool of Maths and Science | 4 | Mathematics and Basic Sciences | Level 1 |
| History of Ideas | 4 | HSS/M Core | Level 1 |
| Principles of Communication System | 3 | ECE Core Engineering | Level 1 |
| Principles of Communication System Lab | 1 | ECE Core Engineering | Level 1 |
| **Semester 5** | **19** | | |
| Digital Communication | 3 | ECE Core Engineering | Level 1 |
| Digital Communication Lab | 1 | ECE Core Engineering | Level 1 |
| Control Theory | 3 | ECE Core Engineering | Level 1 |

| Course Name | Credits | Course Category | Level |
|---|---|---|---|
| Mobile Computing | 4 | ECE Core Engineering | Level 1 |
| Elective-1 | 4 | Electives | Level 1 |
| Elective-2 | 4 | Electives | Level 1 |
| **Semester 6** | **20** | | |
| Elective-3 | 4 | Elective | Level 2 |
| Elective-4 | 4 | Elective | Level 2 |
| Elective-5 | 4 | Elective | Level 2 |
| Elective-6 | 4 | Elective | Level 2 |
| Elective-7 | 4 | Elective | Level 2 |
| **Semester 7** | **20** | | |
| Elective-8 | 4 | Elective | Level 2 / Level 3 |
| Elective – 9 | 4 | Elective | Level 2 / Level 3 |
| Elective – 10 | 4 | Elective | Level 2 / Level 3 |
| Elective – 11 | 4 | Elective | Level 2 / Level 3 |
| Elective – 12 | 4 | Elective | Level 2 / Level 3 |
| **Semester 8** | **20** | | |
| Elective – 13 | 4 | Elective | Level 2 / Level 3 |
| Elective – 14 | 4 | Elective | Level 2 / Level 3 |
| Elective – 15 | 4 | Elective | Level 2 / Level 3 |
| Elective – 16 | 4 | Elective | Level 2 / Level 3 |
| Elective – 17 | 4 | Elective | Level 2 / Level 3 |
| **Semester 9** | **20** | | |
| Combination of Courses and Project | 20 | Masters Project | Masters Project |
| **Semester 10** | **20** | | |
| M.Tech. Project / Thesis | 20 | Masters Project | Masters Project |

**Table 10: Course Sequencing for iMTech (ECE)**

(**1 HSS Elective elective course needs to be completed by student in any semester starting from 5th semester to 9th semester as graduation requirement**.)

# 9 References

1. Report on the Integrated M.Tech curriculum revision approved by the senate.

## APPENDIX A: Mathematics and Basic Science Course Syllabus

### Maths – 1 (4 Credits)

#1: Real numbers, Sequences, Series, Limit, Continuity, Mean value Theorems, Linear Approximation (10 hours)

#2: Power Series, Taylor’s theorem, Approximation to Polynomials (6 hours)

#3: Reimann Integral, Integral Calculus (7 hours)

#4: Space Coordinates, Polar coordinates, Cylinders, Quadric surfaces, volume, area, length, Continuity, Differentiability, Partial derivatives (9 hours)

#5: Double, Triple integrals, and Jacobians (3 hours)

#6: Introduction to first order Differential Equations (7 hours)

**Recommended books:**

Calculus and Analytic Geometry, by Thomas and Finney, 9th edition, Peasrson Education India

Calculus, Vol. 1: One Variable Calculus with an Introduction to Linear Algebra by Tom Apostol.

Calculus Vol 1: Multi-Variate Calculus and Linear Algebra with Applications to Differential Equations and Probability, by Tom Apostol.

### Maths -2 (4 Credits)

#1: Matrices, and various Matrix Operations (8 hours)
#2: Vector Spaces, and its related concepts (11 hours)
#3: Properties of Eigen values, and eigen vectors (10 hours)
#4: Complex Analysis (10 hours)
#5: Introduction to 2nd order differential equations (3 hours)

**Recommended books:**

Linear Algebra, by K Hoffman, and R Kunz, Prentice-Hall, 1971.
Relevant portions from the books written by Artin, Gallian, Herstein
Partial differential equations, by Fritz John 4th edition, 1981

## Maths-3 (4 Credits)

### Statistics and Probability

Data representation, average, probability, permutations, and combinations, random variables, probabilistic distributions, mean, and variance, binomial, poisson, hypergeometric, normal distributions, distribution of several random variables. (25 hours)

Mathematical statistics: random sampling, confidence intervals, testing 7-8 hypotheses, decisions, goodness of fit, $X^2$ test, linear regression. (15 hours)

### Recommended books

Advanced Engineering Mathematics, by Erwin Kreyszig, 8th edition, Wiley Proper web notes (NPTEL notes are available)

Introduction to Mathematical Statistics by Hogg and Craig, 3rd edition, Macmillan, 1971.
The advanced theory of Statistics by Kendall and Stuart, Volume 3, Griffin, 1976.

## Physics

#1: Integral theorems: Gauss divergence theorem, Greens theorem, Stokes theorem, and related topics.
#2: Simple harmonic motion and wave motion
#3: Electromagnetics
#4: Part of Quantum Mechanics: Schroedinger equation, harmonic oscillator, & basic quantum gates.
#5: Euler-Lagrange equation; principle of least action, generalized coordinates & generalized momenta, writing the Lagrangian of a system.

### Physics Lab experiments to accompany the theory needs to be finalized

Physics lab experiments in the previous (2015) curriculum is shown in the next page. This is just for reference.

## Physics Laboratory I - BS 107P (3rd semester), & Physics Laboratory II - BS108P (4th semester)

The following experiments are representative of those to be performed over the two semesters of physics laboratory:

0. Introduction to error analysis, dimensional analysis

1. Calculation of g & coefficient of restitution for a surface

2. Determination of rigidity modulus of brass

3. Determination of Young's modulus of a metal

4. Determination of thermal diffusivity of brass

5. Determination of the value of Stefan's constant

6. Measurement of electrical & thermal conductivity of good & poor conductors, calculation of Lorentz number of Cu

7. Bridge experiments (Maxwell, de Sauty, Wien)

8. Determination of storage capacity of a CD by a simple diffraction experiment

9. Construction of a data-logging pendulum using a mouse & obtaining the phase portrait & time-series of a simple damped pendulum

10. Numerical solution of the differential equation of a forced, damped simple pendulum using 4th order Runge-Kutta technique, and obtaining the phase portrait & time series of the system under different conditions

11. Experimental verification of the Biot-Savart law for the magnetic field of a current - carrying wire

12. Determination of the magnetic field strength of a bar-magnet, obtaining the functional relationship of field-strength to distance, obtaining the magnetic moment & then approximating the value of the Bohr magneton.

13. Understanding DNA diffraction using a spring & a laser pointer

14. Numerical study of a simple, nonlinear system (e.g., a quadratic map) & obtaining its bifurcation diagram, obtaining the Feigenbaum numbers.

15. Determination of period of rotation of the sun & sunspot cycles using FFT analysis of sunspot data.

16. Newton's rings experiment for determining the wavelength of a light source

17. Determination of surface tension of a given liquid using a travelling microscope & capillary tubes

18. Diffraction grating experiments using a sodium vapour lamp & a spectromenter

19. Other optics experiments using a prism & spectrometer

20. Determination of the viscosity of a liquid

## APPENDIX B: Humanities, Social Science and Other Course Syllabus

<table>
<tr><td><b>Course Name</b></td><td>Economics<br>(Proposed as one of the four HSS/M courses)</td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td>Prof. Rajagopalan</td></tr>
<tr><td><b>Course Instructor Name(s)</b></td><td>Prof. Rajagopalan</td></tr>
<tr><td><b>Course Type</b> <i>(Select one)</i><br><i>All course types except “Special Topics” go through the process for Academic Senate approval</i></td><td>Select one from the following:
<table>
<tr><td></td><td>Basic Sciences</td></tr>
<tr><td></td><td>Common Core (IT)</td></tr>
<tr><td></td><td>Elective</td></tr>
<tr><td></td><td>Engineering Science and Skills</td></tr>
<tr><td>X</td><td><b>HSS/M</b></td></tr>
<tr><td></td><td>IT in Domains</td></tr>
<tr><td></td><td>Miscellaneous</td></tr>
</table></td></tr>
<tr><td><b>Credits (L:T:P)</b><br><b>(Lecture : Tutorial : Practical)</b></td><td>4:0:0</td></tr>
<tr><td><b>Grading Scheme</b></td><td>Select one from the following:
<table>
<tr><td>X</td><td><b>4-point scale</b><br><b>(A,A-,B+,B,B-,C+,C,D,F)</b></td></tr>
<tr><td></td><td>Satisfactory/Unsatisfactory (S / X)</td></tr>
</table></td></tr>
<tr><td><b>Semester</b></td><td>Term: Spring<br>Year: 2012</td></tr>
<tr><td colspan="2"><b>Pre-Requisites (where applicable, specify exact course names)</b></td></tr>
<tr><td colspan="2">None</td></tr>
<tr><td colspan="2"><b>Course Description</b></td></tr>
<tr><td colspan="2">The iMTech curriculum requires students to take 4 courses (16 credits) from the area of HSS/M. The Economics course is a Level 1 course offered as the first course in HSS/M. A Level 2 course titled “The Economics of Information and ICTs” is going to be proposed as a follow-on of this first level course.</td></tr>
</table>

**Course Content**

1. Introduction : A conceptual understanding of macroeconomic interrelationships among science, engineering, technology, industry and national economy

2. Production functions and production possibility curves ; externalities ; Information Technology as an input to production

3. Division of labour, comparative advantages, global model of production in Information Technology, on site and off shore

4. National Income accounting, role of international trade, comparative advantage of nations

5. General theory of equilibrium; employment, interest rate and money ; business cycles; concepts of inflation, depression, stagnation , stagflation : Impact on IT industry

6. Monetary and Fiscal policies; pricing theories

7. Innovation : The Economics of Innovation, New Technologies, and Structural Change

8. Econometric methods

**Assessments (optional for Special Topics courses)**

Mid term examination: 30%
End Term Examination : 30%
Four Class Tests ( one after every two modules) : 40%

**Text Book / References**

1.Paul A Samuelson, William D Nordhaus(1983) : Economics ( text book)
2. Brian Snowdon,Howard R. Vane (2005) : Modern Macroeconomics :Its Origins, Development and Current State
3. Antonelli, C. (2003): The Economics of Innovation, New Technologies, and Structural Change
4. John C. Driscoll( 2001):Lecture Notes in Macroeconomics
5. Nicholas Georgescu-Roegen(1976): Energy and Economic Myths : Institutional and Analytical Economic Essays
6. Jack Johnston , John Dinardo (1996): Econometric Methods

***Spring Term**: Jan – Apr; **Summer Term**: Jun – July; **Prep Term**: July; **Fall Term**: Aug – Nov*

## English

**Overview**: To add to students' understanding of various forms of English literature, a brief overview of Grammar and practice in language skills.

**Objectives**:

i) The students will read literary texts analytically and discuss in class (with ensuing Grammar review and vocabulary building)
ii) Students will have an opportunity to be creative and learn to write poetry and appreciate a few modern and classic poets
iii) They will present their discussions through skits written by them based on assigned socially relevant topics/ readings

**Evaluation**:

Participation: 40 % - Students will be expected to come to class having completed the assigned readings/written assignments and prepared to contribute to discussions. Missing class and not contributing to discussions will result in loss of points as there will be in class ongoing assessment.

Online blog/Journal: 10%

Mid-term exam: 25% Final exam: 25 %

**Required readings**:

To be assigned and made available in the online lecture /Moodle class folder.

| | |
|---|---|
| **Course Name** | Technical Communication |
| **Course Proposer Name(s)** | |
| **Course Instructor Name(s)** | Valsala Rajasankaran |
| **Course Type** *(Select one)* | Select one from the following:<br>☐ Core<br>☐ Elective<br>☐ Special Topics Elective*<br>** All course types except "Special Topics Elective" go through the process for Academic Senate approval* |
| **Course Level** *(Select one)* | Select one from the following for elective courses:<br>☐ Level 1 Elective<br>☐ Level 2 Elective<br>☐ N/A |
| **Course Category** *(Select one)* | Select one from the following:<br>☐ Basic Sciences<br>☐ Common Core (IT)<br>☐ Elective<br>☐ Engineering Science and Skills<br>☐ HSS/M<br>☐ IT in Domains<br>☐ Miscellaneous |
| **Credits (L:T:P)**<br>**(Lecture : Tutorial : Practical)** | |
| **Grading Scheme** | Select one from the following:<br>☐ 4-point scale (A,A-,B+,B,B-,C+,C,D,F)<br>☐ Satisfactory/Unsatisfactory (S / X) |
| **Semester** | Term: (I / II / III / Prep)<br>Academic Year: |
| **Pre-Requisites (where applicable, specify exact course names)** | |
| | |
| **Course Description** | |
| This is an advanced communication skills course - both written and spoken. It will also enhance cognitive abilities from a linguistic perspective. The course will help students and professionals to express their ideas and opinions fluently and creatively. Assignments are designed to test communication as well as team skills. Course work includes making presentations and preparing various kinds of documents. Certain assignments will need to be submitted through email. | |

**Course Content**

The course focuses on comprehension, speaking, listening and writing skills. There will be exercises and role playing to reinforce each of these skills all through the duration of the course

Presentation Skills including body language
Listening Skills
Comprehension and Reading exercises
Logical thinking and Case Analysis
Basic Grammar
Writing: précis, essay, report, email
Emotional intelligence in communicating
Persuasion and negotiation
Meeting etiquette

**Assessments (optional for Special Topics courses)**

Presentations: 20%
Written Assignments: 20%
Test: 50%
Class Participation including attendance and prompt submission of assignments: 10%

**Text Book / References**

-NA-

***Prep Term**: July; **Term I**: Aug – Nov; **Term II**: Jan – Apr; **Term III**: Jun – July;*

# HSS 102: A HISTORY OF IDEAS

| | |
|---|---|
| **Course Name** | **A History of Ideas** |
| **Course Branch** | **iMtech** |
| **Course Proposer Name (s)** | **Bidisha Chaudhuri** |
| **Course Instructor Name(s)** | **Bidisha Chaudhuri** |
| **Course Type** | **Core** |
| **Course Level** | **Level 1** |
| **Credits** | **4** |
| **Grading Scheme** | A,A-,B+,B,B-,C+,C,D,F |
| **Course Description** | |

History of Ideas or Intellectual History is an interdisciplinary field of studies traversing the disciplinary boundaries of philosophy, history, natural science, art and literature, political and social thought and so on. As a field it focuses on how ideas about the world, either natural or social, have originated, evolved and transformed over time. The motive for studying such a wide field is to understand how knowledge is produced and disseminated and how epistemological lenses shape the way we perceive and conceptualize the world around us. There is no single way of talking about the history of ideas. Rather, there are many ways in which this field can be approached depending on the area of focus, historical time frame, spatial dimensions and so on.

This course is in no way an exhaustive account of history of ideas. Rather, it is a selection of intellectual trajectories and their proponents on the basis of the relevance and impact of their ideas across time and space, and their ability to permeate disciplinary boundaries and influence the overall pursuit of knowledge in the social sciences. Thus, the focus of the course remains on the economic, political and social ideas growing out of different temporal and intellectual contexts that represent different organizing principles of state and society.

The course starts with a focus on modern political, economic and sociological thought. It starts with a brief introduction to early liberal political philosophy on the nature of the modern state, society and sovereignty through the works of Hobbes, Locke and Rousseau. It also examines the ideas of Adam Smith and Karl Marx and Karl Polanyi to trace the emergence of modern economic thought. It then proceeds to major epistemological traditions in classical sociological thought developed by Weber, Durkheim and Gramsci while exploring a range of ideas on the state and economy, power and domination, the division of labour and social control, religion and society.

Then we move on to focusing on relationship between individual, society and system of knowledge. We explore social psychological ideas by Sigmund Freud and G.H. Mead through their works on the relationship between individual and society. These ideas lead to the relationship between society and knowledge through the works of Karl Mannheim (Sociology of Knowledge), Berger and Luckmann (Social Construction of Reality).

The following section presents a critical understanding of modernity and modern thought through the works of Michel Foucault and Anthony Giddens.

The course also touches upon modern Indian social and political thought through the writings of Indian thinkers such as, Gandhi, Tagore and Ambedkar and their ideas of nation, community, state, democracy and development. In each module, we will touch upon contemporary issues facing India and the relevance of modern social thought in assessing these issues.

The objective of the course is to provide students with a cursory yet critical understanding of some of the major philosophical traditions of 19th and 20th century and the contexts in which they originated and evolved. This basic understanding will help them firstly to comprehend how social sciences perceive and analyze the world around us and secondly how such analytical lenses influence and inform our understanding of the contemporary society in general.

**Course Content**

**01/01/2020**

- Introduction to the Course

**03/01/2020**

- Introduction to the Modernity and Enlightenment

**08/01/2020: Introduction to Modern Political Thought**

- **Hobbes & Locke**: Political Philosophy and Problems of Social Order
  Required Reading

- Bertrand Russell, *History of Western Philosophy* , George Allen and Unwin Ltd, 1947: 568-579, 642-665,711-727

**10/01/2020**

- **Montesquieu & Rousseau**: Political Philosophy and Problems of Social Order

  Required Reading

- Bertrand Russell, *History of Western Philosophy* , George Allen and Unwin Ltd, 1947: 568-579, 642-665,711-727

**15/01/2020: Holiday**

**17/01/2020: Movie/Discussion/Debate**

**22/01/2020: Introduction to Modern Economic Thought**

- **Adam Smith**: Rationality and Modern Economy

  Required Readings

- Adam Smith, "Of the Division of Labour" (Chapter I, Book I) in *The Wealth of Nations-1776* edited by Edwin Cannan, Bantam Books, 2003: 9-21

**24/01/2020**

- **Karl Marx**:

  State and Economy
  Alienation
  Class and Class Struggle

  Required Readings

- Karl Marx and Frederick Engels, "The Communist Manifesto-1848" in *Marx/Engels Selected Works, Vol. One*, Progress Publishers, 1969: Excerpts

**29/01/2020: Post-Marx: Antonio Gramsci & Karl Polanyi**: Culture, Society and Economy

Required Readings

- Femia, J. V. (1987). Gramsci's political thought: hegemony, consciousness, and the revolutionary process.
- Polanyi, K. *The Great Transformation*. New York: Farrar & Rinehart, 1944, selected pages

**31/01/2020: Movie/Discussion/Debate**

**05/02/2020: Introduction to Modern Sociological Thought**

- **Max Weber:**

  Economy and Society
  Politics and Religion

  Required Readings

- George Ritzer, *Classical Sociological Theory*, Second Edition, McGraw-Hill Companies, 1996: 217-263

**07/02/2020**

- **Emile Durkheim:**

  Social Facts and Scientific Study of Society
  Division of Labour
  Social Order and Religion

  Required Readings
- George Ritzer, *Classical Sociological Theory*, Second Edition, McGraw-Hill Companies, 1996: 183-216

**12/02/2020: Group Activity 1**
**14/02/2019: Group Activity 1**

**19/02/2020: Modernity and Mind**

- **Sigmund Freud and G.H. Mead**

  Required Readings
- Daniel K. Lapsley and Paul C. Stey, "Id, Ego and Superego" in Encyclopedia of Human Behavior, Second Edition edited by V.S. Ramachandran, Elsevier, 2011: 1-9
- George Ritzer, *Classical Sociological Theory*, Second Edition, McGraw-Hill Companies, 1996: 362-385

**21/02/2020: Holiday**

**26/02/2020: Knowledge and Modernity**

- **Karl Mannheim, Peter Berger and Thomas Luckmann:**
  Sociology of Knowledge and Reality
  Required Readings
- Lewis Coser, *Masters of Sociological Thought*, Indian Edition, Rawat Publications, 1996: 429-464
- Peter Berger and Thomas Luckmann, "The Foundation of Knowledge" in *Everyday Life in Social Construction of Reality: A Treatise in the Sociology of Knowledge*, Penguin Books, 1966: 31-62

**28/02/2020: Critiques of Modernity**

- **Michael Foucault**:

  Power and Knowledge

  Required Readings
- Michael Foucault, "Introduction" in *The Foucault Reader* edited by Paul Rabinow, Pantheon Books, 1984: 31-75

**2/03/2020 -7/03/2020: Mid Term Exam: Group Activity 2**

**11/03/2020:**

- **Anthony Giddens:** Reflexive Modernity
  <u>Required Readings</u>
- Anthony Giddens, Chapter 1 in The Consequences of Modernity, Polity Press, 1990, 1-53

---

**18/03/2020: Group Activity 2**

**20/03/2020: Group Activity 2**

**25/03/2020: Holiday**

**1/04/2020: Modernity in India**

- **Gandhi, Tagore, Ambedkar, Nehru** : Modern Political and Social Thought in India
  <u>Required Reading</u>
- Ramachandra Guha, *Makers of Modern India*, Penguin Books, 2010: Excerpts
- Ahmad, N. (2006). A note on Gandhi, Nation and Modernity. Social Scientist, 50-69
- Jodhka, S. S. (2002). Nation and village: Images of rural India in Gandhi, Nehru and Ambedkar. Economic and Political Weekly, 3343-3353.

**3/04/2020: Modernity in India**

- Gail Omvedt, "Ambedkarism: The Theory of Dalit Liberation" in *Dalits and the Democratic Revolution: Dr Ambedkar and the Dalit Movement in Colonial India*, SAGE Publications, 1994: Excerpts
- Parekh, B. (1991). Nehru and the national philosophy of India. Economic and Political Weekly, 35-48.
- Ramachandra Guha, *Makers of Modern India*, Penguin Books, 2010: Excerpts

**8/04/2020: Movie/Discussion/Debate**

**10/04/2020: Holiday**

**15/04/2020: Modernity and Indian Politics**

- Kaviraj, S. (2005). On the enchantment of the state: Indian thought on the role of the state in the narrative of modernity. *European Journal of Sociology/Archives Européennes de Sociologie*, *46*(2), 263-296.
- Jayal, N. G. (1994). The gentle leviathan: Welfare and the Indian state. Social Scientist, 18-26.
- Rudolph, L. I. (1965). The modernity of tradition: The democratic incarnation of caste in India. *American Political Science Review*, *59*(4), 975-989)

- Dirks, N. B. (1992). Castes of Mind. Representations, (37), 56-78.
- Mitra, S. K. (1991). Desecularising the State: religion and politics in India after independence. Comparative Studies in Society and History, 33(4), 755-777.
- Pantham, T. (1997). Indian secularism and its critics: Some reflections. The Review of Politics, 59(3), 523-540.

**17/04/2020: Movie/Discussion/Debate**

**22/04/2020: Modernity and Development in India**

- Escobar, A. (2011). Development and the Anthropology of Modernity. The postcolonial science and technology studies reader, 269
- Sen, A. (2001). "Introduction" in Development as Freedom. Oxford Paperbacks. Basole, A. (2005). The Economics of Ahimsa: Gandhi, Kumarappa, and the Non-Modern Challenge to Economics
- Bhaduri, A. (2017). A study in development by dispossession. Cambridge Journal of Economics, 42(1), 19-31.
- Ramachandra Guha and Joan Martinez Aller, *Varieties of Environmentalism: Essays North and South*, Routledge, 1997: 3-45
- Baviskar, A. (1997). Ecology and development in India: A field and its future. *Sociological bulletin*, *46*(2), 193-207.
- Gandy, M. (2008). Landscapes of disaster: water, modernity, and urban fragmentation in Mumbai. Environment and planning A, 40(1), 108-130

**24/04/2020: Movies/Discussions/Debate**

**29/04/2020: Technology and Indian Modernity**

Required Readings

- Arnold, D. (2013). *Everyday Technology: Machines and the Making of India's Modernity*. University of Chicago Press. [Selected Chapters]

**6/04/2020: Wrap Up**

**TBD**

- **Group Activity 3**
- **Group Activity 3**

**Assessments**

The evaluation scheme will be as indicated below.

- **Class Participation: (10+10=20%)**

  - Attendance
  - Reading assigned material and actively participating in class discussions
- **Group Activity (3*15= 45%)**
  This will take the form of storyboard-based group assignments. Groups will be fixed throughout the semester. There will be 2 components for scoring:
  - Presentation (10): 10 minutes to present for each group. Student groups will be presented storyboards ahead of time and asked to present their ideas in class on the designated activity day. These storyboards may include texts as well as audio visual materials. All group members will be uniformly marked.
  - Peer review Score (5): Each member of the group will give a score to their team members on the basis of their engagement and contribution to the group activity.
- **Examinations: 35%**
  - End Term Examination : 35%

## Physical Education

**Broad structure**: Physical Education will involve a daily routine of physical activity with games and sports. It will start with all students coming to the field at 7 am for light physical exercise or yoga. There also be games in the evening or at other suitable times according to the local climate. These activities will help students develop team work. Each student should pick one game and learn it for three weeks. Gardening is also included as an activity, if students are interested.

# APPENDIX C: Engineering Core Course Syllabus

<table>
<tr><td><b>Course Name</b></td><td>CC 103: Computer Networks</td></tr>
<tr><td><b>Course Branch</b></td><td>Select one from the following:
<table>
<tr><td></td><td>CS</td></tr>
<tr><td>X</td><td>ECE</td></tr>
</table></td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td>Prof. Tricha Anjali and Prof. D. Das</td></tr>
<tr><td><b>Course Instructor Name(s)</b></td><td>Prof. Tricha Anjali and Prof. D. Das</td></tr>
<tr><td><b>Course Type</b> <i>(Select one)</i></td><td>Select one from the following:
<table>
<tr><td>X</td><td>Core</td></tr>
<tr><td></td><td>Elective</td></tr>
<tr><td></td><td>Special Topics Elective*</td></tr>
</table>
<i>* All course types except “Special Topics Elective” go through the process for Academic Senate approval</i></td></tr>
<tr><td><b>Course Level</b> <i>(Select one)</i></td><td>Select one from the following for elective courses:
<table>
<tr><td></td><td>Level 1 Elective</td></tr>
<tr><td></td><td>Level 2 Elective</td></tr>
<tr><td></td><td>N/A</td></tr>
</table></td></tr>
<tr><td><b>Course Category</b> <i>(Select one)</i></td><td>Select one from the following:
<table>
<tr><td></td><td>Basic Sciences</td></tr>
<tr><td>X</td><td>Branch Core (CSE / ECE)</td></tr>
<tr><td></td><td>Elective</td></tr>
<tr><td></td><td>Engineering Science and Skills</td></tr>
<tr><td></td><td>HSS/M</td></tr>
<tr><td></td><td>Miscellaneous</td></tr>
</table></td></tr>
<tr><td><b>Credits (L:T:P)</b><br><b>(Lecture : Tutorial : Practical)</b></td><td>
<table>
<tr><th>Hours</th><th>Component</th></tr>
<tr><td>4</td><td>Lecture (1hr = 1 credit)</td></tr>
<tr><td></td><td>Tutorial (1hr = 1 credit)</td></tr>
<tr><td></td><td>Practical (2hrs = 1 credit)</td></tr>
<tr><td></td><td><b>Total Credits</b></td></tr>
</table></td></tr>
<tr><td><b>Grading Scheme</b></td><td>Select one from the following:</td></tr>
</table>

| | |
|---|---|
| X | 4-point scale<br>(A,A-,B+,B,B-,C+,C,D,F) |
| | Satisfactory/Unsatisfactory (S / X) |

**Pre-Requisites**
*(where applicable, specify exact course names)*

None

**Course Description**
*A brief description of the course*

The main aim of this course is to make the students understand, how the different kind of networks are interconnected and the various types of applications run over them by transmitting packets from one part of the globe to the other efficiently. Hence the course deals with application, transport, network and Data link layers protocols/algorithms.

**Course Outcomes**
*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this course, the student is expected to:

1. Know the protocol stack for the Internet
2. Understand the need for protocols
3. Know the detailed operations of the TCP/IP protocols
4. Be able to create new applications that can communicate over the network

**Course Content**

**Lectures 1-2**
The first lecture is to make the students oriented towards the subjects to be covered in this course and why. The grading system and the books referred. Logical and physical topologies and why we need so many topologies?

**Lecture 3**
Client, Server, Connection oriented and connectionless services, Layered architecture, Internet protocol layer, Circuit-Packet-Message switching,

**Lectures 4-6**
Need of services by application layer protocols, HTTP, FTP, SMTP, SMTP vs HTTP, MOME, DNS

**Lecture 7-8**
Socket programming for TCP and UDP

**Lecture 9-10**
Relationship of transport layer with application and network layer, Multiplexing and Demultiplexing, UDP

**Lecture 11-12**
GBN, SR, TCP: connection, segment structure,
**Lecture 13-14**
Flow control, and Congestion control algorithms
**Lecture 15-16**
Link-state routing algorithm, Distance-vector routing algorithm,
**Lecture 17**
Intra-autonomous system routing: RIP, OSPF, Inter-autonomous system routing: BGP
**Lecture 18**
IPv4 and IPv6 packet format and basic differences and alignments,

**Mid Term Exam.**

**Lecture 19**
Mobility at network layer,
**Lecture 20-21**
Error detection and correction techniques; multiple access protocols in LAN: channel portioning, random access, taking turn;
**Lecture 22**
Address resolution protocol
**Lecture 23-26**
Taxonomy of Medium Access Control (MAC), Wired and Wireless LAN medium access Control Protocol Pure/Slotted ALOHA, CSMA, CSMA/CD: Ethernet,

**Lecture 27-28:** Why Software Defined Network (SDN) needed? Architecture of SDN.

**Lab:** has theory part and lab component covers presentation and lab experiment part.

**Assessments / Grading**

There will be a mid-term (25marks) and one final examination (30 marks), two class tests (2 * 10), Lab/Assignments (20 Marks) as well as class performance (5 marks) will be considered for final grading.

**Text Book / References**

- Computer Networking, by Kurose and Ross
- Local Area Network, by G. Keiser
- Performance Analysis of the IEEE 802.11 Distributed Coordination Function, by G. Bianchi, IEEE Journal of Selected Areas in Communications, Vol. 18, No. 3, March 2000.
- Software Defined Network

| | |
|---|---|
| **Course Name** | Programming in C |
| **Course Branch** | Select one from the following:<br>CSE and ECE |
| **Course Proposer Name(s)** | Dr. Madhav Rao |
| **Course Instructor Name(s)** | |
| **Course Type** *(Select one)* | Select one from the following:<br>**Core**<br>Elective<br>Special Topics Elective*<br>** All course types except "Special Topics Elective" go through the process for Academic Senate approval* |
| **Course Level** *(Select one)* | Select one from the following for elective courses:<br>Level 1 Elective<br>Level 2 Elective<br>N/A |
| **Course Category** *(Select one)* | Select one from the following:<br>Basic Sciences<br>Branch Core (CSE / ECE)<br>Elective<br>Engineering Science and Skills<br>HSS/M<br>Miscellaneous |
| **Credits (L:T:P)**<br>**(Lecture : Tutorial : Practical)** | **Hours** \| **Component**<br>**Lecture (1hr = 1 credit)**<br>Tutorial (1hr = 1 credit)<br>**Practical (2hrs = 1 credit)**<br>**Total Credits = 2** |
| **Grading Scheme** | Select one from the following:<br>4-point scale (A,A-,B+,B,B-,C+,C,D,F)<br>Satisfactory/Unsatisfactory (S / X)<br>4-point scale |
| **Pre-Requisites**<br>*(where applicable, specify exact course names)* | |
| | |
| **Course Description** | |

*A brief description of the course*

This course is first of the two programming courses. This knowledge area includes those skills and concepts that are essential to programming practice independent of the underlying specialization. As a result, this area includes units on fundamental programming concepts, basic data structures, algorithmic processes, and basic security. These units, however, by no means cover the full range of programming knowledge that a IT undergraduate must know. It is expected that a second programming course is taught that reinforces these concepts.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this class, a student should understand the concepts of:

- defining, using, and modifying variables
- formulating expressions to represent desired quantities
- controlling the execution of code within a program
- defining and calling functions
- generating thorough test suites
- debugging skills to solve semantic program faults
- organizing code using system utilities

**Course Content**

Theory Contents

- Introduction to computer problem-solving.
- Fundamental data structures (Data types, representation of numeric data, strings, etc.)
- Fundamental algorithms.
- Factoring methods. Array techniques.
- Merging, sorting and searching.
- Text processing and pattern searching.
- Dynamics data structure algorithms.
- Recursive algorithms.

The topics to be covered at a fundamental level with focus more on practice.

All the sessions of the C Programming Lab will end with the description of a stretch exercise that students can work on outside of the lab hours. The C Programming Labs are structured based on specific themes for each lab session. Each lab session is divided into multiple lab exercises.

Lab components:

- Lab 1: Preliminaries.

– Objective: The objective of this lab is to familiarize the students with the C programming environment.

– Exercises:

  - Introduction to Unix.
  - Basic I/O program 1.
  - Basic I/O program 2.
  - Basic I/O program 3.
  - Basic I/O program 4.

– Comment: Lab 1 is intentionally kept light because the basic objective is to familize the student with the programming environment, which includes Unix operating system, editor, compilation, execution, etc.

- Lab 2: Data Types and Expressions.

– Objective: The objective of this lab is to start using variables of various primary data types in the C language and use them as part of various expressions.

– Exercises:

  - Variables and data types.
  - Type casting and data. Expression evaluation.

- Lab 3: Control Flow.

– Objective: The objective of this lab is to provide an introduction to control structures in C language.

– Exercises:

  - Control: if statement.
  - Control: if-else statement.
  - Control: switch-case statement.
  - Iterative: for loop.
  - Iterative: while loop.
  - Iterative: do-while loop.

- Lab 4: Functions.

– Objective: The objective of this lab is to introduce modular software development using functions.

– Exercises:

  - Function exercise #1 (prototypes, void return and void parameters).
  - Function exercise #2 (parameters and return values).
  - Function exercise #3 (global variables).
  - Function exercise #4 (static variables).
  - Function exercise #5 (multi-file programming).
  - Introduction to built-in libraries (math.h, string.h, etc.).

- Lab 5: Recursion.

– Objective: The objective of this lab is to understand recursion in C programming language.

– Exercises:

  - Recursion exercise #1.
  - Recursion exercise #2.

- Lab 6: Arrays.

– Objective: The objective of this lab is to introduce the students to arrays in C programming language.
– Exercises:
    - 1-d array exercise #1.
    - 1-d array exercise #2.
    - 2-d array exercise #3.
    - n-d array exercise #4.

- Lab 7: Pointers.

– Objective: The objective of this lab is to learn about pointers in C language.
– Exercises:
    - Pointers and addresses.
    - Pointers and function arguments.
    - Pointers and arrays.
    - Address arithmetic.
    - Character pointers and functions.

- Lab 8: More on Pointers.

– Objective: The objective of this lab is to learn about advanced concepts about pointers in C language.
– Exercises:
    - Pointer arrays.
    - Pointers to pointers.
    - Pointers to functions.

- Lab 9: Structures.

– Objective: The objective of this lab is to learn about structures in C programming language.
– Exercises:
    - Basics of structures.
    - Structures and functions.
    - Arrays of structures.

- Lab 9: Advanced Structures and Unions.

– Objective: The objective of this lab is to learn about advanced concepts in structures and unions in C programming language.
– Exercises:

    - Pointers to structures.
    - Self-referential structures.
    - Unions.
    - Bit-fields.

- Lab 10: File I/O.

– Objective: The objective of this lab is to learn how to do File I/O using C programming language.
– Exercises:
    - Text I/O sequential access.
    - Binary I/O sequential access.
    - Binary I/O random access.

- Lab 11,12: C Programming Project.

  – Objective: The objective of the last two lab sessions is to do a non-trivial programming project that tries to make use a majority of the C programming language constructs and paradigms. The project can be a group project with 3 members each. The size of the project should be such that completion of the project should be possible in about 4 hours of collective programming (about 10 person hours).

**Assessments / Grading**

25% Mid-term exam
25% Final exam
30% Lab work, Assignments, and Project
20% Quizzes

**Text Book / References**

*The C Programming language* by Kernighan and Ritchie.

*How to solve it by Computers* by Dromey (Reference textbook)

*Code Complete* by McConnell (Reference textbook)

| | |
|---|---|
| **Course Name** | Programming in Python |
| **Course Branch** | Select one from the following:<br>x CSE<br>ECE |
| **Course Proposer Name(s)** | |
| **Course Instructor Name(s)** | |
| **Course Type** *(Select one)* | Select one from the following:<br>x Core<br>Elective<br>Special Topics Elective*<br>** All course types except "Special Topics Elective" go through the process for Academic Senate approval* |
| **Course Level** *(Select one)* | Select one from the following for elective courses:<br>Level 1 Elective<br>Level 2 Elective<br>N/A |
| **Course Category** *(Select one)* | Select one from the following:<br>Basic Sciences<br>x Branch Core (CSE / ECE)<br>Elective<br>Engineering Science and Skills<br>HSS/M<br>Miscellaneous |
| **Credits (L:T:P)**<br>**(Lecture : Tutorial : Practical)** | **Hours** **Component**<br>1 Lecture (1hr = 1 credit)<br>Tutorial (1hr = 1 credit)<br>1 Practical (2hrs = 1 credit)<br>**2 Total Credits** |
| **Grading Scheme** | Select one from the following:<br>x 4-point scale (A,A-,B+,B,B-,C+,C,D,F)<br>Satisfactory/Unsatisfactory (S / X) |
| **Pre-Requisites**<br>*(where applicable, specify exact course names)* | |
| **Course Description**<br>*A brief description of the course* | |

This introductory course in programming introduces the Python programming language. The objective of this course is to equip students with problem solving skills using programming as a tool. Python, being a comparatively high-level programming language as compared to C, gives a good opportunity to concentrate on the fundamental tenets of problem solving instead of getting overwhelmed with syntax and runtime errors. The stress of this course is to enable students to start with non-trivial programming problems and to leverage

Course Outcomes

Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.

At the end of this course, the student is expected to:

- Know programming in Python, its syntax, semantics, library
- Be able to solve computing problems using Python

**Course Content**

Basic feature: expressions, operators, Top-level, REPL, Types, Variables, If-elif-else,Writing programs in files

While loops, Lists, For loops, Tuples, Dictionaries

Functions, design programs with functions, Example – Calendar, Inner Functions, List Comprehension, Recursive functions, Recursive Functions and Eight Queen problem

Application of recursive functions in data-structure and algorithm design: Examples, family tree, Money change, Jug, Power Product

Modules, Using modules using `import` and `from ⋯ import ⋯`, Writing modules

Higher order functions: Functions taking functions as parameters, Comparison with function pointers, Closures, Higher order functions: Functions returning functions

Introduction to Object Oriented Programming, classes, objects, __init__, static attributes, inheritance, polymorphism, duck typing, Object oriented software design

**Assessments / Grading**

4 Quizes: 5 marks each – 20
Project – 20
Mid-term – 30
End-term – 30

The actual marks distribution may differ from the above subject to the dynamics of the course.

**Text Book / References**

Python Essential Reference – David M. Beazley
Online resources

**Course Name**

Data Structures and Algorithms

**Course Branch**

Select one from the following:

| | |
|---|---|
| X | CSE |
| | |

**Course Proposer Name(s)**

Muralidhara V N

Muralidhara V N

**Course Type** *(Select one)*

Select one from the following:

| | |
|---|---|
| X | Core |
| | Elective |
| | Special Topics Elective* |

** All course types except "Special Topics Elective" go through the process for Academic Senate approval*

**Course Level** *(Select one)*

Select one from the following for elective courses:

| | |
|---|---|
| | Level 1 Elective |
| | Level 2 Elective |
| | N/A |

**Course Category** *(Select one)*

Select one from the following:

| | |
|---|---|
| | Basic Sciences |
| X | Branch Core (CSE / ECE) |
| | Elective |
| | Engineering Science and Skills |
| | HSS/M |
| | Miscellaneous |

**Credits (L:T:P)**
**(Lecture : Tutorial : Practical)**

| **Hours** | **Component** |
|---|---|
| 3 | Lecture (1hr = 1 credit) |
| 1 | Tutorial (1hr = 1 credit) |
| 1 | Practical (2hrs = 1 credit) |
| **4+1** | **Total Credits** |

**Grading Scheme**

Select one from the following:

| | |
|---|---|
| | 4-point scale (A,A-,B+,B,B-,C+,C,D,F) |
| | Satisfactory/Unsatisfactory (S / X) |

**Pre-Requisites**
*(where applicable, specify exact course names)*

Programming I

**Course Description**

*A brief description of the course*

The introduces the notion of efficient algorithms. It covers operations on data structures like arrays, linked lists, hashing, stack, queue, binary trees, priority queues, balanced binary search trees and graphs and their application in designing efficient algorithms.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this course, the student is expected to:

1. to know what are efficient algorithms.
2. Compute the time and space complexity of algorithms.
3. Know the difference between worst/best/average  and amortized cost.
4. know about arrays, linked lists, stacks, queue and hashing techniques, sorting and binary search.
5. Understand various types of tree structures and graph as a data structure.
6. Apply the knowledge of data structures to design efficient algorithms.

**Course Content**

Introduction to Algorithms and Complexity.

Sorting Algorithms: Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, Quick Sort, Heap Sort, Lower bound on sorting, Count Sort, Radix Sort, Bucket Sort.

Elementary Data structures: Arrays, Linked Lists, Stack, Queue, hashing including perfect hashing.

Binary Trees: Basic properties, representation, various types, level and height of a node, traversal - in order, pre order, post order, level order.

Priority Queues: Binary Heap, Binomial Heap, Amortized Analysis, Fibonacci Heaps and applications.

Balanced Binary Search Trees: AVL Trees and Red-Black Trees and applications.

Graphs: Different ways of representing graphs, graph traversal (BFS/DFS) with applications Topological Sort and Strongly connected componets, shortest path problem and Dijstra's algorithms, Minimum Spanning Trees - Prime's and Kruskal's Algorithms with applications.

**Assessments / Grading**

Exams/Tests/Quizzes/Assignments

**Text Book / References**

**Introduction to Algorithms by Cormen, Leiserson and Rivest, Stein, Pub: MIT Press(2009)**

**The Design and Analysis of Computer Algorithms by Aho, Hopcroft and Ullman, Pub-Addison Wesley**

## Digital Design

*Objectives:*

Basics of Digital Logic and Circuits component are theory centric whereas Basics of Electronics component is laboratory centric. The grade awarded to a student may reflect the consolidated assessment of the two components.

The objectives of the course can be broadly stated as the following:

(i) Basics of Digital Logic and Circuits: The objectives of this component include providing the students a basic understanding of the principles of digital electronics and digital design.

*Proposed Course Contents*

**(i) Basics of Digital Logic and Circuits:**
3 lecture hours/week + 1 Tutorial, totaling 42 hours + 24 T.

- Number systems and codes:
  – Review of number systems; binary arithmetic; binary weighted and non-weighted codes; error detecting and error correcting codes.
- Boolean algebra:
  – Postulates and theorems, representation of switching functions; SOP & POS forms; Karnaugh map representation; minimization using K-map.
- Design of combinational circuits:
  – Tabular minimization; design of single output and multi output functions; design using con-ventional AND, OR, NOT, NAND, NOR & EX-OR gates; design using MSI & LSI devices; digital multiplexer/selector, decoder, demultiplexer; design of 4 bit adder, carry look-ahead adder, magnitude comparator; BCD converter, logic implementations using ROM, PAL & PLA.
- Introduction to sequential circuits
  – Combinational vs. sequential circuits, asynchronous vs. synchronous circuits; state table and state diagram; state assignment; memory elements and their excitation functions; T flip-flop, D flip-flop, R-S flip-flop, JK flip-flop and their excitation requirements; design of synchronous sequenetial circuits like sequence detectors and binary counters.
- Capabilities and minimization of sequential machines:
  – Melay and Moore machines; capabilities and limitations of finite state machine; state equivalence and machine minimization.
- Algorithmic state machines:
  – ASM chart, timing considerations, control implementation; design with multiplexers and PLA control; introduction to unate functions and threshold logic

The recommended textbooks for this component of the course are as follows:

- “Switching and Finite Automata Theory,” by Kohavi, TMH edition.
- “Digital Logic Design,” by Mano.
- “Introduction to Switching Theory and Logic Design,” by F.J. Hill and Peterson, John Wiley Publications.

“Digital Design with Standard MSI and LSI,” by Thomas, Blakeslee, Wiley Interscience Publication.

# ES 103 Programming II

| | |
|---|---|
| Course Name | ES 103 Programming II |
| Instructors | Prof. Chandrashekar R<br>Office No. 116<br>rc@iiitb.ac.in |
| TA | Vivek Yadav |
| Course credits | 4 |
| Pre-requisite | • Good knowledge of programming in C / Python |

## Course Outline

This is a second course on programming that is expected to reinforces the concepts taught in Programming I and the Data Structures course. This course mainly introduces the students to object-oriented programming and the basics of event driven programming.

This course is accompanied by C++ and Java programming laboratory session.This course will provide an in-depth understanding of data modeling both from a theoretical as well as a practical point of view. Building on the concepts introduced in the DBMS course, this course will provide a thorough understanding of advanced data modeling concepts. In particular, the course will cover in detail advanced relational database concepts, XML, object-oriented databases and Data Warehouse concepts. The emphasis is not only on the logical aspects of data modeling but also the physical/implementation aspects of these data models using freely available software.

- Object-oriented design.
- Encapsulation and information-hiding.
- Separation of behavior and implementation.
- Classes and subclasses.
- Inheritance (overriding, dynamic dispatch).
- Polymorphism (subtype polymorphism vs. inheritance).
- Event-handling methods.
- Event propagation.
- Exception handling.

## Class projects discussions and demos

## Grading Policy

**Final grade will be based on weights given below:**

30%: Mid-Term Exam

20%: Programming Tests / assignments

15%: Project

30%: End-Term Exam

05%: Instructor discretion

References

- An Introduction to Object-Oriented Programming, 3e By Timothy Budd
- C++ Primer Fifth Edition by Stanley Lippman et al
- Java - How to Program, 9/e - Deitel & Deitel

| | |
|---|---|
| **Course Name** | Computer Architecture (EG201) |
| **Course Branch** | Select one from the following: (Place X appropriately)<br>ECE X<br>CSE X |
| **Course Proposer Name(s)** | Nanditha Rao, and Subir K Roy |
| **Course Instructor Name(s)** | Nanditha Rao |
| **Course Type** *(Select one)* | Select one from the following: (Place X appropriately)<br>Core X<br>Elective<br>Special Topics Elective*<br>** All course types except "Special Topics Elective" go through the process for Senate approval* |
| **Course Level** *(Select one)* | Select one from the following for elective courses:<br>(Place X appropriately)<br>Level 1 Elective |

|  |  |
|---|---|
|  | Level 2 Elective<br>N/A X |
| **Course Category** *(Select one)* | Select one from the following:<br>(Place X appropriately)<br>Basic Sciences<br>Branch Core (CSE / ECE)<br>Elective<br>Engineering Science and Skills X<br>HSS/M<br>Miscellaneous |
| **Credits (L:T:P)**<br>**(Lecture : Tutorial : Practical)** | (Place X appropriately)<br>**Hours Component**<br>**Lecture (3hrs = 3 credit)**<br>Tutorial (0hr = 0 credit)<br>**Practical (2hrs = 1 credit)**<br>**Total Credits: 4 (One Letter Grade)** |
| **Grading Scheme** | Select one from the following:<br>(Place X appropriately)<br>4-point scale<br>**(A,A-,B+,B,B-,C+,C,D,F) X**<br>Satisfactory/Unsatisfactory (S / X)<br>One Letter Grade for Lecture + Practical. |

**Pre-Requisites**

*(where applicable, specify exact course names)*

Digital Logic Design, and Computer Arithmetic

**Course Description**

*A brief description of the course*

The course explains and discusses the internal blocks of modern computer architecture, including processor, and instruction set.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of the course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

Students will learn assembly language, I/O's, interrupts, 8085 and interfacing, and MIPS programming, and performance evaluation.

**Course Content**

Following contents will be studied in this course:

**Topic 1: Fundamentals of assembly language:**

Stored program concept, RISC vs CISC, Harvard vs Von Neumann architecture, RISC computer: Princeton/IAS computer, instruction set, programming, Instruction encoding, Stack, functions

**Topic 2: I/O, Interrupts:**

IO fundamentals: handshaking, programmed IO, interrupt driven IO; Interrupt handling mechanism, Buses: protocols, arbitration, direct memory access (DMA), PCI timing.

**Topic 3: 8085 and interfacing:**

Example study: 8085 architecture, timing, 8085 instruction set, Addressing modes, Basics of assembly level programming, Interfacing: Programming counters, delays, interrupt controller, memory interfacing.

**Topic 4: MIPS**

MIPS instruction set, MIPS assembly programming (practical), data and control path design, Pipelining, hazards: data, control, structural hazard, Performance evaluation, Exceptions, forwarding, introduction to branch prediction, Memory: SRAM, DRAM, Cache memory, memory hierarchies, performance evaluation, Case study of a modern day processor architecture

**Assessments / Grading**

Quizzes,

Midterm Exam,

Final Exam,

Lab activities and Assignments

**Text Book / References**

1. Computer Organization and Architecture, W. Stalling
2. Computer Architecture: A Quantitative Approach, by Hennessy & Patterson.
3. Digital Design and Computer Architecture, D Harris, and S. Harris.

## Signals and Systems

Objectives

This course provides students with an exposure to the concepts of signals and systems.

Proposed Course Contents

- Dynamic representation of systems, systems attributes, causality, linearity, stability, time-invariance; special signals, complex exponentials, singularity functions (impulse and step functions); Linear Time-Invariant Systems, differential equation representation, convolution integral; discrete form of special functions; discrete convolution and its properties; realization of LTI systems (differential and difference equations).
- Fourier analysis of continuous time signals and systems, Fourier series, Fourier Transform and properties, Parsevals theorem, frequency response of LTI systems; sampling theorem.
- Fourier analysis of discrete time signals & systems, Discrete-time Fourier series, Discrete-time Fourier Transform (including DFT) and properties; frequency response of discrete time LTI systems.
- Laplace Transform and its inverse, definitions, existence conditions, region of convergence and properties, applications of Laplace Transform for the analysis of continuous time LTI system (stability etc.), significance of poles and zeros.
- Z-Transform and its inverse, definitions, existence, region of convergence and properties, applications of Z-Transform for the analysis of discrete time LTI systems, significance of poles and zeros.
- Random signals, introduction to probability, Bayes Theorem, concept of random variable, proba-bility density and distribution functions, function of a random variable; moments, independence of a random variable; introduction to random processes; auto- and cross- correlations; wide-sense stationarity, power spectral density, white noise, random processes through LTI systems.

The recommended textbooks for this course are:

- “Signals and Systems,” by Alan V. Oppenheim and Alan S. Willsky, Pearson Edn.
- “Communication Systems,” by Haykin Simon, John Wiley.
- “Signals and Systems,” by I. J. Nagrarth, Tata Mc Graw Hill.
- “Signals and Systems,” by Farooq Husain, Umesh Pub.
- “Adaptive Signal Processing,” by W. Bernad, Pearson Edn.

## Operating Systems

*Objectives*

A joint ACM and IEEE Computer Society curriculum recommendation for undergraduate degree in computer science from the report in 2008 proposes that a course in Operating Systems should explain the issues that influence the design of contemporary operating systems. Since operating systems is the software layer that abstracts the hardware to enable programmers to control it, the students will require a laboratory component to experiment with operating systems.

*Proposed Course Contents*

- Overview of operating systems:
  - Role and purpose of operating systems; evolution of operating systems; functionality and purpose of a typical operating system; client-server models; design issues - efficiency, robustness, flexibility, portability, security, compatibility; interactions with computer architecture.
- Operating systems principles:
  - Structuring methods - monolithic, layered, modular, micro-kernel, virtual machine, exokernel; abstractions, processes, resources; device organization; interrupts; user vs. kernel modes, transition of modes.
- Concurrency:
  - States and state diagrams; structures; dispatching and context switching; concurrent execution; mutual exclusion; deadlocks - causes, conditions, prevention; models and mechanisms - semaphores, monitors, condition variables, rendezvous; producer-consumer problems and synchronization; multiprocessor issues - spin-locks, reentrancy .
- Scheduling and dispatching:
  - Preemptive vs. non-preemptive scheduling; schedulers and policies; processes and threads; interprocess communication; classical IPC problems - Dining Philosophers, Readers and Writers.
- Memory management:
  - Physical memory and memory management hardware; paging, virtual memory, segmentation; working sets, thrashing; caching.
- Security and protection:
  - Overview - threats, intruders; cryptography; protection mechanisms; authentication; insider at-tacks; malware; defenses - protection, access control; backups.
- File Systems:
  - Files - data, metadata, organization, buffering; directories - contents, structure; file systems - partitioning, mount/unmount, virtual file system; implementation techniques; memory-mapped files; special purpose file systems; management and optimization - naming, searching, access, backups.

This course will involve lectures, and tutorials, and will be accompanied by an Operating Systems laboratory session.

# APPENDIX D: ECE Core Course Syllabus

<table>
<tr><td><b>Name</b></td><td>Basic Electronics Circuits</td></tr>
<tr><td><b>Course Branch</b></td><td>Select one from the following:<table><tr><td></td><td>ECE</td></tr><tr><td></td><td></td></tr></table></td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td>Madhav Rao and Subhajit Sen</td></tr>
<tr><td><b>Course Instructor Name(s)</b></td><td>Madhav Rao, Subhajit Sen, Subir Roy</td></tr>
<tr><td><b>Course Type</b> <i>(Select one)</i></td><td>Select one from the following:<table><tr><td>x</td><td>Core (ECE)</td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr></table><i>* All course types except “Special Topics Elective” go through the process for Academic Senate approval</i></td></tr>
<tr><td><b>Course Level</b> <i>(Select one)</i></td><td>Select one from the following for elective courses:<table><tr><td></td><td>Level 1 Elective</td></tr><tr><td></td><td>Level 2 Elective</td></tr><tr><td></td><td>N/A</td></tr></table></td></tr>
<tr><td><b>Course Category</b> <i>(Select one)</i></td><td>Select one from the following:<table><tr><td></td><td>Basic Sciences</td></tr><tr><td>x</td><td>Branch Core (CSE / <b>ECE</b>)</td></tr><tr><td></td><td>Elective</td></tr><tr><td></td><td>Engineering Science and Skills</td></tr><tr><td></td><td>HSS/M</td></tr><tr><td></td><td>Miscellaneous</td></tr></table></td></tr>
<tr><td><b>Credits (L:T:P)</b><br><b>(Lecture : Tutorial : Practical)</b></td><td><table><tr><td><b>Hours</b></td><td><b>Component</b></td></tr><tr><td>2 hr</td><td>Lecture (1hr = 1 credit)</td></tr><tr><td></td><td>Tutorial (1hr = 1 credit)</td></tr><tr><td></td><td>Practical (2hrs = 1 credit)</td></tr><tr><td><b>2</b></td><td><b>Total Credits</b></td></tr></table></td></tr>
<tr><td><b>Grading Scheme</b></td><td>Select one from the following:<table><tr><td></td><td>4-point scale (A,A-,B+,B,B-,C+,C,D,F)</td></tr><tr><td></td><td></td></tr></table></td></tr>
<tr><td colspan="2"><b>Pre-Requisites</b><br><i>(where applicable, specify exact course names)</i></td></tr>
</table>

**Course Description**

*A brief description of the course*

The objective of the course is to provide students broad and in depth knowledge in the field of electronics.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of the course, the student is expected to design circuits consisting of passive and opamp based active components. Given a circuit design, students should be able to distinguish the usage of individual circuit blocks in a given design.

**Course Content**

The course content should cover the following topics:
1. DC circuits covering Kirchoffs law, thevenin's law, norton's theorm, source transformation, Pi-Delta transformation.
2. AC Circuits covering average value, RMS, phasor representation of AC signals, Transient and steady state response of RC, RL, RLC circuits, and passive filter circuits using combination of R, L and C, differentiator and integrator circuits and introduction to Spice simulation.
3. Working principle of transformers, DC motors, and induction motors and different electrical power sources (Wind, thermal, solar, fuel cells etc.).
4. Diodes and applications covering ideal versus practical resistance levels, load line analysis, rectifier circuits, Rectifier with and without filters, Zener diode and its applications, opto-electronic devices.
5. Operational amplifiers covering inverting, non-inverting amplifiers, virtual ground concept, summing and difference amplifiers, voltage follower, comparator, integrator, and differentiator.

**Assessments / Grading**

Midterm exam-40%
Final exam-40%
Quizzes-10%
Assignments-10%

**Text Book / References**

1. Fundamental of Electric Circuits - Charles K Alexander and Matthew Sadiku
2. Electronic devices and circuit theory - Boylestad and Nashelsky

<table>
<tr><td><b>Course Name</b></td><td>Electronics Circuits laboratory</td></tr>
<tr><td><b>Course Branch</b></td><td>Select one from the following:<table><tr><td></td><td>ECE</td></tr><tr><td></td><td></td></tr></table></td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td>Madhav Rao and Subhajit Sen</td></tr>
<tr><td><b>Course Instructor Name(s)</b></td><td>Madhav Rao, Subhajit Sen, Subir Roy</td></tr>
<tr><td><b>Course Type</b> <i>(Select one)</i></td><td>Select one from the following:<table><tr><td></td><td>Core (ECE)</td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr></table><i>* All course types except “Special Topics Elective” go through the process for Academic Senate approval</i></td></tr>
<tr><td><b>Course Level</b> <i>(Select one)</i></td><td>Select one from the following for elective courses:<table><tr><td></td><td>Level 1 Elective</td></tr><tr><td></td><td>Level 2 Elective</td></tr><tr><td></td><td>N/A</td></tr></table></td></tr>
<tr><td><b>Course Category</b> <i>(Select one)</i></td><td>Select one from the following:<table><tr><td></td><td>Basic Sciences</td></tr><tr><td></td><td>Branch Core (CSE / <b>ECE</b>)</td></tr><tr><td></td><td>Elective</td></tr><tr><td></td><td>Engineering Science and Skills</td></tr><tr><td></td><td>HSS/M</td></tr><tr><td></td><td>Miscellaneous</td></tr></table></td></tr>
<tr><td><b>Credits (L:T:P)</b><br><b>(Lecture : Tutorial : Practical)</b></td><td><table><tr><th>Hours</th><th>Component</th></tr><tr><td></td><td>Lecture (1hr = 1 credit)</td></tr><tr><td></td><td>Tutorial (1hr = 1 credit)</td></tr><tr><td>4 hrs</td><td>Practical (2hrs = 1 credit)</td></tr><tr><td><b>2</b></td><td><b>Total Credits</b></td></tr></table></td></tr>
<tr><td><b>Grading Scheme</b></td><td>Select one from the following:<table><tr><td></td><td>4-point scale (A,A-,B+,B,B-,C+,C,D,F)</td></tr><tr><td></td><td></td></tr></table></td></tr>
<tr><td colspan="2"><b>Pre-Requisites</b><br><i>(where applicable, specify exact course names)</i></td></tr>
<tr><td colspan="2"></td></tr>
<tr><td colspan="2"><b>Course Description</b><br><i>A brief description of the course</i></td></tr>
</table>

The objective of this course is to introduce electronics laboratory skills to the students.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of the course, the student is expected to understand and experiment with electronic components and circuits in the lab. The student is expected to have acquired the basic skills to
handle components and operate the instruments with confidence. Be able to design and debug electronic circuits to solve a problem.

**Course Content**

The laboratory content should cover experiments on the following topics:
1. Instruments: digital multimeter, Oscilloscope, Signal generator, Probes, bread boards.
2. I-V characteristics of linear passive devices and their combinations, Charging and discharging of Capacitor circuits.
3. Experiments on filter circuits (Low pass, high pass, bandpass, notch) consisting of combination of R, L and C circuits.
4. I-V characteristics of Diodes, rectifier circuits using diodes, clipper and clamper circuits, LEDs, and zener diode.
5. Operation of DC motors, servo motors, Opamp based amplifiers, filter circuits and other applications.
6. Verify digital logic gates and combinational circuits using IC chips.
7. Develop sequential circuits using digital gates.
8. Finite state machine examples such as vending machine, traffic light controller
9. Introduction of Atmega 16 bit microcontroller and applications of microcontrollers such as reading temperature sensor, driving LEDs, driving servo and DC motors.

**Assessments / Grading**

Midterm exam-30%
Final exam-30%
Quizzes-10%
Assignments-10%
Project-20%

**Text Book / References**

1. Student manual for the Art of electronics - Thomas Hayes and Paul Horowitz
2. The art of electronics - Paul Horowitz and Winfield Hill

**Course Name**

Control Theory

**Course Branch**

Select one from the following:

| | |
|---|---|
| x | ECE |
| | CSE |

**Course Proposer Name(s)**

**Course Instructor Name(s)**

**Course Type** *(Select one)*

Select one from the following:

| | |
|---|---|
| X | Core |
| | Elective |
| | Special Topics Elective* |

** All course types except "Special Topics Elective" go through the process for Academic Senate approval*

**Course Level** *(Select one)*

Select one from the following for elective courses:

| | |
|---|---|
| | Level 1 Elective |
| | Level 2 Elective |
| | N/A |

**Course Category** *(Select one)*

Select one from the following:

| | |
|---|---|
| | Basic Sciences |
| X | Branch Core (CSE / ECE) |
| | Elective |
| | Engineering Science and Skills |
| | HSS/M |
| | Miscellaneous |

**Credits (L:T:P)**
**(Lecture : Tutorial : Practical)**

| **Hours** | **Component** |
|---|---|
| 3 | Lecture (1hr = 1 credit) |
| | Tutorial (1hr = 1 credit) |
| | Practical (2hrs = 1 credit) |
| **3** | **Total Credits** |

**Grading Scheme**

Select one from the following:

| | |
|---|---|
| x | 4-point scale (A,A-,B+,B,B-,C+,C,D,F) |
| | Satisfactory/Unsatisfactory (S / X) |

**Pre-Requisites**
*(where applicable, specify exact course names)*

Signals and Systems

**Course Description**
*A brief description of the course*

This course provides students with an exposure to the theory of Control Systems.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this course, the student is expected to understand the concept of feedback control, perform transient and steady state analysis, design controllers using methods such as root locus, frequency response and state space.

**Course Content**

- Concept of a system/plant, Different types of physical systems, Concept of a controller, Different types of control systems - open/closed loop, time invariant/variant, analog/digital, and linear/nonlinear
- Mathematical modeling of physical systems and their analogues, Importance of concept of analogue systems; Order of the physical systems - first, second, and higher order
- Concept of transfer function, impulse response function, and state space representation
- Transient and Steady state analyses - first, second, and higher order systems
- Specification of controllers and performance criteria
- Control system analysis and design - Root Locus method
- Control system analysis and design - Frequency response method
- Control system analysis and design - State space method

**Assessments / Grading**

Midterm, final, quizzes and homework

**Text Book / References**

**1. “Modern Control Engineering: International Edition”  Katsuhiko Ogata, Pearson Edn.**

<table>
<tr><td><b>Course Name</b></td><td>Analog Circuits</td></tr>
<tr><td><b>Course Branch</b></td><td>Select one from the following:
<table>
<tr><td></td><td>ECE</td></tr>
<tr><td></td><td></td></tr>
</table></td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td>Madhav Rao and Subhajit Sen</td></tr>
<tr><td><b>Course Instructor Name(s)</b></td><td>Madhav Rao, Subhajit Sen</td></tr>
<tr><td><b>Course Type</b> <i>(Select one)</i></td><td>Select one from the following:
<table>
<tr><td>X</td><td>Core (ECE)</td></tr>
<tr><td></td><td></td></tr>
<tr><td></td><td></td></tr>
</table>
<i>* All course types except “Special Topics Elective” go through the process for Academic Senate approval</i></td></tr>
<tr><td><b>Course Level</b> <i>(Select one)</i></td><td>Select one from the following for elective courses:
<table>
<tr><td></td><td>Level 1 Elective</td></tr>
<tr><td></td><td>Level 2 Elective</td></tr>
<tr><td></td><td>N/A</td></tr>
</table></td></tr>
<tr><td><b>Course Category</b> <i>(Select one)</i></td><td>Select one from the following:
<table>
<tr><td></td><td>Basic Sciences</td></tr>
<tr><td>X</td><td>Branch Core (CSE / <b>ECE</b>)</td></tr>
<tr><td></td><td>Elective</td></tr>
<tr><td></td><td>Engineering Science and Skills</td></tr>
<tr><td></td><td>HSS/M</td></tr>
<tr><td></td><td>Miscellaneous</td></tr>
</table></td></tr>
<tr><td><b>Credits (L:T:P)<br>(Lecture : Tutorial : Practical)</b></td><td>
<table>
<tr><th>Hours</th><th>Component</th></tr>
<tr><td>3 hr</td><td>Lecture (1hr = 1 credit)</td></tr>
<tr><td></td><td>Tutorial (1hr = 1 credit)</td></tr>
<tr><td>2 hr</td><td>Practical (2hrs = 1 credit)</td></tr>
<tr><td><b>4</b></td><td><b>Total Credits</b></td></tr>
</table></td></tr>
<tr><td><b>Grading Scheme</b></td><td>Select one from the following:
<table>
<tr><td></td><td>4-point scale<br>(A,A-,B+,B,B-,C+,C,D,F)</td></tr>
<tr><td></td><td></td></tr>
</table></td></tr>
<tr><td colspan="2"><b>Pre-Requisites</b><br><i>(where applicable, specify exact course names)</i></td></tr>
<tr><td colspan="2"></td></tr>
<tr><td colspan="2"><b>Course Description</b><br><i>A brief description of the course</i></td></tr>
</table>

The objective of the course is to provide students an in depth knowledge of discrete transistor devices and circuit design using these transistors.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of the course, the student is expected to design and analyze FET, BJT and Opamp circuits for various applications.

**Course Content**

The course content should cover the following topics:

1. Semiconductor Diodes: Barrier formation in metal semiconductor junctions, PN homo and hetero junctions; CV characteristics and dopant profiling; IV characteristics; Small signal models of diodes; Some Applications of diodes.
2. Field Effect Devices : JFET/HFET, MIS structures and MOSFET operation; JFET characteristics and small signal models; MOS capacitor CV and concept of accumulation, depletion and inversion; MOSFET characteristics and small signal models.
3. Bipolar transistors : IV characteristics and ebers-Moll model; small signal models; Charge storage and transient response.
4. Discrete transistor amplifiers: Common emitter and common source amplifiers; Emitter and source followers, cascode, darlington transistors, power amplifiers.
5. Linear digital ICs, Feedback and Oscillator circuits, Voltage regulators, Two and three terminal devices.
6. The course will include weekly 2 hours of lab component. The lab will include the following topics:
    - BJT and FETs I-V characteristics.
    - Various biasing of BJT and FET circuits (Follower circuit, amplifier circuit, current gain, current source, push-pull)
    - Transistor as switch, darlington superbeta, miller effect, Differential amplifiers.
    - Opamp based circuits includes Comparator, Schmitt trigger, Sawtooth wave oscillator, Active rectifier and clamp circuits.
    - Power supply circuit based on three terminal fixed and variable regulators.

**Assessments / Grading**

Midterm exam-25%
Final exam-25%
Assignments and Quizzes-25%
Lab-25%

**Text Book / References**

1. Electronic devices and circuit theory - Boylestad and Nashelsky
2. Linear Integrated circuits - Roy Choudhury and S. Jain.
3. Student manual for The Art of Electronics - Hayes and Horowitz (Lab part)

| | |
|---|---|
| **Course Name** | **Microprocessor and microcontroller (Elective under VLSI systems)** |
| **Course Branch** | Select one from the following:<br>x ECE<br>CSE |
| **Course Proposer Name(s)** | |
| **Course Instructor Name(s)** | |
| **Course Type** *(Select one)* | Select one from the following:<br>X Core<br>Elective<br>Special Topics Elective*<br>** All course types except "Special Topics Elective" go through the process for Academic Senate approval* |
| **Course Level** *(Select one)* | Select one from the following for elective courses:<br>Level 1 Elective<br>Level 2 Elective<br>N/A |
| **Course Category** *(Select one)* | Select one from the following:<br>Basic Sciences<br>X Branch Core (CSE / ECE)<br>Elective<br>Engineering Science and Skills<br>HSS/M<br>Miscellaneous |
| **Credits (L:T:P)**<br>**(Lecture : Tutorial : Practical)** | **Hours** **Component**<br>3 Lecture (1hr = 1 credit)<br>Tutorial (1hr = 1 credit)<br>2 Practical (2hrs = 1 credit)<br>**4** **Total Credits** |
| **Grading Scheme** | Select one from the following:<br>x 4-point scale (A,A-,B+,B,B-,C+,C,D,F)<br>Satisfactory/Unsatisfactory (S / X) |
| **Pre-Requisites**<br>*(where applicable, specify exact course names)* | |
| Signals and Systems | |
| **Course Description**<br>*A brief description of the course* | |

To learn the architecture, programming, interfacing and system design of microprocessors and microcontrollers. ARM architecture and DSP programming will be introduced.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this course, the student is expected to understand the architecture of 8086 based microprocessor and microcontrollers (8251). They will also be introduced to other processors used in embedded systems, ARM and DSPs.

**Course Content**

**Architecture of Microprocessors**

General definitions of mini computers, microprocessors, micro controllers and digital signal processors. Overview of 8085 microprocessor. Overview of 8086 microprocessor. Signals and pins of 8086 microprocessor

**Assembly language of 8086**

Description of Instructions. Assembly directives. Assembly software programs with algorithms Interfacing with RAMs, ROMs along with the explanation of timing diagrams. Interfacing with peripheral ICs like 8255, 8254, 8279, 8259, 8259 etc. Interfacing with key boards, LEDs, LCDs, ADCs, and DACs etc.

**Architecture of Micro controllers**

Overview of the architecture of 8051 microcontroller. Overview of the architecture of 8096 16 bit microcontroller.

**RISC Based architecture and ARM processors**

**Introduction to DSPs (TI or Analog series)**

**Assessments / Grading**

Midterm, Final, 2 Quizzes, Homework Assignments
Laboratory Experiments

**Text Book / References**

1. D. V. Hall. Micro processors and Interfacing, TMGH. 2'1 edition 2006.
2. 2. Kenneth. J. Ayala. The 8051 microcontroller , 3rd edition, Cengage learning, 2010
3. Digital Signal Processors, Architercure, Implementations and Applications, Sen M. Kuo, Woon-Seng Gan, Prentice Hall

<table>
<tr><td><b>Course Name</b></td><td>Principles of Communication Systems</td></tr>
<tr><td><b>Course Branch</b></td><td>Select one from the following:
<table>
<tr><td>x</td><td>ECE</td></tr>
<tr><td></td><td>CSE</td></tr>
</table></td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td></td></tr>
<tr><td><b>Course Instructor Name(s)</b></td><td></td></tr>
<tr><td><b>Course Type</b> <i>(Select one)</i></td><td>Select one from the following:
<table>
<tr><td>X</td><td>Core</td></tr>
<tr><td></td><td>Elective</td></tr>
<tr><td></td><td>Special Topics Elective*</td></tr>
</table>
<i>* All course types except “Special Topics Elective” go through the process for Academic Senate approval</i></td></tr>
<tr><td><b>Course Level</b> <i>(Select one)</i></td><td>Select one from the following for elective courses:
<table>
<tr><td></td><td>Level 1 Elective</td></tr>
<tr><td></td><td>Level 2 Elective</td></tr>
<tr><td></td><td>N/A</td></tr>
</table></td></tr>
<tr><td><b>Course Category</b> <i>(Select one)</i></td><td>Select one from the following:
<table>
<tr><td></td><td>Basic Sciences</td></tr>
<tr><td>X</td><td>Branch Core (CSE / ECE)</td></tr>
<tr><td></td><td>Elective</td></tr>
<tr><td></td><td>Engineering Science and Skills</td></tr>
<tr><td></td><td>HSS/M</td></tr>
<tr><td></td><td>Miscellaneous</td></tr>
</table></td></tr>
<tr><td><b>Credits (L:T:P)</b><br><b>(Lecture : Tutorial : Practical)</b></td><td>
<table>
<tr><th>Hours</th><th>Component</th></tr>
<tr><td>3</td><td>Lecture (1hr = 1 credit)</td></tr>
<tr><td></td><td>Tutorial (1hr = 1 credit)</td></tr>
<tr><td>2</td><td>Practical (2hrs = 1 credit)</td></tr>
<tr><td><b>4</b></td><td><b>Total Credits</b></td></tr>
</table></td></tr>
<tr><td><b>Grading Scheme</b></td><td>Select one from the following:
<table>
<tr><td>x</td><td>4-point scale<br>(A,A-,B+,B,B-,C+,C,D,F)</td></tr>
<tr><td></td><td>Satisfactory/Unsatisfactory (S / X)</td></tr>
</table></td></tr>
<tr><td colspan="2"><b>Pre-Requisites</b><br><i>(where applicable, specify exact course names)</i></td></tr>
<tr><td colspan="2">Signals and Systems</td></tr>
<tr><td colspan="2"><b>Course Description</b><br><i>A brief description of the course</i></td></tr>
</table>

This course provides students with an introduction to the principles of communication systems.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this course, the student is expected to understand the basics of analog Communication systems. Channel types, propagation characteristics at different frequencies, analog modulation techniques will be covered. Basics of Multiple access techniques, CDMA, Optical communication, Satellite communication system will be covered too.

**Course Content**

1: Review of signals and systems, Frequency domain of signals, Principles of Amplitude Modulation Systems- DSB, SSB and VSB modulations. Angle Modulation., Representation of FM and PM signals. Spectral characteristics of angle modulated signals.
2: Gaussian and white noise characteristics. Noise in amplitude modulation systems. Noise in Frequency modulation systems. Pre-emphasis and De-emphasis.
3:Pulse modulation. Sampling process. Pulse Amplitude and Pulse code modulation (PCM). Differential pulse code modulation. Delta modulation. Noise considerations in PCM. Amplitude shift keying, Frequency shift keying, Phase shift keying. Time Division multiplexing. Digital Multiplexers.
4.Broadband Communication, Multiplexing, Time division multiplexing, Frequency division multiplexing, Multiple access techniques, CDMA, Optical communication, Satellite communication systems

**Assessments / Grading**

Midterm, Final, 2 Quizzes, Homework Assigments
Laboratory Experiments

**Text Book / References**

1. Haykin S., "Communications Systems", John Wiley and Sons, 2001.
2. Proakis J. G. and Salehi M., "Communication Systems Engineering", Pearson Education, 2002. 3.Taub H. and Schilling D.L., "Principles of Communication Systems", Tata McGraw Hill, 2001.

## Signal Processing

*Objectives*

This course presents an introduction of the basic analysis tools and techniques for analog and digital processing of signals. In particular, Fourier and Z-Transform based Linear System theory will be covered in details.

*Proposed Course Contents*

- Discrete-time signals and systems.
- The Z-Transform and its applications.
- Frequency domain analysis.
- Discrete Fourier Transform.
- Fast Fourier Transform algorithms. Digital structures.
- Design of digital filters

## Digital Communications

*Objectives*

Digital communication systems are basic workhorses behind the information age. Examples include the wireless systems such as cellular system (3G/4G) and WiFi , wire-line systems including DSL, cable modems. This course aims to provide an introduction to the basic principles behind the design of these broad-band systems and get an insight into the underlying principles behind the design and analysis of digital communication systems.

*Proposed Course Contents*

- Overview:
  - Digital vs. analog communication, block diagram of a digital communication system.
- Source coding and compression:
  - Huffman and Lempel-Ziv compression algorithms; source entropy; quantization; sampling.
- Communication over erasure channels:
  - Notion of reliable communication; feedback vs. forward error correction coding; linear codes and iterative decoding algorithms; example: fountain codes for reliable packet delivery over the Internet.
- Communication over noisy Gaussian channels:
  - Maximum likelihood detector.
- Communication over band-limited channels:
  - Pulse design for band-limited channels, power/bandwidth trade-off, inter-symbol interference, equalization: maximum likelihood sequence detection via the Viterbi algorithm; orthogonal fre-quency division multiplexing (OFDM).
- Communication over wireless channels:

  - Complex base-band representation of pass-band channels; modelling of multi-path wireless chan-nels; key parameters: delay spread, coherence bandwidth, coherence time, Doppler spread; chan-nel fading.
- Time, frequency and antenna diversity techniques:
  - Introduction to MIMO, flavours of OFDM.

| | |
|---|---|
| **Course Name** | **Mobile Computing** |
| **Course Branch** | Select one from the following: (Place X appropriately)<br>ECE: X<br>CSE: |
| **Course Proposer Name(s)** | Prof. Debabrata Das |
| **Course Instructor Name(s)** | Prof. Debabrata Das and Prof. Jyotsna Bapat |
| **Course Type** *(Select one)* | Select one from the following: (Place X appropriately)<br>Core: X<br>Elective<br>Special Topics Elective*<br>** All course types except “Special Topics Elective” go through the process for Senate approval* |
| **Course Level** *(Select one)* | Select one from the following for elective courses:<br>(Place X appropriately)<br>Level 1 Elective<br>Level 2 Elective<br>N/A |
| **Course Category** *(Select one)* | Select one from the following:<br>(Place X appropriately)<br>Basic Sciences<br>Branch Core (CSE / ECE): ECE<br>Elective<br>Engineering Science and Skills<br>HSS/M<br>Miscellaneous |
| **Credits (L:T:P)**<br>**(Lecture : Tutorial : Practical)** | (Place X appropriately)<br>**Hours** **Component** |

| | |
|---|---|
| | Lecture (1hr = 1 credit): 3 Hours |
| | Tutorial (1hr = 1 credit): 1 Hour |
| | Practical (2hrs = 1 credit) |
| | **Total Credits: 4** |

**Grading Scheme**

Select one from the following:

(Place X appropriately)

| | |
|---|---|
| | 4-point scale : X<br>(A,A-,B+,B,B-,C+,C,D,F) |
| | Satisfactory/Unsatisfactory (S / X) |

**Pre-Requisites**

*(where applicable, specify exact course names)*

It is a core course. Without pre-requisite it can be delivered.

**Course Description**

*A brief description of the course*

With the improvements in wireless technology over the last decade, mobile devices are finding a place in our day-to-day lives in ways beyond imagination. Initially designed to support voice and data, but with the advent of social, economic and immersive use cases, the expectation for coverage and capacity in cellular network has increased many folds. In addition to above expectations, the IoT and personal-computational devices are becoming more pervasive as well as complex. The applications requirements over communication network are rapidly more demanding (with respect to QoS and reliability, etc.). Moreover, the network over which the data traverses are becoming more heterogeneous (devices crosses through different versions of cellular networks (3G-4G-5G and futuristic 6G) and WiFi). These above requirements lead to complicated systems of convergence, which are challenging to study and build. Due to huge demand from customers/users, many companies are working to solve problems related to it.

One way to solve above complex problems this course will focus on concepts of existing architectures (topology + protocols) till 4G cellular network and find out technologies envisioned for better mobility as well as QoS for 5G networks. Furthermore, this course will cover important aspects which makes our complicated applications to work by signaling, secured authentication, hand off, and multimedia-data transportation architecture for heterogeneous networks for convergence.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of the course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this course, the student is expected to:

5. The student have understanding of Radio Access Network (RAN) and Core Network (CN) architecture of Cellular network version 2G, 2.5G, 3G, 4G, 5G.
6. Control Signaling in Cellular network.
7. Voice and Data Path as well as protocol stack under standing in above Cellular networks.
8. Secured authentication of users equipment (Mobile) and IoT devices
9. Mobility hand off
10. 5G's Ultra Reliable and low Latency (URLLC), Massive IoT (mIoT) and Enhanced Mobile Broadband (eMBB) control and data channel protocols.

**Course Content**

The Mobile Computing curriculum will majorly focus on 4G and 5G Cellular Radio Access Network (RANs), Core Network (CN) elements, protocol stacks, signaling, authentication and mobility. To understand above, one need fundamental understanding of 2G, 2.5G, 3G architectures and basic signaling (SS7, SIP).

1. Introduction to Mobile computing (1Hr and 30 mins.)

- What is Mobile computing? What are the challenges faced as an wireless network and mobility over homogeneous as well as heterogeneous networks? What are the fundamental things this course will cover and why?

- Migration of Cellular Network from Circuit Switched to Packet Switched

2. Introduction to GSM Architecture (3-Hrs )

- GSM Architecture: BSC, MSC, HLR. VLR, AUC, EIR

- Hand off in 2G network

- Fundamental functionalities of GSM network Frame Structure of GSM : a) Multi Frame b) Super Frame c) Hyper Frame (d) Burst Structure

- Logical channels

TCH/C, TCH/H, BCCH, FCCH, SCH, PCH, RACH, AGCH, SDCCH

FACCH/F, FACCH/H, SACCH

- Location of Mobile

- Base Station and SIM

- Paging Procedures

- Registration Types and Security

3. Signaling Concepts (3 Hours)

- Signaling System 7 (SS7) architecture and functions

- 2G Signaling

4. Synchronization of User equipment and Cellular network (3 Hours)

- how to detect there is a cell, followed by cell search,

- Cell and PLMN camping (Use case level - Signal Bar, Operator name on screen, followed by Data Icon on-screen)

5. Voice and Data Architectures on Cellular Network (12 Hours)

- GPRS (2.5G) architecture and protocol stacks understanding from circuit switching to packet switching between cellular network and Internet;

- Major elements of GPRS architecture

- Radio Requirements of GPRS

- GPRS Reference points and signaling Layers

- Access Point Name, Interconnection to other GPRS networks of other operators and roaming, Managing, the private Dynamic IP addresses for mobiles and NAT

- 3G Architecture and protocols
- 4G network architecture and protocol stacks, understanding of complete packet switching in cellular network
- 4G Evolved packet core (EPC) elements functions
- 4G communication frame structure (RACH for registration and followed by Data flow)

6. Signaling in 4G network - SIP functionalities. (6 hours)
    - SIP entities (Forking proxies, redirect servers)
    - Message format
    - The SIP request and response – (assignment can be given on Tracing Call Flows, using Ethereal and Seagull)
    - Extending SIP -CPL and CCXML

- SIP State Call control machine (can be given assignments too)
- SDP
- BICC (ISUP) state machine and call control explanation
- Media Control:
    - SIP and H.323
    - H324M
- 4G all the Uplink and Downlink control channels and its functions

7. 5G network (6 hours and 30mins)

- 5G network RAN and CN architecture
- Next Generation Core (NGC) elements and its function
- Enhanced Mobile Broad Band (eMBB) protocols
- Massive IoT (mIoT) requirements and protocols
- Ultra Reliable and Low Latency Communication (URLLC) Protocols

8. Macro and Micro Mobility. (4 Hrs and 30 mins)

- Handoff functions in 5G network for Macro Mobility
- Handoff techniques for micro-mobility in 5G

9. Security used in 5G networks (3 hours)

   -Steps of Security used for UE/IoT device to register in 5G

**Assessments / Grading: Weightage Given below**

1. Midterm Exam = 30%
2. Final Exam = 40%
3. Class Test = 10%
4. Assignments/Project = 20%

**Text Book / References**

1. The GSM Evolution – Mobile Packet Data Services, Peter Stuckmann, Wiley.
2. From GSM to LTE-Advanced Pro and 5G: An Introduction to Mobile Networks and Mobile Broadband, By Martin Sauter, 3rd Edition, Wiley Publication;
3. 3GPP standard Releases 8 to 16 (This will Cover 4G)
4. GSM, GPRS and 3G Technical Standards from ITU
5. 3GPP 5G standard (Evolving till date)

# APPENDIX E: CSE Core Course Syllabus

<table>
<tr><td><b>Course Name</b></td><td>Design and Analysis of Algorithms</td></tr>
<tr><td><b>Course Branch</b></td><td>Select one from the following:
<table>
<tr><td>x</td><td>CSE</td></tr>
<tr><td></td><td>ECE</td></tr>
</table></td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td>Meenakshi D'Souza</td></tr>
<tr><td><b>Course Instructor Name(s)</b></td><td>Meenakshi D'Souza</td></tr>
<tr><td><b>Course Type</b> <i>(Select one)</i></td><td>Select one from the following:
<table>
<tr><td>x</td><td>Core</td></tr>
<tr><td></td><td>Elective</td></tr>
<tr><td></td><td>Special Topics Elective*</td></tr>
</table>
<i>* All course types except “Special Topics Elective” go through the process for Academic Senate approval</i></td></tr>
<tr><td><b>Course Level</b> <i>(Select one)</i></td><td>Select one from the following for elective courses:
<table>
<tr><td></td><td>Level 1 Elective</td></tr>
<tr><td></td><td>Level 2 Elective</td></tr>
<tr><td></td><td>N/A</td></tr>
</table></td></tr>
<tr><td><b>Course Category</b> <i>(Select one)</i></td><td>Select one from the following:
<table>
<tr><td></td><td>Basic Sciences</td></tr>
<tr><td>x</td><td>Branch Core (CSE / ECE)</td></tr>
<tr><td></td><td>Elective</td></tr>
<tr><td></td><td>Engineering Science and Skills</td></tr>
<tr><td></td><td>HSS/M</td></tr>
<tr><td></td><td>Miscellaneous</td></tr>
</table></td></tr>
<tr><td><b>Credits (L:T:P)<br>(Lecture : Tutorial : Practical)</b></td><td>
<table>
<tr><th>Hours</th><th>Component</th></tr>
<tr><td>3</td><td>Lecture (1hr = 1 credit)</td></tr>
<tr><td>1</td><td>Tutorial (1hr = 1 credit)</td></tr>
<tr><td></td><td>Practical (2hrs = 1 credit)</td></tr>
<tr><td></td><td><b>Total Credits</b></td></tr>
</table></td></tr>
<tr><td><b>Grading Scheme</b></td><td>Select one from the following:
<table>
<tr><td>x</td><td>4-point scale (A,A-,B+,B,B-,C+,C,D,F)</td></tr>
<tr><td></td><td>Satisfactory/Unsatisfactory (S / X)</td></tr>
</table></td></tr>
<tr><td colspan="2"><b>Pre-Requisites</b><br><i>(where applicable, specify exact course names)</i></td></tr>
<tr><td colspan="2">Discrete Mathematics, Data Structures and Algorithms</td></tr>
</table>

**Course Description**

*A brief description of the course*

As one of the core courses in the iM. Tech. program, Design and Analysis of Algorithms is meant to provide thorough exposure to many fundamental algorithms and algorithm design techniques in Computer Science.

This course is a follow-up of the course on Data Structures and will cover most of the fundamental techniques for design and analysis of algorithms. The emphasis of this course is on algorithm design techniques along with their proofs and theoretical foundations. This course will also introduce complexity classes P and NP and NP-complete problems.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this course, the student is expected to know:

11. The fundamentals of the design of algorithms for various problems and analyzing them in terms of time and memory they consume to solve a particular problem.
12. Algorithms for various classical problems in Computer Science ranging from sorting, graph algorithms, algorithms manipulating numbers and strings etc.
13. Algorithm design techniques and strategies like divide-and-conquer, dynamic programming and greedy algorithms along with several problems using these techniques, proofs of correctness.
14. NP-completeness and some NP-complete problems.

**Course Content**

1. Introduction to algorithms, examples illustrating their role in computing, notations used to represent their time and space complexity.
2. Divide and conquer techniques
3. Dynamic programming
4. Greedy algorithms
5. Graph algorithms: elementary graph algorithms, minimum spanning trees, single-source shortest paths, all-pairs shortest paths, maximum flow.
6. Number-theoretic algorithms
7. String matching algorithms
8. NP-completeness

**Assessments / Grading**

Class tests, mid-semester and final exam, mini project.

**Text Book / References**

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, Prentice-Hall, 3rd edition, 2009.

| | |
|---|---|
| **Course Name** | Discrete Mathematics |
| **Course Branch** | Select one from the following:<br>ECE<br>✓ CSE |
| **Course Proposer Name(s)** | |
| **Course Instructor Name(s)** | |
| **Course Type** *(Select one)* | Select one from the following:<br>✓ Core<br>Elective<br>Special Topics Elective*<br>** All course types except "Special Topics Elective" go through the process for Academic Senate approval* |
| **Course Level** *(Select one)* | Select one from the following for elective courses:<br>Level 1 Elective<br>Level 2 Elective<br>N/A |
| **Course Category** *(Select one)* | Select one from the following:<br>Basic Sciences<br>✓ Branch Core (CSE / ECE)<br>Elective<br>Engineering Science and Skills<br>HSS/M<br>Miscellaneous |
| **Credits (L:T:P)**<br>**(Lecture : Tutorial : Practical)** | **Hours** **Component**<br>Lecture (1hr = 1 credit)<br>Tutorial (1hr = 1 credit)<br>Practical (2hrs = 1 credit)<br>**Total Credits**<br>(L: T) = (3:1) |
| **Grading Scheme** | Select one from the following:<br>4-point scale (A,A-,B+,B,B-,C+,C,D,F)<br>Satisfactory/Unsatisfactory (S / X)<br>4-point scale |
| **Pre-Requisites**<br>*(where applicable, specify exact course names)* | |
| | |
| **Course Description**<br>*A brief description of the course* | |

Discrete mathematics is the study of mathematical structures that are discrete in the sense that they assume only distinct, separate values, rather than in a range of values. It deals with the mathematical objects that are widely used in all most all fields of computer science, such as programming languages, data structures and algorithms, cryptography, operating systems, compilers, computer networks, artificial intelligence, image processing, computer vision, natural language processing, etc. This course covers elementary discrete mathematics that is required for a computer science, engineering or information technology degree.

**Course Outcomes**

*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

One of the main purpose of this course is to enable the students to learn a particular set of mathematical facts and how to apply them. More importantly, this course will teach the student how to think logically and mathematically. This course will carefully blend and balance the following four themes, required from any successful Discrete Mathematics course:

1. **Mathematical reasoning**: students will learn mathematical reasoning in order to read, comprehend, and construct mathematical arguments. The students will learn various proof techniques; at the end of the course, they are expected to know how to apply them to a varieties of problems.

2. **Combinatorial Analysis:** an important problem-solving skill required from any CS student is the ability to count or enumerate objects. The students will learn the basic techniques of counting. At the end of the course, they are expected to perform combinatorial analysis to solve counting problems.

3. **Discrete Structures:** students will be taught how to work with discrete structures, which are the abstract mathematical structures used to represent discrete objects and relationships between these objects. The main discrete structures that students will learn in this course include sets, permutations, relations and graphs. At the end of this course, students are expected to know the fundamental properties of these discrete structures.

4. **Applications and Modeling*:*** Discrete mathematics has tremendous applications to computer science and data networking, as well as diverse areas as chemistry, biology, linguistics, geography, business, and the Internet. Modelling with discrete mathematics is an extremely important problem-solving skill and this course will provide the students ample scope to develop this skill by constructing their own models for solving some real-world problems.

**Course Content**

Roughly the syllabus should be divided into following four units:

1. **Logic and Proof Techniques:** Propositional logic, logical connectives, truth tables, normal forms (conjunctive and disjunctive), validity; predicate logic, limitations of predicate logic, universal and existential quantifcation, modus ponens and modus tollens, notions of implication, converse, inverse, contrapositive, negation, and contradiction; the structure of formal proofs; direct proofs; proof by counter example; proof by contraposition; proof by contradiction; mathematical induction; strong induction; recursive mathematical definitions.

2. **Set Theory:** Definition of set; relations, equivalence relations and equivalence classes, posets, chains and well-ordered sets; functions, recursive functions, sequences and summations; cardinality and countability.

3. **Combinatorics:** Principles of addition and multiplication, arrangements, permutation and combinations, multinomial theorem, partitions and allocations, Pigeonhole principle, inclusion-exclusion principle; generating functions, recurrence relations.

4. **(Algorithmic) Graph Theory**: graphs and graph models, graph isomorphism, connectivity, Euler and Hamilton paths, shortest path problems, planar graphs and graph coloring

5. Abstract Algebra: Groups, Rings, Modules, Ideals, Fields and examples of finite fields

**Assessments / Grading**

Suggested assessment criteria:

1. Class tests --- 20%
2. Assignments --- 20%
3. Mid-sem exam --- 20%
4. End-sem exam --- 40%

**Text Book / References**

The following is the highly recommended standard textbook for Discrete mathematics (now Indian edition is also available):

- Discrete Mathematics and its Applications by Kenneth H Rosen, 7th Edition, McGraw Hill, 2014.

The following textbooks and study materials are recommended as additional reference:

- Elements of Discrete Mathematics, by C. L. Liu, second edition 1985, McGraw-Hill Book Company. Reprinted 2000.
- Proper web notes (NPTEL notes by Prof. Kamala Krithivasan are available on discrete mathematics).
- Discrete Math for Computer Science Students by K. Bogart, S. Drysdale, C. Stein (freely available online).

- Discrete Mathematics by Laszlo Lovasz, Jozsef Pelikan, Katalin L. Vesztergombi, Springer 2003.

**Course Name** | **Programming Languages (moved to Elective)**

**Course Branch**

Select one from the following:

| | |
|---|---|
| X | CSE |
| | ECE |

**Course Proposer Name(s)**

**Course Instructor Name(s)**

**Course Type** *(Select one)*

Select one from the following:

| | |
|---|---|
| X | Core |
| | Elective |
| | Special Topics Elective* |

** All course types except "Special Topics Elective" go through the process for Academic Senate approval*

**Course Level** *(Select one)*

Select one from the following for elective courses:

| | |
|---|---|
| | Level 1 Elective |
| | Level 2 Elective |
| X | N/A |

**Course Category** *(Select one)*

Select one from the following:

| | |
|---|---|
| | Basic Sciences |
| CSE | Branch Core (CSE / ECE) |
| | Elective |
| | Engineering Science and Skills |
| | HSS/M |
| | Miscellaneous |

**Credits (L:T:P)**
**(Lecture : Tutorial : Practical)**

| **Hours** | **Component** |
|---|---|
| 40 | Lecture (1hr = 1 credit) |
| | Tutorial (1hr = 1 credit) |
| | Practical (2hrs = 1 credit) |
| | **Total Credits** |

**Grading Scheme**

Select one from the following:

| | |
|---|---|
| X | 4-point scale (A,A-,B+,B,B-,C+,C,D,F) |
| | Satisfactory/Unsatisfactory (S / X) |

**Pre-Requisites**
*(where applicable, specify exact course names)*

Data-structures and algorithms

**Proficiency in programming with at least a couple of programming languages (e.g. C, Java, etc.)**

**Course Description**
*A brief description of the course*

The objective of this course is to enable the student to view any PL in terms of its fundamental features. The student will learn a wide range of PL features and programming idioms which allow *elegant* as well as *correct* implementations of computer programs. The course will provide tools to rigorously represent and analyse a language both in terms of what is a well-formed program in a PL (syntax), as well as what a well-formed program *means* (semantics). This will be achieved predominantly through hands-on implementations of interpreters and program analysis tools.

**Course Outcomes**
*Course Outcomes are statements that describe what students are expected to know, and be able to do at the end of each course. These relate to the skills, knowledge, and behavior that students acquire in their progress through the course.*

At the end of this course, the student is expected to:

Know

- the important classification of programming languages in terms of their typing, scoping, paradigms etc.
- Object oriented programming concepts like polymorphism, dynamic binding etc.
- Sub-typing as a PL concept
- Functional programming using Ocaml programming language: Recursion, Higher order functions, closures
- Methods of specification and implementation of programming languages

Be able to

- Write non-trivial programs in Ocaml using functional approach
- Implement interpreters for small programming languages to implement specifications.

**Course Content**

- Review of programming concepts and paradigms
- Introduction to functional programming
- Specification and implementation of programming languages

**Assessments / Grading**

**Mid-semester examination**
**: 35**
**Final-semester examination**
**: 35**
**Class tests/assignments**
**: 15**

**Project**
**: 15**

**(The above is subject to minor modification based on the dynamics of the course.)**

**Text Book / References**

Essentials of Programming Languages – Friedman and Wand
Programming Languages – Ravi Sethi
Types in Programming Languages – Benjamin Pierce
Online resources on functional programming in Ocaml
Relevant literature

**Database Systems (Theory and Lab)**

*Objectives*

The ACM document on Computer Science covers Database Systems area under the title "Information Management." Information Management (IM) plays a critical role in almost all areas where comput-ers are used. This area includes the capture, digitization, representation, organization, transformation, and presentation of information; algorithms for efficient and effectiveaccess and updating of stored information, data modeling and abstraction, and physical file storage techniques. It also encompasses information security, privacy, integrity, and protection in a shared environment. The student needs to be able to develop conceptual and physical data models, determine what IM methods and techniques are appropriate for a given problem, and be able to select and implement an appropriate IM solution that reflects all suitable constraints, including scalability and usability.

*Proposed Course Contents*

- Introduction to database systems:
    - History and motivation for database systems; components of database systems; DBMS functions; database architecture and data independence.
- Database design:
    - Data modeling; conceptual models (such as entity-relationship or UML); object-oriented model;
    - Relational data model:
        Functional dependency; decomposition of a schema; lossless-join and dependency-preservation properties of a decomposition; candidate keys, superkeys, and closure of a set of attributes; normal forms (1NF, 2NF, 3NF, BCNF); relational algebra;
    - Mapping conceptual schema to a relational schema.
- DBMS:
    - Components of a DBMS; indexing; transaction processing; concurrency control; query processing; failure and recovery.
- Database programming:
    - SQL, other procedural access methods (JDBC, stored procedures, frameworks, etc.).
- Other topics (at an introductory level):
    - Data warehousing, specialized databases.

This course is to be accompanied by a Database laboratory session.

## Software Engineering

Objectives

This course is intended to provide foundational knowledge in the area of Software Engineering and help them to understand critical concepts encountered while dealing with complex software projects. The course will cover both process and technical aspects of software engineering and will form the basis for further specilized courses in this area.

Proposed Course Contents

- Managing software development:
    - Lifecycle models, quality control, configuration management, project management, personal and team software process, software metrics.
- Software development:
    - Requirements management, design and analysis techniques, architectures of software systems, software testing.

The following textbooks are recommended for this course:

- “Software Architecture in Practice,” Second Edition, by Len Bass, Paul Clements, and Rick Kazman, Addison-Wesley 2003.
- “Models of Software Systems,” by D. Garlan, J. Wing, O. Celiku, and D. Kroening. Draft book.
- “Essentials of Software Engineering,” by Tsui and Karam. (T&K), 2007, Jones and Barrlet Publishing, ISBN-13: 978-07637-3537-1.
- “Software Engineering: A Practitioner’s Approach,” by Roger S. Pressman, sixth Edition, 2005, ” McGraw Hill, ISBN 0-07-301933-X.
- Additional reading material specific to topics being handled.

## Introduction to Automata Theory and Computability

**Course Objectives**:
(i) To understand the basic concepts in the theory of computation and automata theory.
(ii) To understand the formal description of a computer as a Turing Machine.
(iii) To understand the formal equivalence of formal languages and problems, and results like the Halting Problem

**Textbooks:**
(i) *Computability: Computable Functions, Logic, and the Foundations of Mathematics*, 2nd ed., by Epstein and Carnielli, Wadsworth/Thomson Learning, ISBN 0-534-54644-7
(ii) *Introduction to Automata Theory, Languages, and Computation*, 2nd ed., by Hopcroft, Motwani, and Ullman, Addison-Wesley, ISBN 0-201-44124-1.

- **Letter Grade Distribution**: (Subject to change)

A = 80th percentile and above
B = 50th–80th percentile
C = 10th–50th percentile

- **Course Outline**: Major Topics (take this as a plan, not a contract)

– General mathematical background
– Finite Automata, Regular Expressions
– Turing Machines
– Computability
– Church's Thesis
– Primitive Recursive Functions
– Partial Recursive Functions
– Undecidability

# APPENDIX F: List of Electives

| S. No. | Elective name | Specialization | Pre-requisites | Faculty | Semesters |
|---|---|---|---|---|---|
| 1 | Artificial General Intelligence | AIML | | Srinath | 6 & 8 |
| 2 | Artificial Intelligence | AIML | iMTech Introduction to Automata Theory | Shrisha | 7 & 9 |
| 3 | Automatic Speech Recognition | AIML | ML, and Maths for ML | Ram | 6 & 8 |
| 4 | Deep learning for Automatic Speech Recognition | AIML | Automatic Speech Recognition | Ram | 7 & 9 |
| 5 | Digital Image Processing | AIML | ML, and Maths for ML | NS | 6 & 8 |
| 6 | Geographical Information Systems | AIML | | Uttam | 7 & 9 |
| 7 | Machine Learning | AIML | | GSR | 5, 7 & 9 |
| 8 | Maths for ML | AIML | | Ram & Viswanath | 5, 7 & 9 |
| 9 | Multi-agent systems | AIML | | Srinath | 6 & 8 |
| 10 | Natural Language Processing | AIML | | GSR & Srinath | 6 & 8 |
| 11 | Network Science for the web | AIML | | Srinath | 7 & 9 |
| 12 | Neural Networks and Reinforcement Learning | AIML | ML, and Maths for ML | GSR | 7 & 9 |
| 13 | Optimization, Learning and Cognition-2 | AIML | | GNS | 7 & 9 |
| 14 | Optimization, learning, and Cognition-1 | AIML | | GNS | 6 & 8 |
| 15 | Probabilistic Graphical Models | AIML | ML, and Maths for ML | GSR | 6 & 8 |
| 16 | Robotics | AIML | | Sachit | 5, 7 & 9 |
| 17 | Visual recognition | AIML | ML, and Maths for ML | DJ | 6 & 8 |
| 18 | Advanced Visual Recognition | AIML | Visual recognition | DJ | 7 & 9 |
| 19 | Advanced Qualitative Research Methods | DT | | JS | |
| 20 | Cyberspace, Globalization, and Location | DT | | | |
| 21 | Digital Platforms: Technology and Business Components | DT | | | |
| 22 | Digital Sociology | DT | | PM | |
| 23 | Dynamics of the Information Technology Industry | DT | | | |

| | | | | | |
|---|---|---|---|---|---|
| 24 | E-Governance Application Design | DT | | | |
| 25 | Human Computer Interaction | DT | | PM & Linus K | |
| 26 | Information and Communication Technology Policy and Regulation | DT | | | |
| 27 | Privacy in the Digital Age | DT | | JS/Sridhar/TKS | |
| 28 | Social Complexity and Systems Thinking | DT | | | |
| 29 | Social Media Communication | DT | | PM | |
| 30 | Techno-economics of networks | DT | | Sridhar | |
| 31 | Technology and Development | DT | | JS | |
| 32 | Technology and Society | DT | | BC | |
| 33 | The City | DT | | Balaji | |
| 34 | The Web and the Mind | DT | | Srinath | |
| 35 | Advanced Computer Networks | NC | | Samar | 6 & 8 |
| 36 | Advanced Cyber Security | NC | Cyber security – Fundamentals and Techniques | Mohan Ram and Harish S | 7 & 9 |
| 37 | Cyber security – Fundamentals and Techniques | NC | | Mohan Ram and Harish S | 6 & 8 |
| 38 | Internet of Things | NC | | JB & DD | 6 & 8 |
| 39 | Mathematical Analysis of Networks | NC | | TA | 6 & 8 |
| 40 | Network Security | NC | | | 7 & 9 |
| 41 | Software Defined Network and Network Function Virtualization | NC | | Adjunct | 7 & 9 |
| 42 | Wireless Access Networks | NC | | DD | 6 & 8 |
| 43 | Wireless Communication | NC | | PD & JB | 7 & 9 |
| 44 | Advanced Algorithms | TSCD | | MVN | 6 & 8 |
| 45 | Advanced Computer Graphics | TSCD | Computer Graphics | JN & TKS | 6 & 8 |
| 46 | Advanced Data Visualization | TSCD | Data Visualization | JN | 6 & 8 |
| 47 | Approximation Algorithms | TSCD | | MVN | 7 & 9 |
| 48 | Computational Geometry | TSCD | iMTech - Design and Analysis of Algorithms | PA | 7 & 9 |
| 49 | Computer Graphics | TSCD | | JN & TKS | 6 & 8 |
| 50 | Computing on Private Data | TSCD | Foundations of Cryptography | AC | 6 & 8 |
| 51 | Cryptographic Engineering | TSCD | iMTech - Discrete Maths and | SV | 7 & 9 |

| | | | | | |
|---|---|---|---|---|---|
| | | | Design and Analysis of Algorithms | | |
| 52 | Data Modelling | TSCD | iMTech - Database Systems | RC | 6 & 8 |
| 53 | Data Visualization | TSCD | | JN | 5, 7 & 9 |
| 54 | Design Patterns and Enterprise system development | TSCD | | | |
| 55 | Foundations of Cryptography | TSCD | iMTech - Discrete Maths and Design and Analysis of Algorithms | AC | |
| 56 | Graph Theory | TSCD | iMTech - Discrete Maths, Design and Analysis of Algorithms | PA & MDS | 6 & 8 |
| 57 | OOAD UML and Intro to Web 2.0 | TSCD | | K V Dinesha | 6 & 8 |
| 58 | Privacy-Preserving Machine Learning | TSCD | | | |
| 59 | Program Analysis for Software Engineering | TSCD | | SKC | 7 & 9 |
| 60 | Software production engineering | TSCD | | BT | 6 & 8 |
| 61 | Software Testing | TSCD | | MDS | 7 & 9 |
| 62 | Topological Data analysis | TSCD | | Amit C | 6 & 8 |
| 63 | Blockchain and Cryptocurrencies | TSCD | | AC | |
| 64 | Algorithmic Thinking | TSCD | | C Pandu Rangan | 6 & 8 |
| 65 | Advanced Analog Design | VLSI | Analog CMOS VLSI Design | CP & Sen | 8 |
| 66 | Advanced ARM architectures | VLSI | | Girish | 7 & 9 |
| 67 | Analog Circuits and Systems | VLSI | iMtech - Electronic Devices and Circuit Theory | Sen | 6 |
| 68 | Analog CMOS VLSI Design | VLSI | iMtech - Electronic Devices and Circuit Theory | CP | 7 & 9 |
| 69 | ASIC design | VLSI | Digital CMOS VLSI Design | NR & Subir | 6 & 8 |
| 70 | Device Driver Development | VLSI | Virtual | KKS | 7 & 9 |

| | | | | | |
|---|---|---|---|---|---|
| | | | Machines | | |
| 71 | Digital CMOS VLSI Design | VLSI | iMtech - Electronic Devices and Circuit Theory | MR | 5 & 7 |
| 72 | Functional Verification of SOCs | VLSI | | Subir | 6 & 8 |
| 73 | High level synthesis and optimization of Digital Circuits | VLSI | | Subir | 7 & 9 |
| 74 | Processor Architecture | VLSI | | NR | 6 & 8 |
| 75 | RTOS | VLSI | | Girish | 7 |
| 76 | System design with FPGA (2 credits) | VLSI | | NR | 5 & 7 |
| 77 | Testing and Design for Testability | VLSI | | Subir | 7 & 9 |
| 78 | Virtual machines | VLSI | | KKS | 8 |
| 79 | VLSI Subsystem | VLSI | Digital CMOS VLSI Design | MR | 6 & 8 |

| | | | |
|---|---|---|---|
| NR | Nanditha Rao | | |
| Girish | Girish Kumar | DD | Debabrata Das |
| CP | Chetan Parikh | TA | Tricha Anjali |
| Sen | Subhajit Sen | PA | Pradeesha Ashok |
| NS | Neelam Sinha | PD | Priyanka Das |
| GSR | G Srinivas Raghavan | MVN | Muralidhara V N |
| DJ | Dinesh Jayagopi | PM | Preeti Mudaliar |
| JS | Janaki Srinivasan | TKS | T K Srikanth |
| AC | Ashish C | BC | Bidisha C |
| SV | Srinivas Vivek | SKC | Sujit Kumar C |
| RC | Chandrashekar R | BT | B Thangaraju |
| JN | Jaya Nair | MDS | Meenakshi D'Souza |
| JB | Jyotsna Bapat | MR | Madhav Rao |
| KKS | Subramanian K K | Subir | Subir Roy |