# iMTech (Integrated M.Tech.) Curriculum

## (Effective from academic year 2015-16)

International Institute of Information Technology
Bangalore – 560100
February 2017

# Contents

**Document Control**

| Version Number | Revision Date | Change Description |
|---|---|---|
| 1 | April 15, 2015 | Incorporation of changes for iMTech (CSE) and iMTech (ECE) approved in the 44th and 45th Meeting of the Senate of IIITB, held on Feb 11, 2015 and April 8, 2015, respectively |
| 2 | Feb 24, 2017 | Updated document with changes approved in the in 55th Meeting of the Senate of IIITB, held on February 15, 2017 |
| | | |
| | | |
| | | |

# 1 Introduction

This is the curriculum document the 5-year Integrated M.Tech. (iMTech) programme effective from joining IIITB from the year 2015 onwards.

## Overall iMTech Programme Structure

| | |
|---|---|
| **Semester 1 (15 weeks)** | **18 credits**<br>• 6 common core courses |
| **Semester 2 (15 weeks)** | **22 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)** | **18 credits**<br>• 4 common core courses<br>• 1 CSE / ECE core course |
| **Semester 5 (15 weeks)** | **19 credits**<br>• 1 common core courses<br>• 4 CSE / ECE core course |
| **Semester 6 (15 weeks)** | **23 credits**<br>• 1 CSE / ECE core courses<br>• 4 CSE / ECE elective courses<br>• 1 HSS/M elective course |
| **Semester 7 (15 weeks)** | **20 credits**<br>• 4 CSE / ECE elective courses<br>• 1 HSS/M elective course |
| **Semester 8 (15 weeks)** | **20 credits**<br>• 5 CSE / ECE elective courses |
| **Semester 9 (15 weeks)** | **20 credits**<br>• Masters Project / Thesis |
| **Semester 10 (15 weeks)** | **20 credits**<br>• Masters Project / Thesis |
| **Total** | **200 credits** |

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

- Basic Engineering Science and Skills
- Mathematics and Basic Sciences
- Humanities and Social Sciences / Management
- Engineering Core
- Stream Core (for CSE and ECE, respectively)
- Electives
- Masters Project / Thesis
- Others

.

## 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 (within each branch or across branches) 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.

## 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**

The courses are grouped under the following categories:

- Basic Engineering Science / Skills
- Elective
- Engineering Core
- Branch Core (for CSE and ECE, respectively)
- HSS/M
- Masters Project
- Mathematics and Basic Sciences
- Others

# 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 | | | |
|---|---|---|---|
| **Course Categories** | **Number of Courses** | **Total Credits** | **Percent** |
| Basic Engineering Science / Skills | 4 | 16 | 8% |
| CSE Core | 9 | 26 | 13% |
| Elective | 13 | 52 | 26% |
| Engineering Core | 7 | 18 | 9% |
| HSS/M Core | 2 | 8 | 4% |
| HSS/M Elective | 2 | 8 | 4% |
| Masters Project | 2 | 40 | 20% |
| Mathematics and Basic Sciences | 10 | 28 | 14% |
| Others | 4 | 4 | 2% |
| **Grand Total** | **53** | **200** | |

**Table 2: CSE Course Distribution**

| Electronics and Communication Engineering | | | |
|---|---|---|---|
| **Course Categories** | **Number of Courses** | **Total Credits** | **Percent** |
| Basic Engineering Science / Skills | 4 | 16 | 8% |
| ECE Core | 15 | 34 | 17% |
| Elective | 11 | 44 | 22% |
| Engineering Core | 7 | 18 | 9% |
| HSS/M Core | 2 | 8 | 4% |
| HSS/M Elective | 2 | 8 | 4% |
| Masters Project | 2 | 40 | 20% |
| Mathematics and Basic Sciences | 10 | 28 | 14% |
| Others | 4 | 4 | 2% |
| **Grand Total** | **57** | **200** | |

**Table 3: ECE Course Distribution**

## 4.1 Basic Engineering Science and Skills

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 I | 4 | 2:0:4:4 |
| Digital Design | 4 | 3:1:0:4 |
| Programming II | 4 | 3:0:2:4 |
| Signals and Systems | 4 | 3:1:0:4 |

**Table 4: Basic Engineering Sciences / Skills**

## 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 |
|---|---|---|
| Chemistry | 4 | 3:0:2:4 |
| 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 |
| Mathematics - 4 | 4 | 3:1:0:4 |
| Physics - 2 | 4 | 3:0:2:4 |

**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 |
|---|---|---|
| Economics | 4 | 3:1:0:4 |
| History of Ideas | 4 | 4:0:0:4 |

**Table 6: HSS/M**

## 4.4 Other General Core Courses

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 |
|---|---|---|
| Physical Education 1 | 0 | 0:0:0:0 |
| English | 2 | 2:0:0:2 |
| Physical Education 2 | 0 | 0:0:0:0 |
| Technical Communication | 2 | 2:0:0:2 |

**Table 7: Other courses**

## 4.5 Engineering Core

The courses under the Engineering Core category are mandatory for both CSE stream and ECE stream students. The list of courses is given in Table 8 below. All these courses are Level 1 courses.

| Engineering Core Course Name | Credits | L:T:P:C |
|---|---|---|

| | | |
|---|---|---|
| Data Structures and Algorithms | 6 | 3:1:4:6 |
| Computer Networks | 4 | 3:1:0:4 |
| Computer Architecture | 4 | 3:0:2:4 |
| Operating Systems | 4 | 3:0:2:4 |

**Table 8: Engineering Core**

## 4.6 CSE Core

Table 9 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 | 4 | 3:1:0:4 |
| Formal Languages and Automata Theory | 4 | 3:1:0:4 |
| Software Engineering | 4 | 3:0:2:3 |
| Computer Graphics | 3 | 3:0:0:3 |
| Database Systems | 4 | 3:0:2:4 |
| Programming Languages | 3 | 3:0:0:3 |

**Table 9: CSE Core**

## 4.7 ECE Core

Table 10 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 Electronics | 4 | 2:0:2:4 |
| Electronic Devices & Circuit Theory * | 4 | 3:0:2:4 |
| Principles of Communication Systems* | 4 | 3:0:2:4 |
| Analog CMOS Design* | 4 | 3:0:2:4 |
| Microprocessors and Microcontrollers* | 4 | 3:0:2:4 |
| Signal Processing | 3 | 3:0:0:0 |
| Control Theory* | 3 | 3:0:0:0 |
| Digital Communication | 4 | 3:0:2:4 |
| EMT/Antenna Theory* | 4 | 3:1:0:0 |

**Table 10: ECE Core Courses**

## 4.8 Electives

Students are required to select at least 60% of the electives from the branch they belong to. Electives can be either Level 2 or Level 3 courses. All Level 3 courses shall have at least of one of the existing Level 2 course as a pre-requisite. CSE branch students need to take a total of 13 electives (52 credits) and ECE branch students need to take a total of 11 electives (44 credits). CSE branch students must select a minimum of 8 of their electives from CSE branch while ECE students must do 7 electives from the ECE branch.

# 5 Specialization

iMTech students can earn specialization in specified areas within the branch provided they do 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.

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

- Microelectronics and VLSI
- Networking and Communication
- Signal Processing and Pattern Recognition (will work across ECE and CSE streams)

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

- Theoretical Computer Science
- Data Science
- Software Engineering

# 6 Masters Project / Thesis

Students have two options for meeting the Masters Project requirement:

1. (Option A) One semester (20 credits) of project work 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 11 below.

| Course Name | Credits | Course Category | Level |
|---|---|---|---|
| **SEMESTER 1** | **18** | | |
| Chemistry | 3 | Mathematics and Basic Sciences | Level 1 |
| Chemistry Lab | 1 | Mathematics and Basic Sciences | Level 1 |
| Mathematics - 1 | 4 | Mathematics and Basic Sciences | Level 1 |
| Programming I | 4 | Basic Engineering Science / Skills | Level 1 |
| Physical Education 1 | 0 | Others | Level 1 |
| English | 2 | Others | Level 1 |
| Economics | 4 | HSS/M Core | Level 1 |
| **SEMESTER 2** | **22** | | |
| Mathematics - 2 | 4 | Mathematics and Basic Sciences | Level 1 |
| Digital Design | 4 | Basic Engineering Science / Skills | 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 |
| History of Ideas | 4 | HSS/M Core | Level 1 |
| Physical Education 2 | 0 | Others | Level 1 |
| **SEMESTER 3** | **20** | | |
| Mathematics - 3 | 4 | Mathematics and Basic Sciences | Level 1 |
| Computer Architecture | 3 | Engineering Core | Level 1 |
| Computer Architecture Lab | 1 | Engineering Core | Level 1 |
| Programming II | 4 | Basic Engineering Science / Skills | Level 1 |
| Physics - 1 | 3 | Mathematics and Basic Sciences | Level 1 |
| Physics Lab - 1 | 1 | Mathematics and Basic Sciences | Level 1 |
| Discrete Mathematics | 4 | CSE Core | Level 1 |
| **SEMESTER 4** | **18** | | |
| Mathematics - 4 | 4 | Mathematics and Basic Sciences | Level 1 |
| Signals and Systems | 4 | Basic Engineering Science / Skills | Level 1 |
| Physics - 2 | 3 | Mathematics and Basic Sciences | Level 1 |
| Physics Lab - 2 | 1 | Mathematics and Basic Sciences | Level 1 |
| Design and Analysis of Algorithms | 4 | CSE Core | Level 1 |
| Technical Communication | 2 | Others | Level 1 |
| **SEMESTER 5** | **19** | | |
| Formal Languages and Automata Theory | 4 | CSE Core | Level 1 |
| Software Engineering | 3 | CSE Core | Level 1 |
| Software Engineering Lab | 1 | CSE Core | Level 1 |
| Operating Systems | 3 | Engineering Core | Level 1 |
| Operating Systems Lab | 1 | Engineering Core | Level 1 |
| Computer Graphics | 3 | CSE Core | Level 1 |

| Course Name | Credits | Course Category | Level |
|---|---|---|---|
| Database Systems | 3 | CSE Core | Level 1 |
| Database Lab | 1 | CSE Core | Level 1 |
| **SEMESTER 6** | **23** | | |
| Programming Languages | 3 | CSE Core | Level 1 |
| Elective - 1 | 4 | Elective | Level 1 |
| Elective - 2 | 4 | Elective | Level 2 |
| Elective - 3 | 4 | Elective | Level 2 |
| Elective - 4 | 4 | Elective | Level 2 |
| HSS/M Elective - 1 | 4 | HSS/M Elective | Level 2 |
| **SEMESTER 7** | **20** | | |
| HSS/M Elective - 2 | 4 | HSS/M Elective | Level 2 / Level 3 |
| Elective - 5 | 4 | Elective | Level 2 / Level 3 |
| Elective - 6 | 4 | Elective | Level 2 / Level 3 |
| Elective - 7 | 4 | Elective | Level 2 / Level 3 |
| Elective - 8 | 4 | Elective | Level 2 / Level 3 |
| **SEMESTER 8** | **20** | | |
| 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 |
| Elective - 13 | 4 | Elective | Level 2 / Level 3 |
| **SEMESTER 9** | **20** | | |
| M.Tech. Project / Thesis | 20 | Masters Project | Masters Project |
| **SEMESTER 10** | **20** | | |
| M.Tech. Project / Thesis | 20 | Masters Project | Masters Project |

**Table 11: Course Sequencing for iMTech (CSE)**

# 8 Course Sequencing for iMTech (ECE)

The course sequencing for the ECE branch is given in Table 12 below:

| Course Name | Credits | Course Category | Level |
|---|---|---|---|
| **Semester 1** | **18** | | |
| Chemistry | 3 | Mathematics and Basic Sciences | Level 1 |
| Chemistry Lab | 1 | Mathematics and Basic Sciences | Level 1 |
| Mathematics - 1 | 4 | Mathematics and Basic Sciences | Level 1 |
| Programming I | 4 | Basic Engineering Science / Skills | Level 1 |
| Physical Education 1 | 0 | Miscellaneous | Level 1 |
| English | 2 | Miscellaneous | Level 1 |
| Economics | 4 | HSS/M Core | Level 1 |
| **Semester 2** | **22** | | |
| Mathematics - 2 | 4 | Mathematics and Basic Sciences | Level 1 |
| Digital Design | 4 | Basic Engineering Science / Skills | 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 |
| History of Ideas | 4 | HSS/M Core | Level 1 |
| Physical Education 2 | 0 | Miscellaneous | Level 1 |
| **Semester 3** | **20** | | |
| Mathematics - 3 | 4 | Mathematics and Basic Sciences | Level 1 |
| Computer Architecture | 3 | Engineering Core | Level 1 |
| Computer Architecture Lab | 1 | Engineering Core | Level 1 |
| Programming II | 4 | Basic Engineering Science / Skills | Level 1 |
| Physics - 1 | 3 | Mathematics and Basic Sciences | Level 1 |
| Physics Lab - 1 | 1 | Mathematics and Basic Sciences | Level 1 |
| Basic Electronics | 2 | ECE Core | Level 1 |
| Electronics Lab | 2 | ECE Core | Level 1 |
| **Semester 4** | **18** | | |
| Mathematics - 4 | 4 | Mathematics and Basic Sciences | Level 1 |
| Signals and Systems | 4 | Basic Engineering Science / Skills | Level 1 |
| Physics - 2 | 3 | Mathematics and Basic Sciences | Level 1 |
| Physics Lab - 2 | 1 | Mathematics and Basic Sciences | Level 1 |
| Electronic Devices & Circuit Theory | 3 | ECE Core | Level 1 |
| Electronic Devices & Circuit Theory Lab | 1 | ECE Core | Level 1 |
| Technical Communication | 2 | Miscellaneous | Level 1 |
| **Semester 5** | **19** | | |
| Principles of Communication Systems | 3 | ECE Core | Level 1 |
| Principles of Communication Systems Lab | 1 | ECE Core | Level 1 |
| Analog CMOS Design | 3 | ECE Core | Level 1 |
| Analog CMOS Design Lab | 1 | ECE Core | Level 1 |

| Course Name | Credits | Course Category | Level |
|---|---|---|---|
| Microprocessors and Microcontrollers | 3 | ECE Core | Level 1 |
| Microprocessors and Microcontrollers Lab | 1 | ECE Core | Level 1 |
| Operating Systems | 3 | Engineering Core | Level 1 |
| Operating Systems Lab | 1 | Engineering Core | Level 1 |
| Signal Processing | 3 | ECE Core | Level 1 |
| **Semester 6** | **23** | | |
| Control Theory | 3 | ECE Core | Level 1 |
| Digital Communication | 3 | ECE Core | Level 1 |
| Digital Communication Lab | 1 | ECE Core | Level 2 |
| Elective 1 | 4 | Elective | Level 2 |
| Elective 2 | 4 | Elective | Level 2 |
| Elective 3 | 4 | Elective | Level 2 |
| HSS/M Elective - 1 | 4 | HSS/M Elective | Level 2 |
| **Semester 7** | **19** | | |
| HSS/M Elective - 2 | 4 | HSS/M Elective | Level 2 / Level 3 |
| EMT/Antenna Theory | 4 | ECE Core | Level 1 |
| Elective - 4 | 4 | Elective | Level 2 / Level 3 |
| Elective - 5 | 4 | Elective | Level 2 / Level 3 |
| Elective - 6 | 4 | Elective | Level 2 / Level 3 |
| **Semester 8** | **20** | | |
| Elective - 7 | 4 | Elective | Level 2 / Level 3 |
| 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 |
| **Semester 9** | **20** | | |
| M.Tech. Project / Thesis | 20 | Masters Project | Masters Project |
| **Semester 10** | **20** | | |
| M.Tech. Project / Thesis | 20 | Masters Project | Masters Project |

**Table 12: Course Sequencing for iMTech (ECE)**

# 9 References

1. Report on the Integrated M.Tech curriculum revision approved by the senate in April, 2015.

# APPENDIX A: Syllabus of new/updated ECE Core Courses

*Note: For courses whose syllabi is not included in this section, the existing syllabus for that course will be applicable*

# Integrated M.Tech. Course Proposal Template

| | |
|---|---|
| **Course Name** | **Analog CMOS VLSI Design** |
| **Course Branch** | Select one from the following: |

| | |
|---|---|
| | ECE |
| | |

| | |
|---|---|
| **Course Proposer Name(s)** | Suhajit Sen |
| **Course Instructor Name(s)** | Subajit Sen |
| **Course Type** *(Select one)* | Select one from the following: |

| | |
|---|---|
| | Core (ECE) |
| | |
| | |

** 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: |

| | |
|---|---|
| | |
| | |
| | N/A |

| | |
|---|---|
| **Course Category** *(Select one)* | Select one from the following: |

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

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

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

| | |
|---|---|
| **Grading Scheme** | Select one from the following: |

| | |
|---|---|
| | 4-point scale<br>(A,A-,B+,B,B-,C+,C,D,F) |
| | |

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

Basic Electronics

**Course Description**

*A brief description of the course*

1. To introduce how CMOS VLSI chips are fabricated (VLSI Technology)
2. To explain how robust Analog MOS circuits can be designed with a good understanding of VLSI Technology

**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.*

The course will discuss how Analog circuits are designed in a VLSI chip environment starting from an understanding of VLSI technology and fabrication. The methodology adopted for teaching this course is to first provide a simple physical model of the MOSFET transistor that is capable of abstracting the essential electrical behavior of the device. Following this a related small-signal MOSFET model can be derived. The application of DC and small-signal analysis methods on MOSFET circuits can then follow.

The main aim of the course will be to learn how to analyze and build CMOS amplifiers that are the building blocks of almost all VLSI mixed-signal systems. At every stage of the course the students are expected to design, on paper as well as simulation, the circuits discussed in the class. An important aspect of the course will be a project in which the students are expected to design and simulate (using Spice simulator).

**Course Content**

**Topics** : VLSI Technology, MOS device physics, Common-source, common-gate, common-drain, and cascode stages, Differential amplifiers, Current mirrors, Frequency response of amplifiers, One and two-stage operational amplifiers, Stability and frequency compensation, feedback networks, Memory design. The course will be useful for those interested in VLSI Design, mixed-signal embedded hardware and is a pre-requisite for RF Design.

**Assessments / Grading**

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

**Text Book / References**

1. 1 CMOS : Circuit Design, Layout and Simulation, R. Jacob Baker, IEEE Press/Wiley Student Edition.
2. Silicon VLSI Technology Fundamentals, Practice and Modeling, J. D. Plummer, M. D. Deal, and P. B. Griffin

# Integrated M.Tech. Course Proposal Template

<table>
<tr><td><b>Course Name</b></td><td>Basic Electronics</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><th>Hours</th><th>Component</th></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>
<tr><td colspan="2"></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

# Integrated M.Tech. Course Proposal Template

<table>
<tr><td><b>Course Name</b></td><td>Electronics 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>
</table>

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

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

# Integrated M.Tech. Course Proposal Template

<table>
<tr><td><b>Course Name</b></td><td>Control Theory</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></td><td>Practical (2hrs = 1 credit)</td></tr>
<tr><td><b>3</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>
</table>

**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.**

# Integrated M.Tech. Course Proposal Template

<table>
<tr><td><b>Course Name</b></td><td>Engineering Electromagnetics</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></td><td>Practical (2hrs = 1 credit)</td></tr>
<tr><td><b>3</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"></td></tr>
<tr><td colspan="2"><b>Course Description</b></td></tr>
</table>

*A brief description of the course*

This course teaches the physics and applications of electromagnetic field theory as encapsulated in the vector form of Maxwell's equations. The class will show how these laws govern the design and bound the performance of electronic devices, circuits, and 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:

1. understand the coupling between electric and magnetic fields through Maxwell’s equations,
2. understand constitutive parameters and boundary conditions and be able to analyze the relationships between fields and flux densities in material media,
3. be able to analyze electromagnetic waves in material media, and the reflection and transmission of these waves between different media,
4. be able to interpret the energy and power associated with electromagnetic fields,
5. be able to analyze and design basic transmission lines and waveguides,
6. be able to analyze and assess antennas and radiation from antennas.

**Course Content**

1. Introduction (1 class)
2. Vector Analysis (4 classes)
3. Electrostatics (7 classes)
4. Magnetostatics (4 classes)
5. Maxwell’s Equations (3 classes)
6. Plane Wave Propagation (5 classes)
7. Reflection, Transmission and Waveguides (8 classes)
8. Radiation and Antennas (6 classes)
9. Transmission Lines (8 classes)

**Assessments / Grading**

Midterm, Final, 2 Quizzes, Homework Assignments
Laboratory assignments

**Text Book / References**

Field and Wave Electromagnetics, 2nd Edition,
By David K. Cheng, Addison Wesley Publishing Co., 1992.

Engineering Electromagnetics, William H Hayt Jr, McGraw Hill Publishers

# Integrated M.Tech. Course Proposal Template

<table>
<tr><td><b>Course Name</b></td><td>Electronic devices and circuit theory</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><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></td></tr>
</table>

*A brief description of the course*

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)

# Integrated M.Tech. Course Proposal Template

**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) |
| 2 | Practical (2hrs = 1 credit) |
| **4** | **Total Credits** |

**Grading Scheme**

Select one from the following:

| | |
|---|---|
| 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)*

Signals and Systems

**Course Description**
*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

# Integrated M.Tech. Course Proposal Template

<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><td><b>Hours</b></td><td><b>Component</b></td></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 (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></td></tr>
</table>

*A brief description of the course*

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.

# APPENDIX B: Updates to Mathematics and Basic Science Course Syllabus

*Note: For courses whose syllabi is not included in this section, the existing syllabus for that course will be applicable*

## PROPOSED COURSE DESCRIPTIONS OF THE MATHEMATICS COURSES

### Mathematics I: Proofs and Basic Calculus

| Topic | Hours |
|---|---|
| Sets, Cardinality of a set, Numbers, Principles of Inference:<br>Negation, Disjunction, Implication, Equivalence, Truth tables and tautologies | 4 |
| Proofs and Mathematical Writing<br>- These topics are introduced to teach the students the clarity of thought. | 3 |
| Real numbers, Sequences, Series, Limit, Continuity, Differentiability<br>Mean value theorems and applications, Linear Approximation | 10 |
| Power series, Taylors theorem (one variable), Approximation by polynomials, Critical points, Convexity | 6 |
| Riemann integral, fundamental theorems of integral calculus, Improper integrals.<br>Curve tracing, Graphs of polar equations | 7 |
| Space co-ordinates, lines and planes, Polar coordinates, Cylinders, Quadric surfaces, volume, area, length, Continuity, Differentiability of vector functions, arc length Functions of two or more variables, partial derivatives, Statement of Taylors theorem criteria for maxima/minima/saddle points. | 9 |
| Double, triple integrals, Jacobians, | 3 |
| Total | 42 |

Recommended books:

- Calculus and Analytic Geometry, by Thomas and Finney, 9th edition, Pearson Education India.
- Introduction to Logic, by Patrick Suppes, Dover edition, 1999. (For elementary logic portion).
- Calculus, Vol. 1: One-Variable Calculus, with an Introduction to Linear Algebra, by Tom Apostol
- Calculus, Vol. 2: Multi-Variable Calculus and Linear Algebra with Applications to Differential Equations and Probability, by Tom Apostol
- Proper web notes

Additional references:

- Mathematical Analysis, by Tom M. Apostol, Addison-Wesley, 1974.
- Theory of Computation, by Harry Lewis and Christos H. Papadimitriou, Prentice-Hall, 2nd edition, 1997. (For logic).
- Real and Complex Analysis, by Walter Rudin, McGraw - Hill, 2006.

## Mathematics II: Linear Algebra and Introduction to Basic ODE

We are of the opinion that Linear Algebra is one of the most basic and fundamental courses in Mathematics - it provides the necessary language and forms an essential toolkit for all branches of Science and Technology as well as Economics etc. while teaching the Ordinary Differential Equations course we realized that we needed it and thus we have moved the Linea Algebra course to the second semester along with some ODE and PDE.

| Topic | Hours |
|---|---|
| Matrices, Matrix Operations (Addition, Scalar Multiplication, Multiplication, Transpose, Adjoint) and their properties; Special types of matrices (null, Identity, Diagonal, Triangular, Symmetric, Skew-Symmetric, Hermitian, Skew-Hermitin, Orthogonal, Unitary, Normal), Solution of the matrix equation Ax=b, Row-reduced Echelon form, Determinants and their properties (without proof). | 8 |
| Vector Space, Subspaces, Linear Dependence/Independence, Basis, Standard Basis of dimension, Coordinates with respect to a basis, Complementary Subspaces, Standard inner product, Norm, Gram-Schmidt Orthogonalisation Process, Generalisation to the vector space Linear Transformation form (motivation X-AX), Image of a basis identifies the linear transformation, Range Space and Rank, Null Space and Nullity, Matrix Representation of a linear transformation, Structure of the solutions of the matrix equation Ax=b, Linear Operators and their representation by matrices, Similar Matrices and linear operators, Invertible linear operators, Inverse of a non-singular matrix, Cramers method to solve the matrix equation Ax=b, Eigenvalues and eigenvectors of a linear operator, Characteristic Equation, Bounds on eigenvalues, Diagonalisability of a linear operator, Canonical forms, Statement of the Kelly-Hamilton Theorem | 12 |
| Standard Inner product, Norm, Gram-Schmidt Orthogonalisation Process, Self-Adjoint, Normal and Unit array operators, Properties of eigenvalues and eigenvectors, Spectral theorem Self-Adjoint and Normal, Quadratic form XT AX, Positive and Semi-Positive Definite Matrices. | 10 |
| Introduction and Motivation to Differential Equations, First Order ODE, Geometrical interpretation of solution, Equations reducible to separable form, Exact Equations, integrating factor, Linear Equations, Orthogonal trajectories, Picards Theorem for IVP (without proof) and Picards iteration method, Eulers Method, Improved Eulers Method. | 7 |
| Introduction to PDE, basic concepts, Maxwells equation and heat equation This is to cover the essential pre-requisites for a third semester Physics course. | 3 |
| Total | 40 |

Recommended books:

- Linear Algebra, by K. Hoffman and R. Kunz, Prentice-Hall, 1971.
- Relevant portions from the books written by Artin, Gallian, Herstein.
- Proper web notes.

Additional references:

- Linear Algebra and its Applications, by Gilbert Strang, Nelson Engineering, 2007.
- Finite Dimensional Vector Spaces, by P. R. Halmos, Princeton University Press.
- Linear Algebra, by Helson, Holden-day, 1990.
- Lectures on Abstract Algebra, volumes by N. Jacobson, Springer.

## Applied and Applicable Mathematics III: ODE and Abstract Algebra

| Topic | Hours |
|---|---|
| Surfaces, integrals, Vector Calculus,Green, Gauss, Stokes Theorems. | 5 |
| Second Order Linear differential equations, fundamental system of solutions and general solution of homogeneous equation, use of known solution to find another, Existence and uniqueness of solution of IVP, Wronskian and general solution of nonhomogeneous equations, Euler-Cauchy Equation, extensions of the results to higher order linear equations. | 12 |
| Power Series Method - application to Legendre Equations, Legendre Polynomials, Frobenius Method, Bessel equations, Properties of Bessel functions, Sturm comparison Theorem, Sturm Liouvile BVP, Orthogonal functions, Fourier Series and Integrals. | 9 |
| Basic Introduction to Laplace and Fourier Transforms (with less stress on theoretical aspects) | 6 |
| Introduction to Abstract Algebra: Groups, Rings, Modules, Ideals, Fields and examples of finite fields | 10 |
| Total | 42 |

Recommended books:

- Calculus and Analytic Geometry, by Thomas and Finney, 9th edition, Pearson Education India.
- Advanced Engineering Mathematics, by Erwin Kreyszig, 8th edition, Wiley,
- Partial Differential Equations, by Fritz John, 4th edition, 1981.
- Proper web notes (NPTEL notes are available)

Additional references:

- Differential Equations with Applications and Historical Notes, by George F. Simmons, McGraw-Hill Science/Engineering/Mathematics, 2nd edition, 1991
- Introduction to Ordinary Differential Equations, by Shepley L. Ross, 4th edition, Wiley, 1989.
- Elements of Partial Differential Equations, by Ian Sneddon.
- An Elementary Course in Partial Differential Equations, by Amaranath, Alpha Science Intl. Ltd., 1997.
- Advanced Theory of Statistics, by Kendall and Stuart, all volumes.

**Mathematics IV: (2+2 credits)**

| Topic | Hours |
|---|---|
| Mathematics IVB: Statistics and Probability<br>(Adopted from : Chap. 22 and 23 Kreyszig- Part G): | |
| Data representation, average, probability, permutations and combinations, random variables, probabilistic distributions, mean and variance, binomial, Poisson, hypergeometric, Normal distributions, distributions of several random variables. | 14 |
| Mathematical Statistics: random sampling, confidence intervals, testing 7-8 of hypothyses, decisions, goodnes of fit, $\chi^2$ test test, linear regression. | 2 |
| Total | 21-22 |
| Mathematics IVA: Complex Analysis<br>Complex Numbers, geometric representation, powers and roots of complex numbers, Functions of a complex variable, Analytic functions, Cauchy-Riemann equations; elementary functions, | 8 |
| Contours and contour integration, Cauchys theorem, Cauchy integral formula, Power Series, term by term differentiation, Taylor series, Laurent series, Zeros, Singularities, poles, essential singularities, Residue theorem, Evaluation of real integrals and improper integrals. | 12 |
| Total | 20 |

Recommended books:

- **Mathematics IVA**:
  Advanced Engineering Mathematics, by Erwin Kreyszig, 8th edition, Wiley,
  Proper web notes (NPTEL notes are available)
- **Mathematics IVB**:
  Advanced Engineering Mathematics, by Erwin Kreyszig, 8th edition, Wiley,
  Proper web notes

Additional references:

- **Mathematics IVA**:
  Complex Analysis, by Ahlfors, McGraw Hill, 1979.
  Complex Variables and Applications, by James Brown and Ruel Churchill, McGraw Hill, 2008.
- **Mathematics IVB**:
  Introduction to Mathematical Statistics, by Hogg and Craig, 3rd edition, Macmillan, 1971.
  The Advanced Theory of Statistics, by Kendell and Stuart, Volume 3, Griffin, 1976.

# Physics Theory I (BS107)
## Classical Physics
## (3rd semester)
## Curriculum Details

Mechanics / Classical Mechanics:

Introduction; kinematics, Newton's laws.
work-energy theorem; revision of integral theorems -- Gauss's divergence theorem, Green's theorem in the plane, Stokes ' theorem; curvilinear coordinates: vectors in curvilinear systems, arc length & volume element; gradient, divergence & curl in curvilinear coordinates; specific example of spherical & cylindrical coordinate systems; transformation between coordinate systems, Jacobian; **5 hours**
Manipulation of gradient, divergence, curl, laplacian operators on vectors simplified by tensors; Euler-Lagrange equation; principle of least action; generalized coordinates & generalized momenta; writing the Lagrangian of a system (examples: e.g., simple pendulum, double pendulum, etc.), finding the equation of motion; cyclic /ignorable coordinates & constants of motion, Jacobi integral; rotating frames of reference; mention of symmetries & Noether's theorem; energy & momentum conservation as consequences of homogeneity of time & space respectively, angular velocity, angular momentum conservation & isotropy of space; pseudo-forces, Coriolis & centrifugal forces, effects of Coriolis force, Foucault's pendulum, precession; rigid-body motion, moment of inertia; perpendicular & parallel axes theorems; **7 hours**

Central force motion, 2-body central force problem, reduction to 1-body problem; angular momentum conservation; solution of motion of a particle in a central field; the Kepler problem; virial theorem; **3 hours**

Elasticity, stress strain curve, yield point, breakdown stress, etc. **1 hour**

Poincare & the 2+ body problem; Phase space & chaos; examples of specific systems **2 hours**

Simple harmonic motion; undamped, damped regimes, etc. **2 hours**
Wave motion: free vibrations of a stretched string, phase velocity, group velocity, sound waves, water waves, interference & diffraction, etc. **3 hours**

Electrostatics: flux of an electric field, Gauss's law, applications, electric potential energy, the divergence of E, Dirac delta function, conductors, capacitance & combinations of capacitors, energy density, dielectrics, dipole, dipole moment, polarization, electric field calculations of various charge configurations, etc. **6 hours**

Magnetostatics: Lorentz force, cyclotron frequency, magnetic force & current-carrying wires, continuity equation, Biot-Savart law & applications, Ampere's law, magnetic dipole moment, magnetic materials, diamagnets, paramagnets & ferromagnets, magnetization & magnetic susceptibility, hysteresis, Faraday's law of electromagnetic induction, Lenz's law, electromagnetic waves, energy density, Maxwell's equations, potential formulation of electrodynamics -- gauge transformations, wave equation, polarization. **7 hours**

Thermodynamics: concept of equation of state, reversible & irreversible transformations, equilibrium states, cyclical transformations, internal energy, first law of thermodynamics, specific heat & thermal capacity, adiabatic & isothermal transformations of a gas, second law of thermodynamics, Carnot cycle, Clausius's & Kelvin's postulates, entropy, thermodynamic state function, heat conduction, convection & radiation, Stefan's law, etc. **6 hours**

**Total: 42 hours**

**Recommended / suggested books & references:**

1. D. Kleppner & R. Kolenkow, *An introduction to mechanics*, Tata McGraw Hill (2007)
2. H. C. Verma, *Concepts in Physics*, Vols. I & II, Bharati Bhawan (2011).
3. D. J. Griffiths, *Introduction to electrodynamics*, Prentice Hall of India
4. N. C. Rana & P. S. Joag, *Classical mechanics*
5. L. Landau & E. Lifshitz, *Mechanics*
6. Zemansky & Dittmann, *Thermodynamics*
7. *The Feynman Lectures in Physics*, Narosa (2008).

## Physics Theory II (BS108)
## Modern Physics
## (4th semester)
## Curriculum Details

Special theory of Relativity:
inertial frames of reference; galilean transformations, Lorentz transformations; relativistic kinematics: Lorentz-Fitzgerald length contraction, time dilation, velocity transformation; Doppler effect -- inon-relativistic and relativistic; relativistic dynamics -- effect on momentum & mass measurements. **4 hours**

Quantum Mechanics:
Need for QM, the photoelectric effect, wave-particle duality, the Compton effect; de Broglie waves, phase & group velocities; wave-function & probability; a brief discussion on interference & diffraction, particle in a box (without solving Schroedinger's eqn); Heisenberg's uncertainty principle; Thomson & Rutherford's models of the atom; atomic spectra; Bohr's model of the atom & its explanation of spectral lines; Bohr- Sommerfeld quantization; wave-function, probability density, Schroedinger's equation (steady-state & time-dependent) solution for simple problems: particle in a box, tunneling through a potential barrier, simple harmonic oscillator, hydrogen atom, etc; fundamental postulates of wave mechanics, expectation values, operators, commutator relations. **16 hours**

Nuclear physics: models for the atomic nucleus, liquid drop model; nuclear reactions & radioactive decay. **3 hours**

Statistical mechanics:
Phase space, macrostates & microstates, entropy; distinguishable & indistinguishable particles; the most probable distribution; Maxwell-Boltzmann distribution; quantum statistical mechanics, Fermi-Dirac & Bose-Einstein distributions, Fermi energy; Planck's radiation formula. **10 hours**

Specific heat of solids, Dulong-Petit law, Einstein model; free electron theories, metals; solids & crystals; origin of band structure, electrical & thermal properties of solids; **5 hours**
semiconductors;
lasers. **2 hours**

**Total: 40 hours**

**Recommended / suggested books:**

1. Arthur Beiser, *Concepts of Modern Physics*, Tata McGraw Hill.
2. Mathews & Venkatesan, *A textbook of quantum mechanics*
3. D. Kleppner & R. Kolenkow, *An introduction to Mechanics*, Tata McGraw Hill (2007)
4. F. Reif, *Statistical Physics* (Berkeley physics course vol.5), McGraw Hill (1967)
5. C. Kittel, *Solid State Physics* (any of the several editions).
6. Mani & Mehta, *Introduction to modern physics*, Affiliated East-West Press.
7. *The Feynman Lectures in Physics*, Narosa (2008).

# 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 C: Updates to CSE / Engineering Core Course Syllabus

*Note: For courses whose syllabi is not included in this section, the existing syllabus for that course will be applicable*

# Integrated M.Tech. Course Proposal Template

<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)</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>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:

7. 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.
8. Algorithms for various classical problems in Computer Science ranging from sorting, graph algorithms, algorithms manipulating numbers and strings etc.
9. Algorithm design techniques and strategies like divide-and-conquer, dynamic programming and greedy algorithms along with several problems using these techniques, proofs of correctness.
10. 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.

# Integrated M.Tech. Course Proposal Template

<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)<br>(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:
<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">None</td></tr>
</table>

**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:

11. Know the protocol stack for the Internet
12. Understand the need for protocols
13. Know the detailed operations of the TCP/IP protocols
14. 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

# Integrated M.Tech. Course Proposal Template

<table>
<tr><td><b>Course Name</b></td><td>Programming-1 and Lab (C Part)</td></tr>
<tr><td><b>Course Branch</b></td><td>Select one from the following:
<table>
<tr><td></td><td>CSE and ECE</td></tr>
<tr><td></td><td></td></tr>
</table></td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td>Dr. Madhav Rao</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></td><td><b>Core</b></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></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></td><td><b>Lecture (1hr = 1 credit)</b></td></tr>
<tr><td></td><td>Tutorial (1hr = 1 credit)</td></tr>
<tr><td></td><td><b>Practical (2hrs = 1 credit)</b></td></tr>
<tr><td></td><td><b>Total Credits = 2</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>Satisfactory/Unsatisfactory (S / X)</td></tr>
</table>
4-point scale</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>
</table>

**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)

# Integrated M.Tech. Course Proposal Template

**Course Name**

Programming I – Python

**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 |
| | 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** |
|---|---|
| 1 | Lecture (1hr = 1 credit) |
| | Tutorial (1hr = 1 credit) |
| 1 | Practical (2hrs = 1 credit) |
| **2** | **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)*

**Course Description**

*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**

Problem solving: Decomposition, abstraction, composition

Problem solving: Decomposition, abstraction, composition

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**

Programming I – Python

**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 |
| | 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 |
|---|---|
| 1 | Lecture (1hr = 1 credit) |
| | Tutorial (1hr = 1 credit) |
| 1 | Practical (2hrs = 1 credit) |
| **2** | **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)*

**Course Description**

*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**

Problem solving: Decomposition, abstraction, composition

Problem solving: Decomposition, abstraction, composition

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<br>Online resources |

# Integrated M.Tech. Course Template

<table>
<tr><td><b>Course Name</b></td><td>Data Structures and 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></td></tr></table></td></tr>
<tr><td><b>Course Proposer Name(s)</b></td><td>Muralidhara V N</td></tr>
<tr><td></td><td>Muralidhara V N</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>1</td><td>Tutorial (1hr = 1 credit)</td></tr><tr><td>1</td><td>Practical (2hrs = 1 credit)</td></tr><tr><td><b>4+1</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>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">Programming I</td></tr>
</table>

**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:

15. to know what are efficient algorithms.
16. Compute the time and space complexity of algorithms.
17. Know the difference between worst/best/average  and amortized cost.
18. know about arrays, linked lists, stacks, queue and hashing techniques, sorting and binary search.
19. Understand various types of tree structures and graph as a data structure.
20. 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**

# Integrated M.Tech. Course Proposal Template

<table>
<tr><td><b>Course Name</b></td><td>Discrete Mathematics</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>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>✓</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>✓</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></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>
(L: T) = (3:1)</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>Satisfactory/Unsatisfactory (S / X)</td></tr>
</table>
4-point scale</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>
</table>

**Course Description**
*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

## 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.

# Integrated M.Tech. Course Proposal Template

<table>
<tr><td><b>Course Name</b></td><td>Programming Languages</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></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>X</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>CSE</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>40</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:
<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">Data-structures and algorithms</td></tr>
</table>

**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