

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY, TIRUCHIRAPALLI
SEMESTER-4 GUIDELINES BY PRAJWAL SUNDAR**

A. Courses

Programme Core Courses

CSPC41	Formal Languages and Automata Theory	4 credits
CSPC42	Design and Analysis of Algorithms	3 credits
CSPC43	Operating Systems	3 credits

Laboratory Courses

CSLR41	Algorithms Laboratory	2 credits
CSLR42	Operating Systems Laboratory	2 credits

Humanities Department Course

HSIR13	Industrial Economics and Foreign Trade	3 credits
--------	--	-----------

Programme Electives (Any 2 of the following)

CSPE41	Software Engineering	3 credits
CSPE42	Design Thinking	3 credits
CSPE43	Advanced Data Structures and Algorithms	3 credits

Total Credits = (4+3+3) + (2+2) + (3) + (3+3) = 10 + 4 + 3 + 6 = **23 credits**

B. Guide for choosing and studying electives

- Summary: The best choice would be to choose DT and any one of SE or ADSA. If you liked DSA and are keen on exploring DSA further, take ADSA. If you aren't that sort of a person, and find theory more interesting, take SE.
- Software Engineering: It is a completely theoretical subject, handled by Dr. Oswald (combined class for both sections). You will have a lot of slides to mug up theory from. To give you a better picture, for example, you will have to mug up a total of 879 pages. Scoring grades in SE is relatively easier if you are more of a theory person and can mug up slides in an efficient manner. The question papers are also highly detailed, so only if you study all slides efficiently, will you be able to score good grades.
[Complete Study PDF attached]
- Design Thinking: It is highly recommended that all students take this elective. This elective is the easiest elective out of all electives that you will ever be taking in your B. Tech Life. In every batch, a minimum of 95 to 96% of the population opts for this elective, and almost every time, the elective is split between sections – you have a separate professor for each section.
Study Guide: This subject must be studied in groups. Use the power-points given by your respective professor [mostly research scholars only handle this elective]. The key to scoring an S or A grade is that, in the examination, *write as*

much as possible. Even for a 1- or 2-mark question, make sure to write at least a page of answer. This is possible only when you have thoroughly studied the power-point to an extent that you can make up your own stories in the examination.

- Advanced Data Structures and Algorithms: This elective is handled by Dr. Nithya (combined class for both sections). This course is an extension of the Data Structures and Algorithms (CSC32) course which was offered in the third semester. *This is a completely conceptual course*. There are some concepts and methods which must be learnt and applied exactly the same way – just the same way, for example, you learnt about a binary search tree and learnt how to perform insertion and deletion on it – the CTs will be of the exact same format as questions are covered in the class.

Link to Nithya Mam's Online Lectures: [Advanced Data Structure and Algorithm - YouTube](#)

My Advice: Watch these lectures beforehand and make your own notes. In the classroom, make effective use of the time to revise and thorough the concept. Before the day of the examination, just go through the notes and try solving one question in each model from the notes itself – there is no need to refer any other source. This will easily guarantee an S or an A.

C. Guide to study programme core courses

- Formal Languages and Automata Theory: This is a very interesting conceptual subject with 99% concepts and 1% theory. It holds importance in the GATE examination, so make sure to understand this subject to the fullest.

YouTube Neso Link: [Theory of Computation & Automata Theory - YouTube](#)
(a bit basic tutorial)

YouTube Gate Smashers Link: [TOC\(Theory of Computation\) - YouTube](#)
(a good level tutorial)

YouTube Gate-Hub Link: [Theory of Computation - YouTube](#)
(an advanced level tutorial)

Study Guide: This subject is one which becomes easier with time and more exposure. I personally increased my grasp on this subject by watching more playlists and solving more questions from a textbook. You can do the same – first watch Neso, then Gate Smashers, then Gate-Hub. Though you will be watching the same things multiple times, more repeated exposure will help engrain the concept better and help you gain mastery over it.

If you can crack GATE questions from this subject, you are good to go.

- Design and Analysis of Algorithms: This is the most important course of this semester. Please take this course seriously as it is very important for IPMP and CPC. This introduces new methodologies of programming and helps to increase the coding skill of a coder by 100 times.

For Units 1,2,3 – Refer Abdul Bari's playlist - [Algorithms - YouTube](#)
[my notes are attached as PDFs – refer them while watching videos]

For Unit 4 – Refer any YouTube video of your choice – it is a very complicated area of study – to an extent that there is a separate honours course dedicated to study it offered by our department – Randomized Algorithms. But do not worry, only a very few questions [1 or maximum 2] will be asked from this chapter in the examinations.

For Unit 5 – Refer Nithya mam's video [Both algos unit 5 and ADSA unit 5 are the same] - [ADSA Lec #30 - NP Complete - 17/04/2021 \(Last ADSA Class\) - YouTube](#)

- *Operating Systems*: This is a very important course with respect to internship and placement interviews [also in online tests to an extent]. Understand the concepts but remember that you have to necessarily mug up slides as a lot of theory is involved which needs mugging up – you can *understand and mug up*, but you cannot escape from the fact that a lot of theory is involved, and you need to sharpen your memory for this course.

Neso Academy: [Operating System - YouTube](#)

(basic to medium level tutorial)

Gate Smashers: [Operating System \(Complete Playlist\) - YouTube](#)

(good level tutorial)

Ensure you watch all videos of both playlists first. Then open your power-points provided by your professor and make your own short notes for theory.

There are some numerical based concepts that must be studied and applied.

Gate Smashers explains the concepts well, so you can refer that.

[The HOD, Dr. Mary Saira Bhanu, handles this course for one of the sections. She handled my class in the 4th semester, so I have attached her slides also as one big PDF]

D. Guide for laboratory courses

- *Algorithms Laboratory*: Some professors disclose questions beforehand, so if that is the case, please prepare and then go to the lab. Others give on the spot questions – if this is the case, ensure you have studied the concept thoroughly and done at least 15 to 20 leetcode questions on that concept. At any case, do not waste precious time in the laboratory. Make full use of the time provided. Make sure to follow a strict discipline of completing the lab record on the same day as the lab is completed.
Languages: Purely C and C++ will be used. Ensure you master these languages from standard tutorials – they will be very important for internships and placements also.
[my codes are attached as a folder]
- *Operating Systems Laboratory*: Three programming languages need to be used for this course – shell programming, C programming and C++ programming [there are some codes which cannot be done in C++ and are only available in C, so make sure that you do not lose touch with C].

- *Shell Programming*: Refer YouTube videos and my codes. They aren't that hard to learn, just that you need to spend time to learn it with dedication.
- *C Programming*: Many operating system concepts like processes and threads are only available in C – so learn these concepts and code well. *[tip: also learn to code these in python – it will be interesting and easier – this will also be asked in some company interviews, and it is easier to code these concepts in python than in C]. Some mugging up is required here, so do it gracefully without complaining.*
- *C++ Programming*: Concepts like process scheduling, disk scheduling involves algorithms that can be coded easily in C++. You can refer my code and understand how each algorithm works. This is the easiest part of the operating systems laboratory.

[my codes are attached as a folder]

E. Guide for economics

This course is offered by the humanities department. It is completely out of touch with the other computer science courses. It is offered to give a flavour of students pursuing commerce-based courses – and some of the concepts are also important and may help us sometime in the future.

Study Guide: Just follow the power-points provided by your teacher, understand and mug up.

Interest is essential while studying this course. Without interest, you may end up getting a very bad grade, so ensure that you put your heart and soul into this course.

[my teacher's PDF is attached]

Dear Juniors,

With this, I would like to conclude the document. I wish you all the best for your 4th semester [and the upcoming semesters]. Prepare well, score good grades, pass IPMP with flying colours and ensure you do not mess up your grades at any cost. For any issues, feel free to contact me anytime [my number: +917010460164]. I prefer introduction via text and if needed, you can also call anytime. Once again, all the best!
