THE ASSAM
ROYAL GLOBAL UNIVERSITY
GUWAHATI

Royal School of Information Technology (RSIT)

Course Structure & Syllabus

(Based on National Education Policy 2020)

For

Bachelor of Computer Application (BCA)

W.E.F

AY: 2023-2024

TABLE OF CONTENTS

Preamble

The National Education Policy (NEP) 2020 conceives a new vision for India's higher education system. It recognizes that higher education plays an extremely important role in promoting equity, human as well as societal well-being and in developing India as envisioned in its Constitution. It is desired that higher education will significantly contribute towards sustainable livelihoods and economic development of the nation as India moves towards becoming a knowledge economy and society.

Towards the attainment of holistic and multidisciplinary education, the flexible curricula of the University will include credit-based courses, projects in the areas of community engagement and service, environmental education, and value-based education. As part of holistic education, students will also be provided with opportunities for internships with local industries, businesses, artists, crafts persons, and so on, as well as research internships with faculty and researchers at the University, so that students may actively engage with the practical aspects of their learning and thereby improve their employability.

At The Assam RGU, we are committed that at the societal level, higher education will enable each student to develop themselves to be an enlightened, socially conscious, knowledgeable, and skilled citizen who can find and implement robust solutions to its own problems. For the students at the University, Higher education is expected to form the basis for knowledge creation and innovation thereby contributing to a more vibrant, socially engaged, cooperative community leading towards a happier, cohesive, cultured, productive, innovative, progressive, and prosperous nation."

The Bachelor of Computer Applications (BCA) program, designed in alignment with the National Education Policy (NEP) 2020, aims to provide a comprehensive curriculum that emphasizes the development of both theoretical knowledge and practical skills in the field of computer applications. The program is structured to foster critical thinking, problem-solving capabilities, and an entrepreneurial mindset among students. Embracing the holistic and multidisciplinary approach of NEP 2020, the BCA curriculum integrates core computer science courses with interdisciplinary studies and soft skills training, ensuring students are well-prepared to adapt and innovate in the rapidly evolving digital landscape. Additionally, the program offers flexibility in course choices and durations, encouraging students to pursue internships and projects that align with their career goals and interests, ultimately molding them into skilled professionals ready to meet the challenges of the global IT industry.

Section 1:

Overview

1. 1. Introduction:

1.1.1 About NEP 2020

The National Education Policy (NEP) 2020 clearly indicates that higher education plays an extremely important role in promoting human as well as societal well-being in India. As envisioned in the 21st-century requirements, quality higher education must aim to develop good, thoughtful, well-rounded, and creative individuals. According to the new education policy, assessments of educational approaches in undergraduate education will integrate the humanities and arts with Science, Technology, Engineering and Mathematics (STEM) that will lead to positive learning outcomes. This will lead to develop creativity and innovation, critical thinking and higher-order thinking capacities, problem-solving abilities, teamwork, communication skills, more in-depth learning, and mastery of curricula across fields, increases in social and moral awareness, etc., besides general engagement and enjoyment of learning. and more in-depth learning.

The NEP highlights that the following fundamental principles that have a direct bearing on the curricula would guide the education system at large, viz.

i. Recognizing, identifying, and fostering the unique capabilities of each student to promote her/his holistic development.
ii. Flexibility, so that learners can select their learning trajectories and programmes, and thereby choose their own paths in life according to their talents and interests.
iii. Emphasis on conceptual understanding rather than rote learning, critical thinking to encourage logical decision-making and innovation; ethics and human & constitutional values, and life skills such as communication, teamwork, leadership, and resilience.
iv. Extensive use of technology in teaching and learning, removing language barriers, increasing access for Divyang students, and educational planning and management.
v. Respect for diversity and respect for the local context in all curricula, pedagogy, and policy.
vi. Equity and inclusion as the cornerstone of all educational decisions to ensure that all students can thrive in the education system and the institutional environment are responsive to differences to ensure that high-quality education is available for all.

1.1.2 About the BCA Course:

The Bachelor of Computer Applications (BCA) program, structured according to the National Education Policy (NEP) 2020, aims to equip students with a deep understanding of computer applications alongside a broad interdisciplinary educational foundation. This updated curriculum is designed to enhance analytical thinking, algorithmic skills, and software development capabilities while integrating insights from data science, artificial intelligence, and systems design. Embracing the NEP's flexible and student-centric approach, the BCA program allows learners to choose from a wide array of elective subjects, which not only expands their technical prowess but also hones their entrepreneurial and innovative thinking skills.

Furthermore, the BCA under NEP 2020 emphasizes the development of key soft skills such as communication, teamwork, and problem-solving, which are vital in today's collaborative and dynamic work environments. The inclusion of mandatory internship opportunities and project work in the curriculum ensures that students can apply their theoretical knowledge in real-world settings, bridging the gap between academic learning and professional requirements. This hands-on experience is critical in preparing graduates for the technological challenges of the future, making them valuable assets in a global and highly competitive job market.

1.1.3 Vision

To cultivate globally competent information technology professionals through integrated educational experiences and diverse cultural exposure, preparing graduates to excel in a dynamic international environment.

1.1.4 Mission

- To foster academic excellence in the field of information technology through cutting-edge, research-driven, and industry-relevant education
- To instil ethical conduct and compassion through active community engagement.
- To develop responsible leaders who are poised to drive positive change and shape the future of technology and society.

1.2. Credits in Indian Context:

1.2.1. Choice Based Credit System (CBCS)

Under the CBCS system, the requirement for awarding a degree or diploma or certificate is prescribed in terms of the number of credits to be earned by the students. This framework is being implemented in several universities across States in India. The main highlights of CBCS are as below:

- The CBCS provides flexibility in designing curriculum and assigning credits based on the course content and learning hours.
- The CBCS provides for a system wherein students can take courses of their choice, learn at their own pace, undergo additional courses and acquire more than the required credits, and adopt an interdisciplinary approach to learning.
- CBCS also provides opportunity for vertical mobility to students from a bachelor's degree programme to masters and research degree programmes.

1.3. Definitions

1.3.1. Academic Credit:

An academic credit is a unit by which a course is weighted. It is fixed by the number of hours of instructions offered per week. As per the National Credit Framework:

1 Credit = 30 NOTIONAL CREDIT HOURS (NCH)

Yearly Learning Hours = 1200 Notional Hours (@40 Credits x 30 NCH)

30 Notional Credit Hours		
Lecture/Tutorial	**Practicum**	**Experiential Learning**
1 Credit = 15 -22 Lecture Hours	10-15 Practicum Hours	0-8 Experiential Learning Hours

1 Hr. Lecture (L) per week 1 credit	1 credit
1 Hr. Tutorial (T) per week	1 credit
1 Hr. Practical (P) per week	0.5 credits
2 Hours Practical (Lab) per week	1 credit

1.3.2. Course of Study:

Course of study indicate pursuance of study in a particular discipline/programme. Discipline/Programmes shall offer Major Courses (Core), Minor Courses, Skill Enhancement Courses (SEC), Value Added Courses (VAC), Ability Enhancement Compulsory Courses (AECCs) and Interdisciplinary courses.

1.3.3. Disciplinary Major:

The major would provide the opportunity for a student to pursue in-depth study of a particular subject or discipline. Students may be allowed to change major within the broad discipline at the end of the second semester by giving her/him sufficient time to explore interdisciplinary courses during the first year. Advanced-level disciplinary/interdisciplinary courses, a course in research methodology, and a project/dissertation will be conducted in the seventh semester. The final semester will be devoted to seminar presentation, preparation, and submission of project report/dissertation. The project work/dissertation will be on a topic in the disciplinary programme of study or an interdisciplinary topic.

1.3.4. Disciplinary/interdisciplinary minors:

Students will have the option to choose courses from disciplinary/interdisciplinary minors and skill-based courses. Students who take a sufficient number of courses in a discipline or an interdisciplinary area of study other than the chosen major will qualify for a minor in that discipline or in the chosen interdisciplinary area of study. A student may declare the choice of the minor at the end of the second semester, after exploring various courses.

1.3.5. Courses from Other Disciplines (Interdisciplinary):

All UG students are required to undergo 3 introductory-level courses relating to any of the broad disciplines given below. These courses are intended to broaden the intellectual experience and form part of liberal arts and science education. Students are not allowed to choose or repeat courses already undergone at the higher secondary level (12^{th} class) in the proposed major and minor stream under this category.

i. Natural and Physical Sciences: Students can choose basic courses from disciplines such as Natural Science, for example, Biology, Botany, Zoology, Biotechnology, Biochemistry, Chemistry, Physics, Biophysics, Astronomy and Astrophysics, Earth and Environmental Sciences, etc.

ii. Mathematics, Statistics, and Computer Applications: Courses under this category will facilitate the students to use and apply tools and techniques in their major and minor disciplines. The course may include training in programming software like Python among others and applications software like STATA, SPSS,

Tally, etc. Basic courses under this category will be helpful for science and social science in data analysis and the application of quantitative tools.

iii. Library, Information, and Media Sciences: Courses from this category will help the students to understand the recent developments in information and media science (journalism, mass media, and communication)

iv. Commerce and Management: Courses include business management, accountancy, finance, financial institutions, fintech, etc.,

v. Humanities and Social Sciences: The courses relating to Social Sciences, for example, Anthropology, Communication and Media, Economics, History, Linguistics, Political Science, Psychology, Social Work, Sociology, etc. will enable students to understand the individuals and their social behaviour, society, and nation. Students be introduced to survey methodology and available large-scale databases for India. The courses under humanities include, for example, Archaeology, History, Comparative Literature, Arts & Creative expressions, Creative Writing and Literature, language(s), Philosophy, etc., and interdisciplinary courses relating to humanities. The list of Courses can include interdisciplinary subjects such as Cognitive Science, Environmental Science, Gender Studies, Global Environment & Health, International Relations, Political Economy and Development, Sustainable Development, Women's, and Gender Studies, etc. will be useful to understand society.

1.3.6. Ability Enhancement Courses (AEC):

Modern Indian Language (MIL) & English language focused on language and communication skills. Students are required to achieve competency in a Modern Indian Language (MIL) and in the English language with special emphasis on language and communication skills. The courses aim at enabling the students to acquire and demonstrate the core linguistic skills, including critical reading and expository and academic writing skills, that help students articulate their arguments and present their thinking clearly and coherently and recognize the importance of language as a mediator of knowledge and identity. They would also enable students to acquaint themselves with the cultural and intellectual heritage of the chosen MIL and English language, as well as to provide a reflective understanding of the structure and complexity of the language/literature related to both the MIL and English language. The courses will also emphasize the development and enhancement of skills such as communication, and the ability to participate/conduct discussion and debate.

1.3.7. Skill Enhancement Course (SEC):

These courses are aimed at imparting practical skills, hands-on training, soft skills, etc., to enhance the employability of students and should be related to Major Discipline. They will aim at providing hands- on training, competencies, proficiency, and skill to students. SEC course will be a basket course to provide skill-based instruction. For example, SEC of English Discipline may include Public Speaking, Translation & Editing and Content writing. A student shall have the choice to choose from a list, a defined track of courses offered from 1st to 3rd semester.

1.3.8. Value-Added Courses (VAC):

i. Understanding India: The course aims at enabling the students to acquire and demonstrate the knowledge and understanding of contemporary India with its historical perspective, the basic framework of the goals and policies of national development, and the constitutional obligations with special emphasis on constitutional values and fundamental rights and duties. The course would also focus on developing an understanding among student-teachers of the Indian knowledge systems, the Indian education system, and the roles and obligations of teachers to the nation in general and to the school/community/society. The course will attempt to deepen knowledge about and understanding of India's freedom struggle and of the values and ideals that it represented to develop an appreciation of the contributions made by people of all sections and regions of the country, and help learners understand and cherish the values enshrined in the Indian Constitution and to prepare them for their roles and responsibilities as effective citizens of a democratic society.

ii. Environmental science/education: The course seeks to equip students with the ability to apply the acquired knowledge, skills, attitudes, and values required to take appropriate actions for mitigating the effects of environmental degradation, climate change, and pollution, effective waste management, conservation of biological diversity, management of biological resources, forest and wildlife conservation, and sustainable development and living. The course will also deepen the knowledge and understanding of India's environment in its totality, its interactive processes, and its effects on the future quality of people's lives.

iii. Digital and technological solutions: Courses in cutting-edge areas that are fast gaining prominences, such as Artificial Intelligence (AI), 3-D machining, big data analysis, machine learning, drone technologies, and Deep learning with important applications to health, environment, and sustainable living that will be woven into undergraduate education for enhancing the employability of the youth.

iv. Health & Wellness, Yoga education, sports, and fitness: Course components relating to health and wellness seek to promote an optimal state of physical, emotional, intellectual, social, spiritual, and environmental well-being of a person. Sports and fitness activities will be organized outside the regular institutional working hours. Yoga education would focus on preparing the students physically and mentally for the integration of their physical, mental, and spiritual faculties, and equipping them with basic knowledge about one's personality, maintaining self-discipline and self-control, to learn to handle oneself well in all life situations. The focus of sports and fitness components of the courses will be on the improvement of physical fitness including the improvement of various components of physical and skills-related fitness like strength, speed, coordination, endurance, and flexibility; acquisition of sports skills including motor skills as well as basic movement skills relevant to a particular sport; improvement of tactical abilities; and improvement of mental abilities.

These are a common pool of courses offered by different disciplines and aimed towards embedding ethical, cultural and constitutional values; promote critical thinking. Indian knowledge systems; scientific temperament of students.

1.3.9. Summer Internship /Apprenticeship:

The intention is induction into actual work situations**.** All students must undergo internships / Apprenticeships in a firm, industry, or organization or Training in labs with faculty and researchers in their own or other HEIs/research institutions during the***summer term***. Students should take up opportunities for internships with local industry, business organizations, health and allied areas, local governments (such as panchayats, municipalities), Parliament or elected representatives, media organizations,artists, crafts persons, and a wide variety of organizations so that students may actively engage with the practical side of their learning and, as a by-product, further improve their employability. Students who wish to exit after the first two semesters will undergo a 4-credit work-based learning/internship during the summer term to get a UG Certificate.

1.3.9.1. Community engagement and service: The curricular component of 'community engagement and service' seeks to expose students to the socio-economic issues in society so that the theoretical learnings can be supplemented by actual life experiences to generate solutions to real-life problems. This can be part of summer term activity or part of a major or minor course depending upon the major discipline.

Field-based learning/minor project: The field-based learning/minor project will attempt to provide opportunities for students to understand the different socio-economic contexts. It will aim at giving students exposure to development-related issues in rural and urban settings. It will provide opportunities for students to observe situations in rural and urban contexts, and to observe and study actual field situations regarding issues related to socioeconomic development. Students will be given opportunities to gain a first- hand understanding of the policies, regulations, organizational structures, processes, and programmes that guide the development process. They would have the opportunity to gain an understanding of the complex socio-economic problems in the community, and innovative practices required to generate solutions to the identified problems. This may be a summer term project or part ofa major or minor course depending on study.

1.3.10.Indian Knowledge System:

In view of the importance accorded in the NEP 2020 to rooting our curricula and pedagogy in the Indian context all the students who are enrolled in the four-year UG programmes should be encouraged to take an adequate number of courses in IKS so that the ***total credits of the courses taken in IKS amount to at least five per cent of the total mandated credits*** *(i.e. min. 8 credits for a 4 yr. UGP & 6 credits for a 3 yr. UGP)*. The students may be encouraged to take these courses, preferably *during the first four semesters of the UG programme*. At least half of these mandated credits should be in courses in disciplines which are part of IKS and are related to the major field of specialization that the student is pursuing in the UG programme. They will be included as a part of the total mandated credits that the student is expected to take in the major field of specialization. The rest of the mandated credits in IKS can be included as a part of the mandated Multidisciplinary courses that are to be taken by every student. All the students should take a Foundational Course in Indian Knowledge System, which is designed to present an overall introduction to all the streams of IKS relevant to the UG programme. The foundational IKS course should be broad-based and cover introductory material on all aspects.

Wherever possible, the students may be encouraged to choose a suitable topic related to IKS for their project work in the 7/8th semesters of the UG programme.

(Note: Refer "Guidelines for Incorporating Indian Knowledge in Higher Education Curricula", University Grants Commission, March 2023 for further details)

1.3.11. Experiential Learning:

One of the most unique, practical & beneficial features of the National Credit Framework is assignment of credits/credit points/ weightage to the experiential learning including relevant experience and professional levels acquired/ proficiency/ professional levels of a learner/student. Experiential learning is of two types:

Experiential learning as part of the curricular structure of academic or vocational program. E.g., projects/OJT/internship/industrial attachments etc. This could be either within the Program- internship/ summer project undertaken relevant to the program being studied or as a part time employment (not relevant to the program being studied- up to certain NSQF level only). In the case where experiential learning is a part of the curricular structure the credits would be calculated and assigned as per basic principles of NCrF i.e., 40 credits for 1200 hours of notional learning.

a. Experiential learning as active employment (both wage and self) post completion of an academic or vocational program. This means that the experience attained by a person after undergoing a particular educational program shall be considered for assignment of credits. This could be either Full or Part time employment after undertaking an academic/ Vocation program.

In case where experiential learning is as a part of employment the learner would earn credits as weightage. The maximum credit points earned in this case shall be double of the credit points earned with respect to the qualification/ course completed. The credit earned and assigned by virtue of relevant experience would enable learners to progress in their career through the work hours put in during a job/employment

Section 2

Award of Degree

The structure and duration of undergraduate programmes of study offered by the University as per NEP 2020 include:

2.1.1. Undergraduate programmes of either 3 or 4-year duration with Single Major, with multiple entry and exit options, with appropriate certifications:

2.1.2. UG Certificate: Students who opt to exit after completion of the first year and have secured 40 credits will be awarded a UG certificate if, in addition, they complete one vocational course of 4 credits during the summer vacation of the first year. These students are allowed to re-enter the degree programme within three years and complete the degree programme within the stipulated maximum period of seven years.

2.1.3. UG Diploma: Students who opt to exit after completion of the second year and have secured 80 credits will be awarded the UG diploma if, in addition, they complete one vocational course of 4 credits during the summer vacation of the second year. These students are allowed to re-enter within a period of three years and complete the degree programme within the maximum period of seven years.

2.1.4. 3-year UG Degree: Students who will undergo a 3-year UG programme will be awarded UG Degree in the Major discipline after successful completion of three years, securing 120 credits and satisfying the minimum credit requirement.

2.1.5. 4-year UG Degree (Honours): A four-year UG Honours degree in the major discipline will be awarded to those who complete a four-year degree programme with 160 credits and have satisfied the credit requirements as given in Table 6 in Section 5.

2.1.6. 4-year UG Degree (Honours with Research): Students who secure 75% marks and above in the first six semesters and wish to undertake research at the undergraduate level can choose a research stream in the fourth year. They should do a research project or dissertation under the guidance of a Faculty Member of the University. The research project/dissertation will be in the major discipline. The students who secure 160 credits, including 12 credits from a research project/dissertation, will be awarded UG Degree (Honours with Research).

2.1.7. (Note: ***UG Degree Programmes with Single Major:*** A student must secure a minimum of 50% credits from the major discipline for the 3-year/4-year UG degree to be awardeda single major. For example, in a 3-year UG programme, if the total number of credits to be earned is 120, a student of Mathematics with a minimum of 60 credits will be awarded a B.Sc. in Mathematics with a single major. Similarly, in a 4-year UG programme, if the total number of credits to be earned is 160, a student of Chemistry with a minimum of 80 credits will be awarded a B.Sc. (Hons./Hon. With Research) in Chemistry in a 4-year UG programme with single major. Also the **4-year Bachelor's degree programme with Single Major** is considered as the preferred option since it would allow the opportunity to experience the full range of holistic and multidisciplinary education in addition to a focus on the chosen major and minors as per the choices of the student.)

Table: 1: Award of Degree and Credit Structure with ME-ME

Award	**Year**	**Credits to earn**	**Additional Credits**	**Re-entry allowed within (yrs)**	**Years to Complete**
UG Certificate	1	40	4	3	7
UG Diploma	2	80	4	3	7
3-year UG Degree (Major)	3	120	x	x	x
4-year UG Degree (Honours)	4	160	x	x	x
4-year UG Degree (Honors with Research):	4	160	Students who secure cumulative 75% marks and above in the first six semesters		

Section 3

Credit, Credit Points & Credit hours for different types of courses

3.1. Introduction:

'Credit' is recognition that a learner has completed a prior course of learning, corresponding to a qualification at a given level. For each such prior qualification, the student would have put in a certain volume of institutional or workplace learning, and the more complex a qualification, the greater the volume of learning that would have gone into it. Credits quantify learning outcomes that are subject achieving the prescribed learning outcomes to valid, reliable methods of assessment.

The ***credit points*** will give the learners, employers, and institutions a mechanism for describing and comparing the learning outcomes achieved. The credit points can be calculated as credits attained multiplied with the credit level.

The workload relating to a course is measured in terms of credit hours. A credit is a unit by which the coursework is measured. It determines the number of hours of instruction required per week over the duration of a semester (minimum 15 weeks). Each course may have only a lecture component or a lecture and tutorial component or a lecture and practicum component or a lecture, tutorial, and practicum component, or only practicum component.

A course can have a combination of ***lecture credits, tutorial credits, practicum credits and experiential learning credits.*** The following types of courses/activities constitute the programmes of study. Each of them will require a specific number of hours of teaching/guidance and laboratory/studio/workshop activities, field-based learning/projects, internships, and community engagement and service.

- **Lecture courses:** Courses involving lectures relating to a field or discipline by an expert or qualified personnel in a field of learning, work/vocation, or professional practice.
- **Tutorial courses:** Courses involving problem-solving and discussions relating to a field or discipline under the guidance of qualified personnel in a field of learning, work/vocation, or professional practice. Should also refer to the Remedial Classes, flip classrooms and focus on both Slow and Fast Learners of the class according to their merit.

racticum or Laboratory work: A course requiring students to participate in a project or practical or lab activity that applies previously learned/studied principles/theory related to the chosen field of learning, work/vocation, or professional practice under the supervision of an expert or qualified individual in the field of learning, work/vocation or professional practice.

- **Seminar:** A course requiring students to participate in structured discussion/conversation or debate focused on assigned tasks/readings, current or historical events, or shared experiences guided or led by an expert or qualified personnel in a field of learning, work/vocation, or professional practice.
- **Internship:** A course requiring students to participate in a professional activity or work experience, or cooperative education activity with an entity external to the education institution, normally under the supervision of an expert of the given external entity. A key aspect of the internship is induction into actual work situations. Internships involve working with local industry, government or private organizations, business organizations, artists, crafts persons, and similar entities to provide opportunities for students to actively engage in on-site experiential learning.
- **Field practice/projects:** Courses requiring students to participate in field-based learning/projects generally under the supervision of an expert of the given external entity.
- **Community engagement and service:** Courses requiring students to participate in field-based learning/projects generally under the supervision of an expert of the given external entity. The curricular component of 'community engagement and service' will involve activities that would expose students to the socio-economic issues in society so that the theoretical learnings can be supplemented by actual life experiences to generate solutions to real-life problems.

Table:2: Course wise Distribution of Credits

Broad Category ofCourse	**Minimum Credit Requirement**	
	3-year UG	**4-Year UG**
Major (Core)	60	80
Minor Stream	24	32
Interdisciplinary	9	9
Ability Enhancement Courses (AEC)	8	8
Skill Enhancement Courses (SEC)	9	9
Value Added Coursescommon for all UG	6	6
Summer Internship	4	4
Research Project /Dissertation	NA	12
Total	**120**	**160**

Table 3: Credit Distribution for 3-year Course

Semester	**Course Credits**							
	Major	**Minor**	**ID**	**AEC**	**SEC**	**VAC**	**SI**	**Total**
I	6	3	3	2	3	3	0	**20**
II	6	3	3	2	3	3	0	**20**
III	8	4	3	2	3	0	0	**20**
IV	12	6	0	2	0	0	0	**20**
V	12	4	0	0	0	0	4	**20**
VI	16	4	0	0	0	0	0	**20**
	60	**24**	**9**	**8**	**9**	**6**	**4**	**120**

Table 4: Credit Distribution for 4-year Course

Semester	Course Credits								Total
	Major	**Minor**	**ID**	**AEC**	**SEC**	**VAC**	**SI**	**RP**	
I	6	3	3	2	3	3	0	0	**20**
II	6	3	3	2	3	3	0	0	**20**
III	8	4	3	2	3	0	0	0	**20**
IV	12	6	0	2	0	0	0	0	**20**
V	12	4	0	0	0	0	4	0	**20**
VI	16	4	0	0	0	0	0	0	**20**
VII	16	4	0	0	0	0	0	0	**20**
VIII	4	4	0	0	0	0	0	12	**20**
	80	**32**	**9**	**8**	**9**	**6**	**4**	**12**	**160**

Section 4

Level of Courses

4.1 NHEQF levels:

The NHEQF levels represent a series of sequential stages expressed in terms of a range of learning outcomes against which typical qualifications are positioned/located. NHEQF level 4.5 represents learning outcomes appropriate to the first year (first two semesters) of the undergraduate programme of study, while Level 8 represents learning outcomes appropriate to the doctoral-level programme of study.

Table: 5: NHEQF Levels

NHEQF level	Examples of higher education qualifications located within each level	Credit Requirements
Level 4.5	Undergraduate Certificate. Programme duration: First year (first two semesters) of the undergraduate programme, followed by an exit 4-credit skills-enhancement course(s).	40
Level 5	Undergraduate Diploma. Programme duration: First two years (first four semesters) of the undergraduate programme, followed by an exit 4-credit skills-enhancement course(s) lasting two months.	80
Level 5.5	Bachelor's Degree. Programme duration: First three years (Six semesters) of the four-year undergraduate programme.	120
Level 6	Bachelor's Degree (Honours/ Honours with Research). Programme duration: Four years (eight semesters).	160
Level 6	Post-Graduate Diploma. Programme duration: One year (two semesters) for those who exit after successful completion of the first year (two semesters) of the 2-year master's programme	160
Level 6.5	Master's degree. Programme duration: Two years (four semesters) after obtaining a 3- year Bachelor's degree (e.g. B.A., B.Sc., B.Com. etc.).	80
Level 6.5	Master's degree. Programme duration: One year (two semesters) after obtaining a 4 -year Bachelor's degree (Honours/ Honours with Research) (e.g. B.A., B.Sc., B.Com. etc.).	40
Level 7	Master's degree. (e.g., M.E./M.Tech. etc.) Programme duration: Two years (four semesters) after obtaining a 4-year Bachelor's degree. (e.g., B.E./B.Tech. etc.)	80
Level 8	Doctoral Degree	Credits for course work, Thesis, and published work

4.2. Course Code based on Learning Outcomes:

Courses are coded based on the learning outcomes, level of difficulty, and academic rigor. The coding structure is as follows:

i. 0-99: ***Pre-requisite courses*** required to undertake an introductory course which will be a pass or fail course with no credits. It will replace the existing informal way of offering bridge courses that are conducted in some of the colleges/ universities.

ii. 100-199: ***Foundation or introductory courses*** that are intended for students to gain an understanding and basic knowledge about the subjects and help decide the subject or discipline of interest. These courses may also be prerequisites for courses in the major subject. These courses generally would focus on foundational theories, concepts, perspectives, principles, methods, and procedures of critical thinking in order to provide a broad basis for taking up more advanced courses.

iii. 200-299: ***Intermediate-level courses*** including subject-specific courses intended to meet the credit requirements for minor or major areas of learning. These courses can be part of a major and can be pre-requisite courses for advanced-level major courses.

iv. 300-399: ***Higher-level courses*** which are required for majoring in a disciplinary/interdisciplinary area of study for the award of a degree.

v. ***400-499: Advanced courses*** which would include lecture courses with practicum, seminar-based course, term papers, research methodology, advanced laboratory experiments/software training, research projects, hands-on-training, internship/apprenticeship projects at the undergraduate level or First year post-graduate theoretical and practical courses.

vi. 500-599: ***Courses at first-year PG degree level*** for a 2-year post-graduate degree programme

vii. 600-699: ***Courses for second year of 2-year PG*** or 1-year post-graduate degree programme

viii. 700 -799 & above: Courses limited to doctoral students.

Section 5

Graduate Attributes & Learning Outcomes

5.1 Introduction

As per the NHEQF, each student on completion of a programme of study must possess and demonstrate the expected ***Graduate Attributes*** acquired through one or more modes of learning, including direct in-person or face-to-face instruction, online learning, and hybrid/blended modes. The graduate attributes indicate the quality and features or characteristics of the graduate of a programme of study, including learning outcomes relating to the disciplinary area(s) relating to the chosen field(s) of learning and generic learning outcomes that are expected to be acquired by a graduate on completion of the programme(s) of study.

The graduate profile/attributes must include,

- capabilities that help widen the current knowledge base and skills,
- gain and apply new knowledge and skills,
- undertake future studies independently, perform well in a chosen career, and
- play a constructive role as a responsible citizen in society.

The graduate profile/attributes are acquired incrementally through development of cognitive levels and describe a set of competencies that are transferable beyond the study of a particular subject/disciplinary area and programme contexts in which they have been developed.

Graduate attributes include,

- ***learning outcomes that are specific to disciplinary areas*** relating to the chosen field(s) of learning within broad multidisciplinary/interdisciplinary/ transdisciplinary contexts.
- ***generic learning outcomes*** that graduate of all programmes of study should acquire and demonstrate.

5.2 Graduate Attributes:

Table: 6: The Learning Outcomes Descriptors and Graduate Attributes

Sl.no.	Graduate Attribute	The Learning Outcomes Descriptors *(The graduates should be able to demonstratethe capability to:)*
GA1	Disciplinary Knowledge	acquire knowledge and coherent understanding of the chosen disciplinary/interdisciplinary areasof study.
GA 2	Complex problem solving	solve different kinds of problems in familiar and non-familiar contexts and apply the learning to real-life situations.
GA 3	Analytical & Critical thinking	apply analytical thought including the analysis and evaluation of policies, and practices. Able to identify relevant assumptions or implications. Identify logical flaws and holes in the arguments of others. Analyse and synthesize data from a variety of sources and draw valid conclusions and support them with evidence and examples.
GA 4	Creativity	create, perform, or think in different and diverse ways about the same objects or scenarios and deal with problems and situations that do not have simple solutions. Think 'out of the box' and generate solutions to complex problems in unfamiliar contexts by adopting innovative, imaginative, lateral thinking, interpersonal skills, and emotional intelligence.
GA 5	CommunicationSkills	listen carefully, read texts and research papers analytically, and present complex information in a clear and concise manner to different groups/audiences. Express thoughts and ideas effectively in writing and orally and communicate with others using appropriate media.
GA 6	Research-relatedskills	develop a keen sense of observation, inquiry, and capability for asking relevant/ appropriate questions. Should acquire the ability to problematize, synthesize and articulate issues and design research proposals, define problems, formulate appropriate and relevant research questions, formulate hypotheses, test hypotheses using quantitative and qualitative data, establish hypotheses, make inferences based on the analysis and interpretation of data, and predict cause-and-effect relationships. Should develop the ability to acquire the understanding of basic research ethics and skills in practicing/doing

		ethics in the field/ in personal research work.
GA 7	Collaboration	work effectively and respectfully with diverse teams in the interests of a common cause and work efficiently as a member of a team.
GA 8	Leadership readiness/qualities	plan the tasks of a team or an organization and setting direction by formulating an inspiring vision and building a team that can help achievethe vision.
GA 9	Digital and technological skills	use ICT in a variety of learning and work situations. Access, evaluate, and use a variety of relevant information sources and use appropriate software for analysis of data.
GA 10	Environmental awareness and action	mitigate the effects of environmental degradation, climate change, and pollution. Should develop the technique of effective waste management, conservation of biological diversity, management of biological resources and biodiversity, forest and wildlife conservation, and sustainable development and living.

5.3 Programme Learning Outcomes (PLO)

The term 'programme' refers to the entire scheme of study followed by learners leading to a qualification. Individual programmes of study will have defined learning outcomes that must be attained for the award of a specific certificate/diploma/degree. Programme Learning Outcomes describe what students are expected to know or be able to do by the time of graduation. PLOs are statements about the knowledge, skills and attitudes (attributes) the graduate of a formal engineering program should have. PLOs deal with the general aspect of graduation for a particular program, and the competencies and expertise a graduate will possess after completion of the program. The identified PLOs are as follows

- **PLO1- Knowledge of Computer Application:** Acquiring knowledge on basics of Computer Science and ability to apply to design principles in the development of solutions for problems of varying complexity.
- **PLO2- Ability to Solve Complex Problems:** Improved reasoning with strong mathematical ability to Identify, formulate and analyze problems related to computer science and exhibiting a sound knowledge on data structures and algorithms.
- **PLO3- Analytical and Critical Thinking:** Ability to devise and conduct experiments, interpret data and provide well informed conclusions.

Possessing a sound knowledge on computer application software and design and develop app for applicative problems.

- **PLO4- Develop and Demonstrate Creativity:** Ability to design and development of algorithmic solutions to real world problems and acquiring a minimum knowledge on statistics and optimization problems. Establishing excellent skills in applying various design strategies for solving complex problems.
- **PLO5- Enhance Communication Skills:** Must have a reasonably good communication knowledge both in oral and writing.
- **PLO6- Formulate Research-Related Skills:** Develop the ability to identify social problems and issues and devise realistic solutions through detailed research by following ethics.
- **PLO7- Develop Ability to Collaborate:** Develop the ability to work in teams and figure out the possibilities to collaborate with relevant organisations while solving the problems at hand.
- **PLO8- Develop Leadership Qualities:** Develop the qualities that will help in leading a team efficiently and extract maximum throughput out of team effort.
- **PLO9- Execute Digital and Technological Skills:** Identify, select and use a modern scientific and IT tool or technique for modeling, prediction, data analysis and solving problems in the area of Computer Science and making them mobile based application software.
- **PO10- Identify Environmental Issues, Develop Awareness and Action:** Exhibiting professional ethics to maintain the integrity in a working environment and also have concern on societal impacts due to computer-based solutions for problems.

5.4 Programme Specific Outcomes (PSOs)

- **PSO1- Knowledge of Computing Systems:** An ability to understand the principles and working of computer systems.
- **PSO2- Project Development Skills:** An ability to understand the structure and development methodologies of software systems.
- **PSO3: Software Development Skills:** Familiarity and practical

competence with a broad range of programming language and open-source platforms.

- **PSO4: Mathematical Skills:** An ability to apply mathematical methodologies to solve computation task, model real world problem using appropriate data structure and suitable algorithm.

5.4 Course Learning Outcomes (CLOs)

The programme learning outcomes are attained by learners through the essential learnings acquired on the completion of selected courses of study within a programme of study. The term 'course' is used to mean the individual courses of study that make up the scheme of study for a programme.

Course learning outcomes are specific to the learning for a given course of study related to a disciplinary or interdisciplinary/multi-disciplinary area of learning. Some courses of study are highly structured, with a closely laid down progression of compulsory/core courses to be taken at different phases/stages of learning.

5.5 The Qualification Specifications:

Table: 5: NHEQF Qualification specifications

Qualification type	Purpose of the qualification
Undergraduate Certificate	The students will be able to apply technical and theoretical concepts and specialized knowledge and skills in a broad range of contexts to undertake skilled or paraprofessional work and/or to pursue further study/learning at higher levels.
Undergraduate Diploma	The students will be able to apply specialized knowledge in a range of contexts to undertake advanced skilled or paraprofessional work and/or to pursue further learning/study at higher levels.
Bachelor's degree	The students will be able to apply a broad and coherent body of knowledge and skills in a range of contexts to undertake professional work and/or for further learning.
Bachelor's degree (Honours/ Honours with Research)	The students will be able to apply the knowledge in a specific context to undertake professional work and for research and further learning.
	The students will be able to apply an advanced body of knowledge in a range of contexts to undertake professional work and apply specialized knowledge and skills for research and scholarship, and/or for further learning relating to the chosen field(s) of learning, work/vocation, or professional practice.

Section 6

Course Structure and Syllabus of the Framework

6.1 Course Structure of BCA

BCA Course Structure for the Session 2023-2024

1st Semester				
Sl. No.	**Subject Code**	**Names of subjects**	**Level of Course**	**Credit**
Major				
1	CAP052M101	Discrete Structures	100	3
2	CAP052M102	Introduction to C Programming	100	3
Minor				
3	CAP052N101	Fundamentals of Web Design (Offered to Others)	100	3
Interdisciplinary				
4	IKS992K101	Introduction to Indian Knowledge System-I	100	3
Ability Enhancement Courses (AEC)				
5	AEC982A101	Communicative English and Behavioural Science-I	100	2
Skill Enhancement Courses (SEC)				
6	CAP052S101	Windows Programming using C#	100	3
Value Addition Courses (VAC)				
7	VAC-1	Basket Course (Office Automation -RSIT) (List Offered by University)	100	3
		TOTAL		**20**
2nd Semester				
Sl. No.	**Subject Code**	**Names of subjects**	**Level of Course**	**Credit**
Major				
1	CAP052M201	Data Structures	100	3
2	CAP052M202	Computer Architecture	100	3
Minor				
3	INT052N201	Server-Side Programming ((Offered to Others))	100	3
Interdisciplinary				
4	IKS992K201	Introduction to Indian Knowledge System-II	100	3
Ability Enhancement Courses (AEC)				

5	AEC982A201	Communicative English and Behavioural Science-II	100	2
Skill Enhancement Courses (SEC)				
6	CAP052S201	Computer Hardware and Networking	100	3
Value Addition Courses (VAC)				
7	VAC992V2409	Basket Course (Cybersecurity - RSIT) (List Offered by University)	100	3
		TOTAL	**17**	**20**
3rd Semester				
Sl. No.	**Subject Code**	**Names of subjects**	**Level of Course**	**Credit**
Major				
1	CAP052M301	Java Programming	200	4
2	CAP052M302	Database Management Systems	200	4
Minor				
3	CAP052N301	Font End Development with React (Offered to Others)	200	4
Interdisciplinary				
4	INT052I301	Introduction to Python (Offered to all by RSIT)	200	3
Ability Enhancement Courses (AEC)				
5	AEC982A101	Communicative English and Behavioural Science-III	200	2
Skill Enhancement Courses (SEC)				
6	CAP052S301	SEC-3 (System Administration)	200	3
		TOTAL		**20**
4th Semester				
Sl. No.	**Subject Code**	**Names of subjects**	**Level of Course**	**Credit**
Major				
1	CAP052M401	Operating Systems	200	4
2	CAP052M402	Data Communication Networks	200	4
3	INT052M402	Indian Mathematics in Computer Science	200	4
Minor				
4	CAP052N401	Front End Development with Angular (to be offered for other)	200	3
5	CAP052N402	Server-Side Programming with Node JS	200	3
Ability Enhancement Courses (AEC)				
6	AEC982A101	Communicative English and Behavioural Science-IV	200	2
		TOTAL	20	**20**
5th Semester				
Sl. No.	**Subject Code**	**Names of subjects**	**Level of Course**	**Credit**
Major				
1	C-301	Major Core 1 (High level)	300	4
2	C-302	Major Core 2 (High level)	300	4

3	C-303	Major Core 3 (High level)	300	4
Minor				
4	M-301	Minor 6	300	4
Internship				
5		Internship	300	4
		TOTAL		**20**
6th Semester				
Sl. No.	**Subject Code**	**Names of subjects**	**Level of Course**	**Credit**
Major				
1	C-304	Major Core 4 (High level)	300	4
2	C-305	Major Core 5 (High level)	300	4
3	C-306	Major Core 6 (High level)	300	4
4	C-307	Major Core 7 (High level)	300	4
Minor				
5	M-302	Minor 7	300	4
		TOTAL		**20**
7th Semester				
Sl. No.	**Subject Code**	**Names of subjects**	**Level of Course**	**Credit**
Major				
1	C-401	Major Core 1 (Advanced)	400	4
2	C-402	Major Core 2 (Advanced)	400	4
3	C-403	Major Core 3 (Advanced)	400	4
4	C-404	Major Core 4 (Advanced)	400	4
Minor				
5	M-401	Minor 8	400	4
		TOTAL	**20**	**20**
8th Semester				
Sl. No.	**Subject Code**	**Names of subjects**	**Level of Course**	**Credit**
Major				
1	RM-301	Major Core 5 (High level)	400	4
Minor				
2	M-402	Research Methodology	400	4
Dissertation				
3		Dissertation	400	12
Advanced Level Core Course in-lieu of Dissertation				
4	C-407	Major Core 6 (Advanced Level)	400	4
5	C-408	Major Core 7 (Advanced Level)	400	4
6	C-409	Major Core 8 (Advanced Level)	400	4
		TOTAL		**20**

6.2 Detailed Syllabus of all the Courses Semester-wise

SYLLABUS (1st SEMESTER)

Paper I/Subject Name: Discrete Structures		**Subject Code: CAP052M101**
Course Type: Major		**Course Level:100**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Course Objective:

The objectives of the course are to make the students learn the concept of mathematical logic, sets, relations, and functions, generating functions and recurrence relations, Graph Theory for solving engineering related problems.

Prerequisites: None

Course Outcomes

On successful completion of the course the students will be able to:		
Sl No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** basic concepts of discrete structures and terminology used.	**BT 1**
CO 2	**Explain** applications of discrete structures in computer science	**BT 2**
CO 3	**Apply** concepts of Counting, Probability, Relations, and Graphs	**BT 3**
CO 4	**Analyse** and evaluate various methodology of proving a theorem.	**BT 4**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Sets, Relations and Functions**	Operations and Laws of Sets, Binary, Relation, Partial Ordering Relation, Equivalence Relation, Functions, Inverse and Composite Function, Finite and infinite Sets, Countable and uncountable Sets, Poset, Lattice. The Well-Ordering Principle, The Division algorithm: Prime numbers, The Greatest Common Divisor, The least common multiple, Euclidean Algorithm, The Fundamental Theorem of Arithmetic, Congruence, Euler's phi function.	**15**
II	**Graph Theory and Combinatorics**	Graphs and their properties, Degree, subgraphs, walks, paths and circuits, connected and disconnected graphs, Isomorphism, Eulerian and Hamiltonian graphs, Complete graphs, Bipartite graph, Trees, Properties of trees, Pendant vertex, Distance and Centers, Binary tree, Spanning trees, Planar graphs, Matrix representation of graphs, Chromatic number, Chromatic polynomial, Five colors theorem. Pigeon-hole principle, permutation and combination, Recurrence relations, Generating functions.	**15**
III	**Algebraic Structures**	Algebraic Structures with one Binary Operation, Semi Groups, Monoids, Groups, Congruence Relation and Quotient Structures, Free and Cyclic Monoids and Groups, Permutation Groups, Normal Subgroups, Algebraic Structures with two Binary Operation, Rings, Integral Domain and Fields.	**15**

IV	**Propositional Logic**	Proposition, connectives, tautology, contradiction, logical equivalence, normal forms-DNF, CNF, argument, Validity of argument, fallacy, Rules of Inference, Quantifiers. Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function.	**15**
Total			**60**

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3*20 NCH = 60 NCH	-	30 NCH (Problem Solving, Internship, Seminar, Case Study, Discussion)

Text Books:

1. *A text book of Discrete Mathematics*, Sarkar S. K., Revised Edition, 2016, S Chand & Co Ltd.

Reference Books:

1. Deo N; *Graph Theory with applications to engineering and computer science*, New Edition, 2009, PHI Learning Private Limited.
2. Chandrasekaran N. and Umaparvathi, *Discrete Mathematics*, Eastern Economic Edition, 2013, PHI
3. *Discrete Mathematics and its Applications,* Rosen, K.H., 6th Edition, 2006, McGraw Hill.
4. Tremblay, J.P. and Manohar, R., *Discrete Mathematical Structures with Applications to Computer Science*, 35th Reprint, 2007, Tata McGraw Hill

Paper II/Subject Name: Introduction to C Programming **Subject Code: CAP052M102**

Course Type: Major **Course Level:100**

L-T-P-C – 2-0-2-3 **Credit Units: 03** **Scheme of Evaluation: TP**

Objective:

The objectives of the course are to give the students exposure to computer programming and make them capable of using the concepts to solve basic as well as advanced computing problems.

Prerequisites: None

Course Outcomes

On successful completion of the course the students will be able to:		
Sl No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** different constructs and building blocks present in C programming Language	**BT 1**
CO 2	**Demonstrate** the working of C programming language and its control structures.	**BT 2**
CO 3	**Apply** the programming concepts to solve various problems.	**BT 3**
CO 4	**Analyse** and debug the errors while writing the programs.	**BT 4**
CO 5	**Assess** and **design** a new algorithm to solve a new real-life problem	**BT 5**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**C Programming Fundamentals**	History and importance of C language, Basic structure of programs, programming style, execution of C programs. Character set, Tokens, Keywords and Identifiers, Constants, variables, data types, statements, comments, declaration of storage class, assigning values to variables. Basic idea of Computer Algorithms and Flow Charts. Managing I/O, reading and writing characters, formatted Input/output. Arithmetic operators, relational operators, logical operators, assignment operators, increment & decrement operators, conditional operators, bitwise operators, special operators. Arithmetic expressions, operator precedence & associativity.	**11**
II	**Decision Making, Branching & Lopping**	Importance of decision making, decision making with *if* statement, *if-else* statement, nested *if-else* statements, *switch-case* statement, *goto* statement, the *?:* operator, examples. Importance of loping, the *while* statement, *do-while* statement, *for* statement, nested looping, examples.	**11**
III	**Arrays, Strings & User-Defined Functions**	Significance of Arrays, creation and use of one- & two-dimensional arrays, Dynamic arrays. Declaration and use of string variables, reading and writing strings, operations on strings. Benefits of user-defined functions, creation and use of user-defined functions, parameter passing, return types.	**11**

IV	**Advanced Programming Concepts**	Creation and use of Structures and Unions in programs. Introduction to Pointers, declaration & initialization of pointer variables, accessing a variable through its pointer. Defining, opening & closing files in C, Input/output operations on files.	**11**
Total			**44**

Introduction to C Programming Lab Syllabus

Total Lab Hours for the semester = 30 (2 hours per week)

Minimum 20 Laboratory experiments based on the following-

1. Character set, Tokens, Keywords and Identifiers, Constants, variables, data types, statements, comments, declaration of storage class, assigning values to variables.
2. Managing I/O, reading and writing characters, formatted Input/output.
3. Arithmetic operators, relational operators, logical operators, assignment operators, increment & decrement operators, conditional operators, bitwise operators, special operators. Arithmetic expressions, operator precedence & associativity.
4. Importance of decision making, decision making with if statement, if-else statement, nested if-else statements, switch-case statement, goto statement, the ?: operator.
5. Importance of loping, the while statement, do-while statement, for statement, nested looping.
6. Significance of Arrays, creation and use of one & two dimensional arrays, Dynamic arrays.
7. Declaration and use of string variables, reading and writing strings, operations on strings.
8. Benefits of user-defined functions, creation and use of user-defined functions, parameter passing, return types.
9. Creation and use of Structures and Unions in programs.
10. Use of Pointers, declaration & initialization of pointer variables, accessing a variable through its pointer.
11. Defining, opening & closing files in C, Input/output operations on files.

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
2 * 22 NCH = 44 NCH	2 * 15 NCH = 30 NCH	8 * 2 NCH = 16 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

T ext Book:

1. *Computer Fundamentals and Programming in C,* Reema Thareja, 2nd Edition, 2016, Oxford University Press, Delhi.

Reference Books:

1. E Balaguruswamy, *Computing Fundamentals and C Programming*, 1st Edition, 2017, McGraw Hill.
2. Venugopal and Prasad, *Mastering C*, 2nd Edition, 2017, Tata McGraw Hill.
3. Yashawant Kanetkar, *Let us C,* 15th Edition, 2017, BPB Publication.

- **Detailed Syllabus of the Minor Course**

Paper III/Subject Name: Fundamentals of Web Design	**Subject Code: CAP052N101**
Course Type: Minor	**Course Level:100**

L-T-P-C – 3-0-0-3 **Credit Units: 03** **Scheme of Evaluation: T**

Objective:

The objectives of the course are to enable the students to build a robust foundation for computational thinking and make them learn client-side web development.

Prerequisites: None

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** basic building and design blocks of an website.	**BT 1**
CO 2	**Understand** the basic characteristics and concepts of web development.	**BT 2**
CO 3	**Build** static web pages and manipulate data using JavaScript and work with the HTML Canvas	**BT 3**
CO 4	**Analyse** and **evaluate** websites in terms of its design and basic processing at the client side.	**BT 4 & 5**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Introduction to Web and creating website**	The Internet: Client & Server, IP address and URL, The World Wide Web (WWW), Installing Visual Studio Code, Installing the Prettier VSCode extension, Install Ubuntu in Windows, using WSL, Install Ubuntu using virtual machine software, making and hosting website. Introduction to HTML tags, Looking inside websites using "Inspect Element"	**15**
II	**Styling and Working with Strings**	Working with modern HTML and CSS to produce an attractive, informative multi-page website based on the client's requirements, Creating a multipage website using HTML5, Control the look of a website using CSS, Formating a web page to display complex information, Adding graphical elements and maps to a website, Implement web forms to capture user input, Testing a website for compliance with standards and to ensure that it works with a range of browsers, Implementation of CSS using Bootstrap, Styling and Working with Strings: Introduction to strings, Joining strings together, Switching to the VSCode editor: Putting HTML and JS together, Adding comments to HTML and JS, Find the length of a string, Search for a string inside another string, String equality comparison, Sort a collection of strings, Split strings by a pattern,	**15**
III	**Functions**	Numbers, Booleans, Objects and Arrays, Number Data Type, Numbers Boolean Data Type, Boolean - comparisons and logical operations, Object Literals - create, read & update + nesting objects, Arrays - handling ordered values, Functions: Explicitly return a value from a function, Passing a function as an argument , introduction to Firebase.	**15**

IV	**Advanced Techniques of JavaScript**	Iterating over Arrays: Iterating over an array using the for each method, Generate an HTML list from an array, Using the index of the array value during iteration, Nested Array iteration, Transforming Arrays, Generate an HTML list from an array using the map function, Using index of array value with map, Transforming Nested Arrays, Filtering Arrays: Filter an array based on some criteria, A minimal UI for filtering flight search results, Use the index of the array value with filter, Building a game with Canvas,HTML canvas element, introduction to AJAX, JSON, RESTful API.	**15**
Total			**60**

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3*20 NCH = 60 NCH	-	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

Text Book:

1. *Internet and World Wide Web How to program*, Deitel H.M. and Deitel P.J, 4th Edition, 2012, Pearson International, New Delhi
2. *Web Technology*, Gopalan N.P. and Akilandeswari J., 2nd Edition, 2014, Prentice Hall of India, New Delhi.
3. *Java How to Program*, Paul Dietel and Harvey Deitel, 8th Edition, 2014, Prentice Hall of India, New Delhi

Reference Books:

1. Uttam K. Roy, *Web Technologies*, 2010, Illustrated Oxford University Press.
2. Godbole A. S. & Kahate A., *Web Technologies*, 2nd Edition, 2006, TMH, New Delhi.

- **Detailed Syllabus of Skill Enhancement Course (SEC-I)**

Paper VI/Subject Name: Windows Programming using C#		**Subject Code: CAP052S101**
Course Type: SEC		**Course Level: 100**
L-T-P-C – 2-0-2-3	**Credit Units: 03**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are to enable the students to learn concepts on C# and .NET framework.

Prerequisites: None

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** semantic syntax and control structures of C# and .NET	**BT 1**
CO 2	**Understand** introductory programming concepts using C#	**BT 2**
CO 3	**Apply** logical alternatives with C# decision structures utilizing iteration, class methods, fields, and properties.	**BT 3**
CO 4	**Simplify** forms, classes, and controls into C# solutions utilizing arrays and file/database access methods	**BT 4**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Introduction to .NET and C#**	Installation, Components of .NET, Common Language Specification (CLS), Common Language Runtime (CLR), Microsoft Intermediate Language ("MSIL" or "IL"), The Common Type System (CTS), .NET Framework Base Classes, Web Services, Web Forms, and Windows Forms, The .Net Languages, Execution of Sample Programs, Command Line Arguments, Programming Examples, And Multiple Main Methods. Keywords, Identifiers, Literals, Variables, Data Types, Boxing and Unboxing. Operator Precedence and Associativity, Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operators, Bitwise Operators, Special Operators, Type Conversions	**11**
II	**Branching, Looping and Methods**	Decision Making Statements, The Switch Statement, The? Operator, Decision Making and Looping, Jumps in Loops, Labelled Jumps, Single Dimensional Arrays, Multidimensional Arrays, Jagged Arrays, System. Array Class, Array List Class, Strings, Regular Expressions, Declaring Methods, Main Method, Invoking Methods, Nesting of Methods, Method Parameters	**11**
III	**Structures, Classes, Objects and OOP Concepts**	Defining a Structure, Assigning Values to Members, Copying Structures, Structures with Methods, Nested Structures, Classes Vs Structures, Guidelines to use Structures; Enumerations-Enumerator Initialization, Enumerator Base Types, Enumerator Type Conversion, Constructors & Destructors, Member Initialization, 'this' Reference Variable, Nesting of Classes, Members, Properties, Classical Inheritance, Containment Inheritance, defining a Subclass, Visibility Control, Subclass Constructor, Method Overriding, Hiding Methods, Abstract Classes, Abstract Methods, Sealed Classes, Sealed Methods, Polymorphism	**11**

IV	**Exception Handling, Interfaces and Windows Application**	An Overview, Exception Handling Syntax, Multiple Catch Statements, The Exception Hierarchy, General Catch Handler, using 'Finally', Nested Try Blocks, User Defined Exceptions, Operators – Checked and Unchecked, Defining Interfaces, Extending Interfaces, Implementing Interfaces, Explicit Interface Implementation, Abstract Classes and Interfaces, Delegates, Multicast Delegates, Events, The Console Class, Console Input and Output, Formatted Output, Custom Numeric Format. Developing Windows Applications, Developing Web Applications.	**11**
Total			**44**

Total Lab Hours for the semester = 30 (2 hours per week)

Minimum 20 Laboratory experiments based on the following-

- Basic C# programs
- Classes and Objects
- Inheritance
- Operator Overloading
- Threading, Events and Delegates
- Working with Windows Forms Controls, Validating data
- Creating Custom Dialog Box and Designing an MDI application with Menu
- Retrieving Data from Database & Working with Disconnected Environment

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
2 * 22 NCH = 44 NCH	2 * 15 NCH = 30 NCH	8 * 2 NCH = 16 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

Text Book:

1. *Programming in C#*, E Balagurusamy, 3rd Edition, 2010, Tata McGraw Hill, New Delhi

Reference Books:

1. Poul Klausen, *Introduction to programming and C# Language*, Bookbon, 1st 2012, New Delhi.

- **Detailed Syllabus of Indian Knowledge System -1 Course**

Paper IV/Subject Name: Introduction to Indian Knowledge System-I Subject Code: IKS992K101

Course Type: IKS **Course Level: 100**

L-T-P-C – 3-0-0-3 **Credit Units: 03** **Scheme of Evaluation: T**

Objectives:
This foundation course is designed to present an overall introduction to all the streams of IKS relevant to the UG programme. It would enable students to explore the most fundamental ideas that have shaped Indian Knowledge Traditions over the centuries.

Prerequisite: None

Course Outcomes:

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Illustrate** literature of Indian civilization-the Vedic – Itihasas, languages, mathematics, and Ayurveda.	**BT 2**
CO 2	**Explain** observation of the motion of celestial bodies in the Vedic corpus	**BT 3**

Modules	**Topics**	**Course content**	**Periods**
I	**Bharatavarsha —A Land of Rare Natural Endowments**	Demographical features of the ancient Bharatvarsha, Largest cultivable area in the world. Protected and nurtured by Himalayas. The Sindhu-Ganga plain and the great coastal plains. The great rivers of India. Climatic changes: Abundant rains, sunshine and warmth, vegetation, animals and mineral wealth. Most populous country in the world. India's prosperity held the world in thrall. Splendid geographical isolation of India and the uniqueness of Indian culture.	**10**
II	**Foundational Literature of Indian Civilization:**	The Vedic Corpus. The Itihasas— Ramayana and Mahabharata, and their important regional versions. The Puranas. Foundational Texts of Indian Philosophies, including the Jaina and Bauddha. Foundational Texts of Indian Religious Sampradayas, from the Vedic period to the Bhakti traditions of different regions. i. **The Vedangas and Other Streams of Indian Knowledge System:** The Vedic Corpus: Introduction to Vedas and synopsis of the four Vedas and Sub-classification of Vedas; Messages in Vedas; Introduction to Vedāṅgas : Siksha, Vyakarana, Chandas, Nirukta, Jyotisha and Kalpa ; Vedic Life: Distinctive Features. Other streams of Indian Knowledge System such as Ayurveda, Sthapatya, Natyasastra, Dharmasastra, Arthasastra, etc. The Indian way of continuing the evolution of knowledge through commentaries, interpretations and revisions of the foundational texts. The large corpus of literature in Indian languages. **ii. Indian Language Sciences:** Language Sciences and the preservation of the Vedic corpus. Varnamala of Indian languages based on classification of sounds on the basis of their origin and effort involved. The special feature of the scripts of most Indian languages, that each symbol is associated with a unique sound. Word formation in Sanskrit and Indian languages. Major insights in the Science of Vyakarana as established by Panini. Important	**24**

		texts of Indian Language Sciences —Siksha or phonetics, Nirukta or etymology, Vyakarana or Grammar, Chandas or Prosody. Navyanyaya and Navya-vyakarana in Navadvipa, Varanasi and West and South India. iii. **Indian Mathematics:** Numbers, fractions and geometry in the Vedas. Decimal nomenclature of numbers in the Vedas. Zero and Infinity. Simple constructions from Sulba-sutras. The development of the decimal place value system which resulted in a simplification of all arithmetical operations. Linguistic representation of numbers. Important texts of Indian mathematics. Brief introduction to the development of algebra, trigonometry and calculus. How Indian mathematics continued to flourish in the 18/19/20th centuries. Kerala School. Ramanujan.	
III	**Indian Astronomy**	Ancient records of the observation of the motion of celestial bodies in the Vedic corpus. Sun, Moon, Nakshatra & Graha. Astronomy as the science of determination of time, place and direction by observing the motion of the celestial bodies. The motion of the Sun and Moon. Motion of equinoxes and solstices. Elements of Indian calendar systems as followed in different regions of India. Important texts of Indian Astronomy. Basic ideas of the planetary model of Aryabhata and its revision by Nilakantha. Astronomical instruments. How Indian astronomy continued to flourish in the 18/19th centuries. Astronomical endeavours of Jaisingh, Sankaravarman, Chandrasekhara Samanta.	**15**
IV	**Indian Health Sciences**	Vedic foundations of Ayurveda. Ayurveda is concerned both with maintenance of good health and treatment of diseases. Basic concepts of Ayurveda. The three Gunas and Three Doshas, Pancha-mahabhuta and Sapta-dhatu. The importance of Agni (digestion). Six Rasas and their relation to Doshas. Ayurvedic view of the cause of diseases. Dinacharya or daily regimen for the maintenance of good health. Ritucharya or seasonal regimen. Important Texts of Ayurveda. Selected extracts from Astāngahrdaya (selections from Sūtrasthāna) and Suśruta-Samhitā (sections on plastic surgery, cataract surgery and anal fistula). The large pharmacopeia of Ayurveda. Charaka and Sushruta on the qualities of a Vaidya. The whole world is a teacher of the good Vaidya. Charaka's description of a hospital. Hospitals in ancient and medieval India. How Ayurveda continued to flourish till 18/19th centuries. Surgical practices, inoculation. Current revival of Ayurveda and Yoga.	**17**
		Total	**66**

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3 * 22 NCH = 66 NCH	-	8 * 3 NCH = 24 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

Textbooks

1. *Samskrta Śāstrom ka Itihās*, Baladev Upadhyaya, 2010, Chowkhambha, Varanasi
2. *A Concise History of Science in India*, D. M. Bose, S. N. Sen and B. V. Subbarayappa, Eds., 22nd Edition, 2010, Universities Press, Hyderabad

Reference Books:

1. Dharampal, *Some Aspects of Earlier Indian Society and Polity and Their Relevance Today, 1987,* New Quest Publications, Pune
2. Dharampal, *Indian Science and Technology in the Eighteenth Century: Some Contemporary European Accounts*, 2021, Dharampal Classics Series, Rashtrotthana Sahitya, Bengaluru
3. Dharampal, *The Beautiful Tree: Indian Indigenous Education in the Eighteenth Century*, 2021, Dharampal Classics Series, Rashtrotthana Sahitya, Bengaluru.
4. J. K. Bajaj and M. D. Srinivas, *Timeless India Resurgent India*, 2001, Centre for Policy Studies, Chennai.

- **Detailed Syllabus of Ability Enhancement Course: 1+1 = 2 credits**

Paper V/Subject Name: Introduction to Effective Communication **Subject Code: CEN982A101**

Course Type: AEC **Course Level: 100**

L-T-P-C – 1-0-0-1 **Credit Units: 01** **Scheme of Evaluation: TP**

Objectives:

The objectives of this course are to make the students understand the four major aspects of communication by closely examining the processes and figuring the most effective ways to communicate with interactive activities.

Prerequisites: None

Course Outcomes:

On successful completion of the course the students will be able to

SI No	Course Outcome	Blooms Taxonomy Level
CO 1	**Identify** the elements and processes that make for successful communication and **recognise** everyday activities that deserve closer attention in order to improve communication skills	**BT 1**
CO 2	**Contrast** situations that create barriers to effective communication and **relate** them to methods that are consciously devised to overcome such hindrance	**BT 2**
CO 3	**Use** language, gestures, and para-language effectively to avoid miscommunication and **articulate** one's thoughts and build arguments more effectively	**BT 3**

Detailed Syllabus		
Units	**Course Contents**	**Periods**

I	Introduction to Effective Communication
	• Listening Skills
	  o The Art of Listening
	  o Factors that affect Listening
	  o Characteristics of Effective Listening
	• Guidelines for improving Listening skills
		5

II	• Speaking Skills
	  o The Art of Speaking
	  o Styles of Speaking
	  o Guidelines for improving Speaking skills
	  o Oral Communication: importance, guidelines, and barriers
		5

III	• Reading Skills
	  o The Art of Reading
	  o Styles of Reading: skimming, surveying, scanning
	• Guidelines for developing Reading skills
		5

IV	• Writing Skills	**5**

	○ The Art of Writing ○ Purpose and Clarity in Writing ○ Principles of Effective Writing	

Credit Distribution		
Lecture/Tutorial	**Practicum**	**Experiential Learning**
20 hours	-	10 hours - Movie/ Documentary screening - Peer teaching - Seminars - Field Visit

Textbooks:

1. *Business Communication: Essential Starategies for 21st Century Managers,* Shalini Verma, 2nd Edition, 2014, Vikas Publisher

References Books:

1. P.D. Chaturvedi and Mukesh Chaturvedi, *Business Communication,* 4th Edition, 2017, Pearson Education
2. Meenakshi Raman and Sangeeta Sharma, *Technical Communication: Principles and Practice*, 3rd Edition, 2015, Oxford University Press

Course Type: AEC **Course Level: 100**

L-T-P-C – 1-0-0-1 **Credit Units: 01** **Scheme of Evaluation: TP**

Objectives:

The objective of the course is to increase one's ability to draw conclusions and develop inferences about attitudes and behaviour, when confronted with different situations that are common in modern organizations.

Prerequisites: None

On successful completion of the course the students will be able to

SI No	Course Outcome	Blooms Taxonomy Level
CO 1	**Understand** self & process of self-exploration	BT 2
CO 2	**Learn** about strategies for development of healthy self-esteem.	BT 2
CO 3	**Apply** the concepts to build emotional competencies.	BT 3

Detailed Syllabus:

Modules	Course Contents	Periods
I	**Introduction to Behavioral Science** Definition and need of Behavioral Science,Self: Definition components, Importance of knowing self, Identity Crisis,Gender and Identity, Peer Pressure, Self image: SelfEsteem, Johari Window ,Erikson's model.	4
II	**Foundations of individual behavior** Personality- structure, determinants, types of personalities.Perception: Attribution, Errors in perception. Learning- Theories of learning: Classical, Operant and Social	4
III	**Behaviour and communication.** Defining Communication, types of communication, barriers to communication, ways to overcome barriers to Communication, Importance of Non-Verbal Communication/Kinesics, Understanding Kinesics, Relation between behaviour and communication.	4
IV	**Time and Stress Management** Time management: Introduction-the 80:20, sense of time management, Secrets oftime management, Effective scheduling. Stress management: effects of stress, kinds of stress-sources of stress, CopingMechanisms. Relation between Time and Stress.	4
Total		16

Credit Distribution		
Lecture/Tutorial	**Practicum**	**Experiential Learning**
20 hours	-	10 hours - Movie/ Documentary screening - Peer teaching - Seminars - Field Visit

Text books:

1. *Theories and Models in Applied Behavioural Science*, 1991, J William Pfeiffer (ed.), Management; Pfeiffer & Company

Reference Books:

1. Blair J. Kolasa, *Introduction to Behavioural Science for Business*, 1 9 6 9 , John Wiley & SonsInc
2. K.Alex, *Soft skills*; 2014, S. Chand.

Paper IX/Subject Name: Basket Course 1	**Subject Code: VAC-1**
Course Type: VAC	**Course Level: 100**

L-T-P-C – 2-0-2-3	Credit Units: 03	Scheme of Evaluation: TP

Detailed Syllabus of Value Addition Course (VAC-I)
***** These subjects are Basket Courses that needs to be opted from other Departments/ Schools by the students of RSIT**

List of Value-Added Course For UG 1st Semester

Area	Sl no	Course Title	Course Availability
Knowing India	1	India: Land of Diversity	Open for all
	2	Understanding Sankardeva	Open for all
	3	Introduction to Indian Art: AnAppreciation	Not available for Bachelor ofFine Arts
	4	Gandhian Studies	Open for all
	5	Innovation and Startup Ecosystem of India	Open for all
	6	History of India: Ancient to Modern	Not available for BA History
	7	Indian Architectural Heritage	Open for all
	8	Film and Society: An Indian Perspective	Not available for BA JMC
	9	Cultural Heritage Tourism of India	Not available for BA/BScTTM
Healthcare / Yoga	10	Community Health and Social Work	Not available for BSW
	11	Nutrition and Dietetics for Optimal Health	Not available for BSc Nutrition and Deictics
	12	Sports Psychology	Not available for BA Psychology/Applied Psychology
	13	Comprehensive Healthcare	Open for all
	14	Yoga	Open for all
	15	Stress Management	Open for all
Environment Science andEducation	16	Climate Writing	Open for all
	17	Climate Change	Open for all
	18	Chemistry of the Environment	Open for all
	19	Renewable Energy and SustainableTechnology	Open for all
	20	Eco tourism	Not available for BA/BScTTM
	21	Disaster Management	Open for all
Digital Technology	22	Office Automation	Not available for RSIT
	23	Introduction to Graphic Design	Not available for B.Des (Communication Design, Product Design and GraphicDesign)

SYLLABUS (2nd SEMESTER)

Paper I/Subject Name: Data Structures	**Subject Code: CAP052M201**
Course Type: Major	**Course Level: 100**

L-T-P-C – 2-0-2-3 **Credit Units: 03** **Scheme of Evaluation: TP**

Objective:

The objectives of the course are to expose the students with the concepts of algorithm design and various types of data structures.

Prerequisites: Basics of C Programming

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** various data structures used in programming.	**BT 1**
CO 2	**Understand** the basic constructs of data structure and its implementation.	**BT 2**
CO 3	**Utilise** the appropriate data structures to solve a given problem.	**BT 3**
CO 4	**Analyse** and **evaluate** the data structures used for problem solving	**BT 4 & 5**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Linear Data Structure- I**	Introduction: Why we need data structure? Concepts of data structures: Data and data structure, Abstract Data Type and Data Type. Algorithms and programs, basic idea of pseudo-code. Algorithm efficiency and analysis, time and space analysis of algorithms – order notations. Array: Different representations – row major, column major. Sparse matrix - its implementation and usage. Array representation of polynomials. Linked List: Singly linked list, circular linked list, doubly linked list, linked list representation of polynomial and applications.	**11**
II	**Linear Data Structure- II**	**Stack and Queue:** Stack and its implementations (using array, using linked list applications. Queue, circular queue, dequeuers. Implementation of queue- both linear and circular (using array, using linked list), applications. **Recursion:** Principles of recursion – use of stack, differences between recursion and iteration, tail recursion. Applications - The Tower of Hanoi, Eight Queens Puzzle.	**11**
III	**Nonlinear Data Structures**	**Trees:** Basic terminologies, forest, tree representation (using array, using linked list). Binary trees - binary tree traversal (pre-, in-, post-order), threaded binary tree (left, right, full) - non-recursive traversal algorithms using threaded binary tree, expression tree. Binary search tree- operations (creation, insertion, deletion, searching). Height balanced binary tree – AVL tree (insertion, deletion with examples only). **Graphs:** Graph definitions and concepts (directed/undirected graph, weighted/un-weighted edges, sub-graph, degree, cut-vertex/articulation point, pendant node, clique, complete graph,	**11**

		connected components – strongly connected component, weakly connected component, path, shortest path, isomorphism). Graph representations/storage implementations – adjacency matrix, adjacency list, adjacency multi-list. Graph traversal and connectivity – Depth-first search (DFS), Breadth-first search (BFS) – concepts of edges used in DFS and BFS (tree-edge, back-edge, cross-edge, and forward-edge), applications. Minimal spanning tree – Prim's algorithm (basic idea of greedy methods). B-Trees operation	
IV	**Searching, Sorting**	**a. Sorting Algorithms:** Bubble sort and its optimizations, insertion sort, shell sort, selection sort, merge sort, quick sort, heap sort (concept of max heap, application – priority queue), radix sort. **b**. **Searching Algorithms:** Sequential search, binary search, interpolation search.	**11**
Total			**44**

Subject Name: Data Structures using C++ Lab Syllabus

Total Lab Hours for the semester = 30 (2 hours per week)

Minimum 20 Laboratory experiments based on the following-

1. Some common programs of C as revision.
2. Programs on Arrays- Traversal, Insertion, Deletion, Polynomial Representation, etc.
3. Programs on Linked List- Creation Insertion, Deletion, Polynomial Representation, etc.
4. Programs on Stacks-Creation, Push Pop, Infix to Postfix Conversion, Evaluation.
5. Programs on Queues-Creation, Insertion, Deletion, etc.
6. Programs on Trees- Binary Tree Creation, Tree Traversal, BST
7. Programs on Searching- Linear Search, Binary Search
8. Programs on Sorting- Bubble Sort, Insertion Sort, Selection Sort, Quick Sort, Merge Sort, Heap Sort.

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
2 * 22 NCH = 44 NCH	2 * 15 NCH = 30 NCH	8 * 2 NCH = 16 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

Text Book:

1. *Data structures, Algorithms and Applications in C++*, S.Sahni, 2nd Edition, 2004, University Press (India) Pvt. Ltd.
2. *Data structures and Algorithms in C++,* Michael T.Goodrich, R.Tamassia and .Mount, 2nd Edition, 2011, John Wiley and Sons

Reference Books:

1. Seymour Lipschutz, *Data Structures*, 1st Edition (reprint) 2017, McGraw Hill Education.
2. Yashavant P. Kanetkar, *Data Structure through C++*, 2nd Edition, 2003, BPB Publications.

Paper II/Subject Name: Computer Architecture		**Subject Code: CAP052M202**
Course Type: Major		**Course Level:100**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Objective:

The objectives of the course are to make the students understand the machine instruction, basic computer organization and memory hierarchy with pipelining processing.

Prerequisites: Basics of Digital Logic and Computer Design

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** the different hardware and its working in a Computer Systems in architectural level	**BT 1**
CO 2	**Demonstrate** computer architecture concepts related to design of modern processors, memories, and I/O	**BT 2**
CO 3	**Solve** problems related to computer Organization and Architecture	**BT 3**
CO 4	**Analyse** the performance of commercially available computers in architectural level.	**BT 4**

Detailed Syllabus:

Modules	**Topics**	**Course Content**	**Periods**
I	**Introduction to Computer Hardware and Digital Logic**	Introduction to computer hardware- what is computer hardware, History of computing, the digital computer, PC versus workstation. Gates, circuits, and combinational logic- Analog and digital systems, Fundamental gates, applications of gates, Introduction to Digital Works, introduction to Boolean algebra, Special-purpose logic elements, Programmable logic, Sequential logic, Combinational Circuits..	**15**
II	**Machine Instruction**	Instruction Set Architecture, Assembly language Programming, Addressing modes, Instruction cycle, Registers and storage, RISC versus CISC architecture, Inside CPU.	**15**
III	**Computer Arithmetic & Information Representation**	Bits, bytes, words, and characters, Number bases, Number base conversion, Special-purpose codes, Error-detecting codes, Data-compressing codes, Binary arithmetic- half-adder, full-adder, addition of words, Signed numbers- Sign and magnitude representation, Complementary arithmetic, Two's complement representation, One's complement representation, Floating point numbers- Representation, Normalization, Floating point arithmetic, Multiplication and division.	**15**
IV	**CPU, Buses, Peripherals and Memory**	Input-Output device such as Disk, CD-ROM, Printer etc., Interfacing with IO device, Keyboard & Display Interface. Buses and input/output mechanisms- The bus, I/O fundamentals, Direct Memory Access, Parallel and serial interfaces. Computer memory- Static and Dynamic memory, Random and Serial Access Memories, Memory hierarchy, Memory technology, Cache memory	**15**
Total			**60**

Credit Distribution

Lecture/ Tutorial	Practicum	Experiential Learning
3*20 NCH = 60 NCH	-	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

Text Book:

1. *Computer System and Architecture*, Moris Mano, 3rd Edition, 2007, PHI.
2. *Structured Computer Organization,* A. S. Tanenbaum, 5th Edition, 2009, Prentice Hall of India

Reference Books:

1. V. C. Hamacher, Z. G. Vranesic and S. G. Zaky, *Computer Organization,* 5th Edition, 2002 McGraw Hill.
2. J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach,* 4/e, 2006, Morgan Kaufmann.
3. D. V. Hall, *Microprocessors and Interfacing,* 2nd Edition, 2006, McGraw Hall.

- **Detailed Syllabus of Minor Course**

Paper III/Subject Name: Server-Side Programming		**Subject Code: CAP052N201**
Course Type: Minor		**Course Level: 100**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Objective:

The objectives of the course are to teach students the process to build web applications using the Ruby on Rails framework.

Prerequisites: None

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** building blocks of web application in Rubby on Rails framework	**BT 1**
CO 2	**Understand** the process of building web applications using Rails database	**BT 2**
CO 3	**Build** simple sever side web applications.	**BT 3**
CO 4	**Compare** and **criticise** the design of web applications.	**BT 4 & 5**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Introduction**	Using of GitHub, collaboration of code with others using the git tool, an introduction to Ruby, set up a developer environment and VSCode for Ruby and use irb, basics of Ruby programming language, the use of Ruby hashes and how to write recursive methods, POSIX command line and best practices of git	**15**
II	**Object oriented Programming and Database**	Introduction to object-oriented programming, define classes and also understand the difference between two types of relationships between classes - Composition and Inheritance Introduction to databases and set up a PostgreSQL database connect to a database from a Ruby application Active Record models to manipulate data. RubyGems development of Rails application and connection to the PostgreSQL database	**15**
III	**HTML, CSS & ERB Pipeline**	Basics of the CRUD pattern, designing their HTML pages with CSS and experimenting with using classes, selectors and layouts, basics of the MVC pattern, render dynamic data inside their HTML pages using ERB templates,	**15**
IV	**HTML forms and Rails form helpers & User Authentication**	Accept user input on their application via form element in HTML and also using Rails form helper creation of resources using forms, and learn about Cross Site Request Forgery (CSRF), authenticity tokens, ActiveRecord association, migration and validation password storage and play around with browser cookies, sessions, user authentication	**15**
Total			**60**

Credit Distribution

Lecture/ Tutorial	Practicum	Experiential Learning
3*20 NCH = 60 NCH	-	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

Text Book:

1. *Learn Rails 6: Accelerated Web Development with Ruby on Rails*, Adam Notodikromo ,1st Edition, 2020, Apress

Reference Books:

1. Joseph Joyner, *Ruby on Rails For Beginners: Rails Web Development Programming and Coding Tutorial,* 2nd Edition, 28 September 2015, Minhails Konoplovs

- **Detailed Syllabus of Skill Enhancement Courses (SEC-II)**

Paper VI/Subject Name: Computer Hardware and Networking	**Subject Code: CAP052S201**
Course Type: SEC	**Course Level: 100**

L-T-P-C – 3-0-0-3 **Credit Units: 03** **Scheme of Evaluation: T**

Objective:

The objectives of the course are to explain the different hardware components of a computer system and learn its assembling and disassembling along with various networking devices.

Prerequisites: None

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** different types of Computers and Networking hardware and software	**BT 1**
CO 2	**Understand** basic idea of installation process and components of a PC	**BT 2**
CO 3	**Experiment** with some hardware components to assemble a computer system or network system.	**BT 3**
CO 4	**Analyse** and **evaluate** different computer systems, network and server for their use.	**BT 4 & 5**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Basics of Computer**	Basic Introduction About (Hardware & Software) OS installation (Windows & Linux) , Operating Systems (Edition, Requirement, Types of Installation, Driver Installation) Dual OS installation, Dos Command, Backup and Restore, User Control, Control Panel Computer Peripherals – (Input and Output Devices, Primary Component, Computers Language, Serial and parallel Communication, SMPS) Motherboard (Type, form factor, BUS, IRQ, Chipset, I/O Ports and Connectors, CMOS) Memory (Type, Memory Modules, Development of RAM Comparison of RAM, Virtual Memory, BIOS, POST) Hard Disc (Type, Port, File systems, jumper Setting, Disk Type, Components) Processor (Type, ZIF and SEC, Supports, Virtual Support, Cache, Cores information) External Drives (CD and DVD Drive, Blue-ray, Floppy Disk, Modem, and Printers) Ubuntu (H/W Requirement, basic Command, installation)Assembling and Dis-assembling components, Component upgrade Troubleshooting	**15**
II	**Network Devices and IP Addressing**	Introduction (Types of Networks, Topology, protocols, and Ports) Networking devices (Routers, Switches, Hub, Repeater, NIC Cards, Bridge) Networking Media (Wire, Wireless, Cables, Crimping, UTP) Networking Layers (OSI, TCP/IP) IP Addresses (Version, Classes, Types) Subnetting (VLSM, FLSM, CIDR, Super netting) Setting IP addresses, Sharing files and folders. Network troubleshooting with PING test, ipconfig etc.	**15**
III	**Basics of Networking**	Routing, Routing Components, Ports, Network Simulators (CISCO Packet Tracer), Networking Functions (Static, and Default Routing, Password) Dynamic Routing 1 (RIP1, RIP2, Troubleshooting Commands) Dynamic Routing 2 (IGRP, EIGRP,	**15**

		and OSPF) Routing Security (Standard, Extended, Named ACL) Switching 1 (Types, Command, Password, VLAN, and Commands) Switching 2 (Inter VLAN and Commands, Trunking Protocol, VTP, STP) WAN Security (Static Dynamic, and PAT NAT), Basic Network Troubleshooting	
IV	**Servers**	Basics of configuring NFS, NIS, DNS, FTP, Squid Proxy, DHCP server Mail server, Web server(Apache), File server(Samba), ip tables and firewall	**15**
Total			**60**

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3*20 NCH = 60 NCH	-	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

Text Book:

1. Computer Hardware & Networking With Free CD, A Panel of Authors, 2nd Edition, 2021, Computech Publications

Reference Books:

1. Ajit Mittal and Ajay Rana, *Mastering Pc Hardware And Networking,* 1st Edition, 2014, Khanna Book Publishing.
2. Joginder Singh Saini and Jagdeep Singh Saini, *Royal new pattern computer hardware & network maintenance,* 1st Edition, 2017, Royal Book Dept.

- **Detailed Syllabus of Indian Knowledge System Course**

Paper IV/Subject Name: Introduction to Indian Knowledge System-II		**Subject Code: IKS982I201**
Course Type: IKS		**Course Level: 100**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Objectives:
The objective of this course id to present an overall introduction to all the streams of IKS relevant to the UG programme. It would enable students to explore the most fundamental ideas that have shaped Indian Knowledge Traditions over the centuries.

Prerequisite: Basic knowledge of IKS-I

Course Outcomes:
On completion of this course students will be expected to –

CO	Contents	BT Level
CO_1	**Recall** about classical literature in Sanskrit and other languages	BT 1
CO_2	**Recall** traditional Indian knowledge system and Indian education	BT 1
CO_3	**Summarize the Indian** Art, Architecture, Agriculture, Polity and Economy	BT 2

Detailed Syllabus:

Module	Course Contents	Periods
I	**Classical Literature in Sanskrit and Other Indian Languages**: The nature and purpose of Kavya. Drisya and Sravya Kavyas. The ideas of Indian aestheticians on what constitutes the soul of Kavya. Important examples of classical literature in Sanskrit and other Indian languages **Indian Education**: Preservation of culture, tradition and Dharma through education. Svadhyaya, Pravachana. Also continuity of the family and the vamsha, who are the carriers of knowledge, tradition and Dharma. The extent, inclusiveness and the sophistication of indigenous education in early19th century India. **The Purpose of Knowledge in India**: Para Vidya and Apara Vidya. The corpus connected with Para Vidya. Learning and formalization of concepts associated with Para Vidya also form part of Apara. Apara Vidya. Nature and purpose of sciences, technologies, and all human knowledge concerning the world and the society. The concept of Rita, Dharma. The cycle of mutual dependence of humans and all aspect of creation. Yajna and the inviolable discipline of sharing and caring.	14
II	**Methodology of Indian Knowledge System**: Systematization of knowledge fields as Sastra. Each Sastra has a clearly defined purpose in Vyavahara. The means of valid knowledge (Pramanas). Perception (Pratyaksha), Inference (Anumana) and Textual Tradition (Agama), as discussed in the canonical texts of all the disciplines. The importance of Pratyaksha and Agama in relation to Anumana. **Indian Architecture and Town Planning**: The importance of Sthapatya-veda. The ancient cities of the Indus Saraswati region. Town planning and drainage systems. Examples of the significance of architecture and materials in Ramayana and Mahabharata. Public opulence and private austerity in Indian architecture. Why there are many more of Temples than Palaces. Important texts of Architecture and Sculpture. The prevalence of high Indian architecture in almost all parts of India except the Ganga plains. Examples of high Indian architecture from ancient and medieval periods from different parts of India. The building of Jaipur in the 18th century. How temple art and architecture continue to flourish in modern India. **Indian Fine Arts:** The importance of Gandharva-veda. Natyasastra on the nature and purpose of fine arts. Basic concepts of Indian music and dance. Important texts of Indian music, dance and painting. Indian musical instruments. Different schools of music, dance and painting in different regions of India. Important examples of Indian painting in various part of India. Musicology as a science. Harmonising Lakshya and Lakshana (practise and theory). Major developments in the science and practice of music the 17/18/19th centuries. The current revival of music and dance in India.	22
III	**Indian Agriculture**: The significance of agriculture and irrigation as emphasised in the Ramayana,	15

	Mahabharata and other texts. Mention of Indian agriculture by the Greek historians and later travellers. Significance of agriculture and irrigation for the kings of Indian tradition. Major water-bodies of the ancient times. The Ery system of south India. Excellence of Indian agricultural technologies as observed by more recent European observers. Productivity of Indian agriculture in medieval Thanjavur and eighteenth century Allahabad, Chengalpattu, etc. Indian attitude towards agriculture, based on Walker and later reports. **Indian Textiles**: India as the ancient home of cotton and silk fabrics. Weaving formed the most significant part of Indian economy after agriculture. Varieties of textiles and dyes developed in different regions of India. India as a leading exporter of textiles in the world in the 17/18/19th centuries. **Indian Metallurgy:** Vedic references to metals and metal working. Mining and manufacture in India of Zinc, Iron, Copper, Gold, etc., from ancient times. Indian texts which refer to metallurgy. Important specimens of metal workmanship preserved/found in different parts of India. The significance and wide prevalence of ironsmith and other metal workers in the pre-modern era. European observers on the high quality and quantity of Indian iron and steel in the 18/19th centuries.	
IV	**Indian Polity and Economy**: Indian conception of well-organised Polity and flourishing Economy as expounded in the foundational texts. The notion of Bharartavarsha as a Chakravarti-Kshetra and important attributes of Chakravartin. King as the protector of Dharma. King as the strength and support of the weak. King as the protector of Varta. King as the protector of the times. Meaning of Varta: Krishi, Gopalana and Vanijya forming the basis of Varta and the core of economic activity in society. The importance of sharing. Grama as the centre of the polity. **The Outreach of Indian Knowledge System**: The outreach of Indian Knowledge System beyond Indian boundaries forms the ancient times. Outreach to East, Southeast, Central and Southeast Asia of Indian phonetic script, decimal value place system-based arithmetic, algebra, astronomy and calendar, medical pharmacopeia, architecture, methods of making iron and steel, cotton textiles, etc. The transmission of Indian linguistics, knowledge of plants, iron and steel metallurgy, textiles and dyeing, shipbuilding etc., to Europe in 17/18/19th centuries. Current global outreach of Ayurveda, Yoga and Indian Fine Arts.	15
	Total	66

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3 * 22 NCH = 66 NCH	-	8 * 3 NCH = 24 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

Textbooks:

1. *Samskrta Śāstrom ka Itihās,* Baladev Upadhyaya, 2010, Chowkhambha, Varanasi.
2. *A Concise History of Science in India,* D. M. Bose, S. N. Sen and B. V. Subbarayappa, Eds., 2nd Edition, 2010, Universities Press, Hyderabad

Reference Books:

1. Astāngahrdaya, Vol. I, Sūtrasthāna and Śarīrasthāna, Translated by K. R. Srikantha Murthy, Vol. I, Krishnadas Academy, Varanasi, 1991.
2. Dharampal, Some Aspects of Earlier Indian Society and Polity and Their Relevance Today, New Quest Publications, Pune, 1987.
3. Dharampal, Indian Science and Technology in the Eighteenth Century: Some Contemporary European Accounts, Dharampal Classics Series, Rashtrotthana Sahitya, Bengaluru, 2021
4. Dharampal, The Beautiful Tree: Indian Indigenous Education in the Eighteenth Century, Dharampal Classics Series, Rashtrotthana Sahitya, Bengaluru, 2021.
5. J. K. Bajaj and M. D. Srinivas, Indian Economy and Polity in Eighteenth century Chengalpattu, in J. K. Bajaj ed., Indian Economy and Polity, Centre for Policy Studies, Chennai, 1995, pp. 63-84.
6. J. K. Bajaj and M. D. Srinivas, Annam Bahu Kurvita Recollecting the Indian Discipline of Growing and Sharing Food in Plenty, Centre for Policy Studies, Chennai, 1996.
7. J. K. Bajaj and M. D. Srinivas, Timeless India Resurgent India, Centre for Policy Studies, Chennai, 2001.
8. M. D. Srinivas, The methodology of Indian sciences as expounded in the disciplines of Nyāya, Vyākarana, Ganita and Jyotisa, in K. Gopinath and Shailaja D. Sharma (eds.), The Computation Meme: Explorations in Indic Computational Thinking, Indian Institute of Science, Bengaluru, 2022 (in press).

- **Detailed Syllabus of Ability Enhancement Course: 1+1=2 Credits**

Paper V/Subject Name: Approaches to Verbal and Non-Verbal Communication		**Subject Code: CEN982A201**
Course Type: AEC		**Course Level: 100**
L-T-P-C – 1-0-0-1	**Credit Units: 01**	**Scheme of Evaluation: TP**

Course Objectives

To introduce the students to the various forms of technical communication and enhance their knowledge in the application of both verbal and non-verbal skills in communicative processes.

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Identify** the different types of technical communication, their characteristics, their advantages and disadvantages.	**BT 1**
CO 2	**Explain** the barriers to communication and ways to overcome them.	**BT 2**
CO 3	**Discover** the means to enhance conversation skills.	**BT 3**
CO 4	**Determine** the different types of non-verbal communication and their significance.	**BT 4**

Detailed Syllabus

Modules	Topics (if applicable) & Course Contents	Periods
I	**Technical Communication** Communicating about technical or specialized topics, Different forms of technology-enabled communication tools used in organisations Telephone, Teleconferencing, Fax, Email, Instant messaging, Blog, podcast, Videos, videoconferencing, social media	**4**
II	**Communication Barriers** Types of barriers: Semantic, Psychological, Organisational, Cultural, Physical, and Physiological. Methods to overcome barriers to communication.	**4**
III	**Conversation skills/Verbal Communication** Conversation – Types of Conversation, Strategies for Effectiveness, Conversation Practice, Persuasive Functions in Conversation, Telephonic Conversation and Etiquette Dialogue Writing, Conversation Control.	**4**
IV	**Non-verbal Communication** Introduction; Body language- Personal Appearance, Postures, Gestures, Eye Contact, Facial expressions Paralinguistic Features-Rate, Pause, Volume, Pitch/Intonation/ Voice/ modulation Proxemics , Haptics, Artifactics, Chronemics	**4**
	Total	**16**

Texts:

1. Rizvi, M. Ashraf. (2017). *Effective Technical Communication*. McGraw-Hill.
2. Chaturvedi, P. D. and Chaturvedi, Mukesh. (2014). *Business Communication.*Pearson.
3. Raman, Meenakshi and Sharma, Sangeeta. (2011). *Technical Communication: Principles and Practice* (2nd Edition): Oxford University Press.

References:

1. Hair, Dan O., Rubenstein, Hannah and Stewart, Rob. (2015). *A Pocket Guide to Public Speaking.* (5th edition). St. Martin's. ISBN-13:978-1457670404
2. Koneru, Aruna.(2017) *Professional Communication*. New Delhi: Tata McGraw Hill ISBN-13: 978-0070660021
3. Raman, Meenakshi and Singh, Prakash.(2012). *Business Communication* (2nd Edition): Oxford University Press
4. Sengupta, Sailesh.(2011) *Business and Managerial Communication*. New Delhi : PHI Learning Pvt. Ltd.

Paper V/Subject Name: Behavioural Sciences -II		**Subject Code: BHS982A102**
Course Type: AEC		**Course Level: 100**
L-T-P-C – 1-0-0-1	**Credit Units: 01**	**Scheme of Evaluation: T**

Course objectives:

To increase one's ability to draw conclusions and develop inferences about attitudes and behaviour, when confronted with different situations that are common in modern organizations.

Course outcomes: On completion of the course the students will be able to:

CO 1: Develop an elementary level of understanding of culture and its

implications onpersonality of people.
CO2: Understand the concept of leadership spirit and to know its impact on performance ofemployees.
CO3: Understand and apply the concept of Motivation in real life.

Modules	Course Contents	Periods
I	**Culture and Personality** Culture: Definition, Effect, relation with Personality, Cultural Iceberg, Overview of Hofstede's Framework, Discussion of the four dimensions of Hofstede's Framework.	4
II	**Attitudes and Values** Attitude's definition: changing our own attitudes, Process of cognitive dissonance Types of Values, Value conflicts, Merging personal and Organizational values	4
III	**Motivation** Definition of motivation with example, Theories of Motivation (Maslow, McClelland's theory& Theory X and Y)	4
IV	**Leadership** Definition of leadership, Leadership continuum, types of leadership, Importanceof Leadership, New age leaderships: Transformational & transactional Leadership, Leaders as role models.	4
	Total	16

Text books:

1. J William Pfeiffer (ed.) Theories and Models in Applied Behavioural Science, Vol 3, Management, Pfeiffer &Company
2. Blair J. Kolasa, Introduction to Behavioural Science for Business, John Wiley & SonsInc.
3. Organizational Behaviour by Kavita Singh (Vikas publishers, 3rd Edition).

- **Detailed Syllabus of Value-Added Course (VAC-II)**

Paper VII/Subject Name: Basket Course 2		**Subject Code:** VAC992V2409
Course Type: VAC		**Course Level: 100**
L-T-P-C – 2-0-2-3	**Credit Units: 03**	**Scheme of Evaluation: TP**

***** These subjects are Basket Courses that needs to be opted from other Departments/ Schools by the students of RSIT**

SYLLABUS (3rd SEMESTER)

Paper I/Subject Name: JAVA Programming		**Subject Code:** CAP052M301
Course Type: Major		**Course Level:200**
L-T-P-C – 3-0-2-4	**Credit Units: 03**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are to teach the concepts and implementations of object-oriented programming using JAVA language.

Prerequisites: Basics of Procedural or Object-Oriented Programming

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** basic structures, constructs, control structure and syntax of JAVA	**BT 1**
CO 2	**Understand** basic idea of object-oriented programming using JAVA	**BT 2**
CO 3	**Apply** the concepts of Java, multithreading and Exception handling to develop efficient and error free codes.	**BT 3**
CO 4	**Analyse** and **evaluate** JAVA programs for its efficiency	**BT 4 & 5**

Detailed Syllabus:

Modules	Topics	Course content	Periods
I	**Introduction**	A look at procedure-oriented programming, Object-oriented paradigm, Basic concepts of object-oriented programming (OOP) (encapsulation, inheritance, polymorphism), How Java differs from C and C++, Applications of OOP. Overview of JAVA, Use of math functions, comments, Constructing a java program, Introduction of JVM, Command line argument, Data types, Variables: declaration, scope, Type Conversion and Type Casting, Constants, Operators, Evaluation of Expression, Precedence of Operators, Control statements: selection, iteration, and jump, Introduction to Socket Programming.	**15**
II	**Classes and Objects**	Class: definition and example, Declaring objects, Method overloading and overriding, Binding: concept of binding, static vs. dynamic binding, Using this and super keywords, Access Control, Inheritance: Extending a class, Final, Abstract classes, Constructors Arrays: one-dimensional and multi-dimensional, Strings: string processing functions	**15**
III	**Packages, Interfaces, Exception Handling**	Defining a package, accessing a package and using a package, Interfaces: multiple inheritance, Defining interfaces, implementing interfaces and extending interfaces. Exception handling fundamentals, Exception type: using try…catch, Multiple catch clauses, Throw and Throws Creating threads, Extending the thread class, Stopping and blocking a thread, Life cycle of thread, Threads methods, Thread exceptions, , JDBC,	**15**
IV	**Applets and Files**	Introduction: local and remote applets, How to write applets, Building applet code, Applet life cycle, Creating an executable applet I/O basics, concept of streams, Stream classes: byte stream classes, character stream classes, I/O exceptions, Creation of files, Random access files	**15**
Total			**60**

Text Books:

1. *Programming with Java: A Primer*; Balagurusamy E., 3rd Edition, 2005, Tata McGraw-Hill, New Delhi
2. *Thinking in Java*, Eckel B., 4th Edition, 2006, PHI.

Reference Books:

1. Maurice N. et al, *Java Generics and Collections*, 1st Edition, 2006, O'REILLY Publication.
2. Booch G., Rumbaugh J. Jacobson I., *The Unified Modeling Language User Guide*, 2nd Edition, 2005, Pearson Education.
3. Schildt H., *The Complete Reference Java,* 7th Edition, 2007, Tata McGraw-Hill, New Delhi

JAVA Programming Lab

Objective:

The objectives of this course are to make the students understand and analyze practically the utility of JAVA programming language.

Prerequisites: Basics of Procedural or Object-Oriented Programming

Detailed Syllabus:

Total Lab Hours for the semester = 48 (4 hours per week)

Minimum 20 Laboratory experiments based on the following-

- Write a program in java that outputs your name in giant letters.
- Write a program in Java to find the day of the week of a given date.
- Write a program in Java called Grades Statistics, which reads in n grades (of int between 0 and 100, inclusive) and displays the average, minimum, maximum, and standard deviation.
- Write a program in Java to compute execution time by generating random numbers.
- Write a program in Java to implement the following:
    a. Tokenize the paragraph into single word.
    b. Find the number of word in a paragraph?
    c. Find the number of similar words from the input word.
    d. Find the number of occurrence of each word.
- Write a program in Java to implement some inheritance hierarchy.
- Design and implement an address book object that contains a person's name, home address and phone number, business address and phone number, and numbers for their fax machine, cellular phone, and pager. Write a program in Java to this test plan for the object and implement a driver [finally prepare a package].
- Write a program in Java to demonstrate the use of try, catch, finally throw and throws keywords and demonstrate the following points in the program.
    a. Multiple catch blocks.
    b. try-catch-finally combination.
    c. try-finally combination.
    d. Exception propagation among many methods.
    e. Use of getMessage(), printStackTrace() function of Throwable class.
    f. Nested try blocks
- Write a program that does the following.
    a. Prompts the user for an input file name through a dialog box.
    b. Prompts the user for an output file name through a dialog box.
    c. Copies the input file into the output file, subject to the removal of the space characters listed below from each line.
        i. The leading space characters
        ii. The trailing space characters
        iii. The space characters that are preceded by space characters
- Write a program in Java to design forms.
- Write a program in Java to design a student information form to enter data into the database.
- Write a program in Java to connect some form designed with the back-end database and test them by inserting some records.

Text Books:

1. *Programming with Java: A Primer*; Balagurusamy E., 3rd Edition, 2005, Tata McGraw-Hill, New Delhi

2. *Thinking in Java*, Eckel B., 4th Edition, 2006, PHI.

Reference Books:

1. Maurice N. et al, *Java Generics and Collections*, 1st Edition, 2006, O'REILLY Publication.
2. Booch G., Rumbaugh J. Jacobson I., *The Unified Modeling Language User Guide*, 2nd Edition, 2005, Pearson Education.
3. Schildt H., *The Complete Reference Java,* 7th Edition, 2007, Tata McGraw-Hill, New Delhi

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3* 20 NCH = 60 NCH	2 * 15 NCH = 30 NCH	8 * 2 NCH = 16 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

Paper II/Subject Name: Introduction to Database Management Systems		**Subject Code:** CAP052M302
Course Type: Major		**Course Level:200**
L-T-P-C – 3-0-2-4	**Credit Units: 03**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are to make the students learn about databases and the process of designing and constructing data models.

Prerequisites: C/C++, Concepts of Data Structures.

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** various terms used in Database Management system	**BT 1**
CO 2	**Understand** the basic concepts and applications of database systems	**BT 2**
CO3	**Apply** the basic concepts to design Database	**BT 3**
CO 4	**Evaluate** the process Database design, transaction processing and concurrency control	**BT 4**

Detailed Syllabus:

Modules	Topics	Course content	Periods
I	**Introduction**	Introduction to Data System, Drawbacks of Conventional File System, Purpose of database systems, DBMS Components, Architecture, Data Independence, Data modelling, Entity Relationship Model, Relational, Network, Hierarchical and object-oriented models, Data Modelling using the Entity Relationship Model.	**15**
II	**Relational Databases**	Relational databases, relational algebra, relational calculus. Data definition with SQL, insert, delete and update statements in SQL, views, data manipulation with SQL, triggers and assertions, cursors, Embedded SQL	**15**
III	**Normalization**	Relational Database Design guidelines, Integrity Constraints, Domain Constraints, Referential integrity, Functional Dependency, Normalization using Functional Dependencies, Normal forms (1NF, 2NF, 3NF, BCNF), Multi-valued Dependencies and Forth Normal Form, Join Dependencies and Fifth Normal Form, Pitfalls in Relational Database Design, Lossless Non-additive Join Property of Decomposition, Dependency Preserving Decomposition	**15**
IV	**Transaction Processing, Concurrency and Recovery**	Introduction. ACID Properties, Schedules and Recoverability -Serializability of Schedules- Concurrency Control, Database Recovery Concepts- Caching, Checkpoints, Transaction Rollback, Case Study-Recovery Techniques in DBMS	**15**
Total			**60**

Text Book:

1. *Fundamentals of Database System,* Elmasri, Navathe, 7th Edition, 2016, Pearson Education Asia
2. *Database System Concepts*, Korth H.F., Silberschatz A.; 6th Edition, 2013, Mc Graw Hill.
3. *Introduction to Database Management System*, Kahate A., 1st Edition, 2004, Pearson Educations
4. *DataBase Management System*, Paneerselvam, 2nd Edition, 2011, PHI Learning

Reference Books:

1. Date C.J.*, An Introduction to Database Systems,* 8th Edition, 2003, Pearson Education Asia
2. Desai B.C., *An Introduction to Database Systems,* Revised Edition, 2012, Galgotia Publications

Introduction to Database Management Systems Lab

Objective:

The objectives of the course to teach the student database design and query processing through MySql.

Prerequisites: C/C++, Concepts of Data Structures

Detailed Syllabus:

Total Lab Hours for the semester = 48 (4 hours per week)

Minimum 20 Laboratory experiments based on the following-

- Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
- Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET,
- Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
- Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
- Creation of simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
- Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
- Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
- Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.
- Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
- Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
- Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
- Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers

Text Book:

1. *Fundamentals of Database System,* Elmasri, Navathe, 7th Edition, 2016, Pearson Education Asia
2. *Database System Concepts*, Korth H.F., Silberschatz A., 6th Edition, 2013, Mc Graw Hill.

Reference Books:

1. Date C.J.*, An Introduction to Database Systems,* 8th Edition, 2003, Pearson Education Asia
2. Desai B.C., *An Introduction to Database Systems,* Revised Edition, 2012, Galgotia Publications

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**

2 * 22 NCH = 44 NCH	2 * 15 NCH = 30 NCH	8 * 2 NCH = 16 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

- **Detailed Syllabus of Minor Course**

Paper III/Subject Name: Front-End Development with React		**Subject Code:** CAP052N301
Course Type: Minor		**Course Level: 100**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Objective:

The objectives of the course are to teach the students about React & Type Script to enable them to create web pages.

Prerequisites: Fundamentals of Web Development and Server Programming

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** semantics and syntax of Typescripts .	**BT 1**
CO 2	**Understand** static types and know how to port untyped JavaScript	**BT 2**
CO 3	**Apply** the concepts learnt to create Single Page Web Applications (SPA) using React, Typescript and Tailwind CSS.	**BT 3**
CO 4	**Inspect** different elements of front-end development	**BT 4**

Detailed Syllabus:

Modules	Topics	Course content	Periods
I	**React Fundamentals and Sate Management**	Introduction to TypeScript by setting up a development environment, the TypeScript programming language and the React framework, and demonstrates some of the basic concepts that underpin the use of React for building dynamic reactive user interfaces. the Hooks feature of React on the usage of call-back functions and how to use them to build dynamic components that maintain an internal state. Standard hooks and the creation and use of custom hooks. This module also demonstrates state management by building a form and accepting user input.	**09**
II	**Client-side routing**	The concept of client-side routing as a separate behaviour from server-side route management. Demonstration of the various aspects of client-side routing such as the use of path parameters, query parameters, programmatic navigation and the operation of links and URLs that are handled client-side.	**09**
III	**Modelling and managing complex states**	Managing complex states using the state reducer pattern, and then demonstrates the pattern by implementing it using Reacts use Reducer hook. Introduction to APIs to interface client-side code with the server-side, creating model types to allow interaction to take place, maintain a session with the backend, and working with pageable APIs	**09**
IV	**Production React Apps**	Front-end development including the importance of accessibility and WAI-ARIA standards, and use of third-party packages from the NodeJS ecosystem. Production-specific optimizations of a React application, build & deployment process, and configuration of a progressive web app.	**09**
Total			**36**

Textbooks:

1. *Learn React with TypeScript 3: Beginner's guide to modern React web development with TypeScript 3,* Carl Rippon, 2018, Pact Publishing.
2. *Full-Stack React, TypeScript, and Node: Build cloud-ready web applications using React 17 with Hooks and GraphQL* , David Choi, 2020, Packt Publishing Limited.

Reference Books:

1. Frank Zammetti, *Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, Python, Django, and Docker*, 2nd Edition, 2022, APress

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3*20 NCH = 60 NCH	2*15 NCH=30 NCH	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

- **Detailed Syllabus of Interdisciplinary Subject**

Paper IV/Subject Name: Introduction to Python		**Subject Code:** INT052I301
Course Type: IS		**Course Level: 200**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Objective:

The objectives of the course are to teach the students about Programming with Python and use it to solve real world problems.

Prerequisites: Fundamentals of Computers

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** uses systematic, syntax and control structures of Python	**BT 1**
CO 2	**Understand** the basic concepts and terminologies of Python Programming	**BT 2**
CO 3	**Apply** the concepts learnt to write efficient programs	**BT 3**
CO 4	**Analyze** and **evaluate** the codes to fix the errors	**BT 4 & 5**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Introduction to Python**	Introduction to Python Programming: Python interpreter/shell, indentation; identifiers and keywords; literals, numbers, and strings; operators (arithmetic operator, relational operator, Boolean operator, assignment, operator, ternary operator and bitwise operator) and expressions	**09**
II	**Programming With Python**	Input and output statements, defining functions, control statements (conditional statements, loop control statements, break, continue and pass, exit function.), default arguments,	**09**
III	**Python Functions and Strings**	Python Functions , Python Lambda, Python Arrays, Python Classes/Object, Inheritance, Iterator, Polymorphism , Scope, Modules, Dates, Maths, JSON,RegEx, PIP, User Input , Strings	**09**
IV	**Python Modules**	Introduction to Numpy, Pandas,SciPy, Dijango	**09**
Total			**36**

Introduction to Python Programming Lab

Detailed Syllabus:

Total Lab Hours for the semester = 30 (2 hours per week)

Minimum 20 Laboratory experiments based on the following-

1. **Hello World Program:**
    - Write a simple Python program to print "Hello, World!" to the console.
2. **Variable Declaration and Printing:**

- Practice declaring variables of different types (int, float, string) and printing their values.

3. **Basic Arithmetic Operations:**
   - Write Python scripts to perform basic arithmetic operations such as addition, subtraction, multiplication, and division.
4. **Conditional Statements:**
   - Create programs using if-else statements to perform tasks based on certain conditions.
5. **Loops (for and while):**
   - Practice writing for and while loops to iterate over sequences or execute code repeatedly.
6. **Lists and List Operations:**
   - Explore lists in Python and perform operations like appending, removing, and accessing elements.

Level: Intermediate

7. **Functions:**
   - Define and call functions to encapsulate reusable code blocks. Practice passing arguments and returning values from functions.
8. **String Manipulation:**
   - Work on tasks involving string manipulation, such as concatenation, slicing, and searching.
9. **File Handling:**
   - Write Python scripts to read from and write to files. Practice handling exceptions during file operations.
10. **Dictionaries and Sets:**
    - Experiment with dictionaries and sets in Python. Perform operations like adding, removing, and accessing elements in dictionaries and sets.
11. **Object-Oriented Programming (OOP) Concepts:**
    - Introduce students to OOP concepts like classes, objects, inheritance, and polymorphism. Have them implement simple classes and explore inheritance hierarchies.
12. **Exception Handling:**
    - Practice handling exceptions using try-except blocks to gracefully manage errors in Python programs.
13. **Data Structures and Algorithms:**
    - Implement common data structures (e.g., stacks, queues, linked lists) and algorithms (e.g., sorting, searching) using Python.
14. **Regular Expressions:**
    - Introduce regular expressions and their usage in Python for pattern matching and text processing tasks.
15. **Modules and Packages:**
    - Explore the concept of modules and packages in Python. Have students create their own modules and packages and import them into other scripts.
16. **GUI Programming with Tkinter:**
    - Introduce GUI programming using Tkinter. Have students create simple graphical user interfaces (GUIs) for basic applications

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3*20 NCH = 60 NCH	2*15 NCH=30 NCH	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

Textbooks:

1. *Introduction to computation and programming using Python.* Guttag, J.V., 2nd Edition, 2016, MIT Press.

Reference Books:

1. Kamthane, A. N., & Kamthane, A.A. *Programming and Problem Solving with Python*, 2017, McGraw Hill Education.
2. Liang, Y. D., *Introduction to Programming using Python.* 2013, Pearson Education.

- **Detailed Syllabus of Ability Enhancement Courses (AECC-III/IV)**

Paper VI/Subject Name: Career Oriented Communication		**Subject Code: CEN982A301**
L-T-P-C – 1-0-0-1	**Credit Units: 01**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are to prepare students to adopt different communication strategies and meet the competitive market of employment by considering relevant information related to job requirements.

Prerequisites: Basic understanding of the need to groom oneself for employment and the need for preparation of the same.

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Understand** the requirement of the job market.	**BT 2**
CO 2	**Build** oneself for the competitive market of employment with the concepts learnt.	**BT 3**

Detailed Syllabus:

Modules	Topics	Course content	Periods
I	**Perfecting the Art of Speaking**	**Informative Speaking** Types of informative speaking, Informative versus persuasive topics, techniques of informative speaking, Sample informative speech and its Analysis **Persuasive Speaking** Characteristics of Persuasion, categorizing types of persuasion, Creating the Persuasive Message, Adapting to the Audience, Building Credibility as a Speaker, Sample Persuasive Speech and its analysis	**3**
II	**Employment communication for Internship and Campus placement**	Employability versus employability, filling the industry-academia gap, Enhancing Employability : A five step approach (SWOT, JOHARI, Gathering job related information through research, Planning for employment, writing job suitability statement, reaching out to the prospective employer, preparing for the recruitment and selection process)	**3**
III	**Learning the Written Process**	Principles of effective writing Different forms of written communication used in organisations – • Business Letters- parts of business letters, Order, acceptance & cancellation, complaint & adjustment letters. • Project report – format and elements Internal office communication - office order, circular, notice, agenda, minutes.	**3**
IV	**Communication for Employment**	Preparing Resumes, Job Cover letter, Objectives of Interviews, Types of Interviews, Preparing for the Job interview, Different types of questions asked in Job interview, Qualifying the English Language Test (Synonyms And Antonyms, Confusing Words, Idioms and Phrases, Sentence Completion, Spellings, Grammar, Reading Comprehension, Verbal logic)	**3**
Total			**12**

Text Book:

1. *Business Communication: Essential Strategies for twenty-first century Managers,* Verma, Salini. 2nd Edition, 2015, Vikas Publishing House Pvt Ltd. pp 59-86, 119-165, 191-232, 243-259..

Reference Books:

1. Dufrene, Sinha, *BCOM: An Innovative Approach to learning and teaching Business Communication,* Lehman, 2011, Cengage Learning Pvt. Ltd.pp.399-405, 332-355

Paper VII/Subject Name: Behavioral Science-III		Subject Code: BHS982A304
L-T-P-C – 1-0-0-1	Credit Units: 01	Scheme of Evaluation: TP

Objective:

The objectives of the course are:

- **Detailed Syllabus of Skill Enhancement Courses (SEC-III)**

Paper VI/Subject Name: System Administration		**Subject Code:** INT052S301
Course Type: SEC		**Course Level: 200**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Objective:

The objectives of the course are to make the students familiar with system administration and tools and techniques used in it.

Prerequisites: Basics of Operating Systems

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** various system admirations command and tools	**BT 1**
CO 2	**Understand** the need of basic system administration and tools associated with it .	**BT 2**
CO 3	**Apply** system administration command and tools to administer Unix/Linux machines as standalone workstations or in a network	**BT 3**
CO 4	**Analyse** different commands of Linux system	**BT 4**

Detailed Syllabus:

Modules	Topics	Course content	Hours
I	**The System Administrator and User Management**	Overview of system administration roles and responsibilities Introduction to different operating systems (Linux, Windows) Basics of command-line interface (CLI) and shell scripting Boot and Shut Down: Run levels, Processes and daemons, Configure startup scripts. User Management: Add user, User groups, User and system security, Collapse User environment, Shell startup scripts, What not to do in startup scripts, Other dot files	**15**
II	**File Management, Networking and Backup**	File system structure: Manage disk storage, Partition, Format, Fix errors on disk, Mount Links: hard, symbolic, Permission Permission bits, Special permission, ACLs, Quotas. Networking: Network concepts overview, History, ISO/OSI., Layers description, Name to address translation, File sharing with NFS, NIS, Services and inetd. Backup strategy, Selecting the backup devices and software, Automating the backup procedure, Third party product overview, Auto-mounter Requirements and Mechanism	**15**
III	**Backup System Administration Tools**	Monitor processes: truss/strace, ps top.\, Monitor network: lsof, netstat, Working with files: strings, awk, od, du, df, find, Misc: which, whereis, dmesg, Logfiles, Operating System Installation, System installation, Linux/Solaris installation, Patches, Installing and removing packages (RPM), Download compile and install using source code, Kernel reconfig, Get the kernel source code, Add new adapter and update drivers, Kernel upgrade.	**15**
IV	**The proc File system and System Monitoring**	Map of /proc, Process entries, Hardware information, Kernel information, Kernel settings, Swap space tunings, Detecting physical memory shortage, System resource loads: CPU, I/O, Disk, Raid disks, Setting limits to processes, Measuring network load.	**15**
Total			**60**

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3*20 NCH = 60 NCH	-	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

Text Book:

1. *Essential System Administration: Tools and Techniques for Linux and Unix Administration,* Aeleen Frisch, 3rd Edition, 2013, O'Reilly Media

Reference Books:

1. Evi Nemeth, Synder, Hein, Whaley, MAckin, *UNIX and Linux System Administration Handbook,* 5th Edition, 2017, Addison Welsley

SYLLABUS (4th SEMESTER)

Paper I/Subject Name: Operating Systems		**Subject Code:** CAP052M401
Course Type: Major		**Course Level:200**
L-T-P-C – 3-0-2-4	**Credit Units: 03**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are to teach the basic concepts and functions of operating systems and make them understand the principles of concurrency.

Prerequisites: Concepts of Computer Organization and Architecture

Course Outcomes

On successful completion of the course the students will be able to:		
Sl No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define & Name** terminology and processes associated with Operating System	**BT 1**
CO 2	**Understand** the basic concepts of Operating systems.	**BT 2**
CO 3	**Apply** the principles of scheduling, and concurrency to solve various problems related to OS	**BT 3**
CO 4	**Analyze** and **evaluate** different OS systems in terms of process management, memory management and I/O systems.	**BT 4 & 5**

Detailed Syllabus:

Modules	**Topics**	**Course Contents**	**Hours**
I	**Operating Systems Overview**	Introduction and history of Operating systems, structure and operations; processes and files. **Computer System Overvie**w- Basic Elements, Instruction Execution, Interrupts Memory Hierarchy, Cache Memory, Direct Memory Access, Multiprocessor and Multicore Organization. **Operating system overview**-objectives and functions, Evolution of Operating System.- Computer System Organization- Operating System Structure and Operations- System Calls, System Programs, OS Generation and System Boot	**09**
II	**Process Management and Concurrency Control**	**Processes**-Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication; Threads- Overview, Multicore Programming, Multithreading Models; Thread and SMP Management. **Process Synchronization** – Critical Section Problem, Mutex Locks, Semaphores, Monitors; CPU Scheduling and scheduling algorithms. **Deadlocks**- Shared resources, resource allocation and scheduling, resource graph models, deadlock detection, deadlock avoidance, deadlock prevention algorithms	**09**
III	**Storage Management**	Memory Management requirements, Memory partitioning: Fixed and Variable Partitioning, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Fragmentation, Swapping, and Paging. Segmentation, Demand paging, Virtual Memory: Concepts, management of VM, Page Replacement Policies (FIFO, LRU, Optimal, Other Strategies), Thrashing. 32 and 64 bit architecture Examples; Allocating Kernel Memory, OS Examples	**09**
IV	**I/O and File Systems**	I/O Devices, Organization of I/O functions, Operating System Design issues, I/O Buffering, Overview of mass storage structure-disks and tapes. Disk structure – accessing disks, Swap Space. Disk Scheduling (FCFS, SCAN, C-SCAN, SSTF), RAID, Disk Cache. Disk Protection– Goals, Principles, Domain. **File System Interface**: File Concepts – Attributes – operations – types – structure – access methods. File system mounting. Protection. File system implementation. Directory implementation – allocation methods. Free space Management.	**09**
TOTAL			**36**

Text Books:

1. *Operating System Concepts,* Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, 9th Edition, 2012, John Wiley and Sons Inc.

Reference Books:

1. William Stallings, *Operating Systems – Internals and Design Principles*, 7th Edition, 2011, Prentice Hall.
2. Andrew S. Tanenbaum, *Modern Operating Systems*, 2nd Edition, 2001, Addison Wesley.
3. D M Dhamdhere, *Operating Systems: A Concept-Based Approach*, 2nd Edition, 2007, Tata McGraw-Hill Education.

Operating Systems Lab

Objective:

The objectives of the course are to make the students learn about process and disc scheduling practically along with the working of system calls.

Prerequisites: Fundamentals of Computer Programming

Detailed Syllabus:

Total Lab Hours for the semester = 48 (4 hours per week)

Minimum 20 Laboratory experiments based on the following-

1. Basic Linux Commands and Overview.
2. Write Shell Script for followings.
    - To find the global complete path for any file.
    - To broadcast a message to a specified user or a group of users logged on any terminal.
    - To copy the file system from two directories to a new directory in such a way that only the latest file is copied in case there are common files in both the directories.
    - To compare identically named files in two different directories and if they are same, copy one of them in a third directory.
    - To delete zero sized files from a given directory (and all its sub- directories).
    - To display the name of those files (in the given directory) which are having multiple links.
    - To display the name of all executable files in the given directory.
    - Write a script to display the date, time and a welcome message (like Good Morning etc.). The time should be displayed with "a.m." or "p.m." and not in 24 hours notation.
    - Write a script to display the directory in the descending order of the size of each file.
3. Implementation of FCFS (First Come First Serve) CPU Scheduling.
4. Implementation of SJF (Shortest Job First) CPU Scheduling.
5. Implementation of Round Robin (RR) CPU Scheduling.
6. Implementation of Priority CPU Scheduling Algorithm.
7. Implementation of FIFO Replacement Algorithm.
8. Implementation of Optimal Page Replacement Algorithm.
9. Implementation of LRU Page Replacement Algorithm by Stack method
10. Implement the producer-consumer problem using threads.

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**

2 * 20 NCH = 60 NCH	2 * 15 NCH = 30 NCH	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

Text Books:

1. *Operating System Concepts,* Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, 9th Edition, 2012, John Wiley and Sons Inc.

Reference Books:

1. William Stallings, *Operating Systems – Internals and Design Principles*, 7th Edition, 2011, Prentice Hall.
2. Andrew S. Tanenbaum, *Modern Operating Systems*, 3rd Edition, 2009, Addison Wesley.
3. D M Dhamdhere, *Operating Systems: A Concept-Based Approach*, 2nd Edition, 2007, Tata McGraw-Hill Education.

Paper II/Subject Name: Data Communications and Networks		**Subject Code:** INT052M402
Course Type: Major		**Course Level:200**
L-T-P-C – 3-0-2-4	**Credit Units: 03**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are to make the students understand the significance and concepts of computer networks along with the layered architecture.

Prerequisites: Basics of internet technologies and graph theory

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** various networking model and their workings	**BT 1**
CO 2	**Understand** the significance and concepts of computer networks.	**BT 2**
CO 3	**Apply** networking concepts to solve problems and develop networks.	**BT 3**
CO 4	**Analyse** and **evaluate** basic protocols and design issues for layered model.	**BT 4 & 5**

Detailed Syllabus:

Modules	**Topics**	**Course Contents**	**Hours**
I	**Data Link Layer and Medium Access Sub-layer**	Design issues, Framing, Error detection and correction codes: checksum, CRC, hamming code, Data link protocols for noisy and noiseless channels, Sliding Window Protocols: Stop & Wait ARQ, Go-back-N ARQ, Selective repeat ARQ, Data link protocols: HDLC and PPP Static and dynamic channel allocation, Random Access: ALOHA, CSMA protocols, Controlled Access: Polling, Token Passing, IEEE 802.3 frame format, Ethernet cabling, Manchester encoding, collision detection in 802.3, Binary exponential back off algorithm	**15**
II	**Network Layer**	Design issues, IPv4 classful and classless addressing, subnetting, Routing algorithms: distance vector and link state routing, Congestion control: Principles of Congestion Control, Congestion prevention policies, Leaky bucket and token bucket algorithms	**15**
III	**Transport Layer**	Elements of transport protocols: addressing, connection establishment and release, flow control and buffering, multiplexing and de-multiplexing, crash recovery, introduction to TCP/UDP protocols and their comparison	**15**
IV	**Application Layer**	World Wide Web (WWW), Domain Name System (DNS), E-mail, File Transfer Protocol (FTP), SMTP, HTTP, Introduction to Network security	**15**
	TOTAL		**60**

Text Books:

1. *Data and Computer Communication*, William Stallings, 10th Edition, 2013, PHI.
2. *Data Communications and Networking*, Behrouz A Forouzan, 4th Edition, 2017, Tata McGraw Hill
3. *Computer Networks*, Tannenbaum, 3rd Edition, 1996, Pearson Education.

Reference Books:

1. L.L. Peterson & B.S. Davie, *Computer Networks: A Systems Approach*, 5th Edition, 2011, Morgan Kaufmann
2. Anuranjan Misra, *Computer Networks*, 2006, Acme Learning, Morgan Kaufman Publication, New Delhi
3. BhushanTrivedi, *Computer Networks*, Reprint Edition, 2011, Oxford press

Data Communication and Networks Lab

Objective:

The objectives of the course are to make the students learn socket programming and to make them familiar with simulation tools.

Prerequisites: Fundamentals of Computer Programming and Data Communication

Detailed Syllabus:

Total Lab Hours for the semester = 48 (4 hours per week)

Minimum 20 Laboratory experiments based on the following-

- To study various topologies for establishing computer networks.
- To learn the usage of various basic tools (crimping, krone etc.) used in establishing a LAN.
- To familiarize with switch and hub used in networks.
- To learn the usage of connecters and cables (cabling standards) used in networks
- To make certain copper and fiber patch cords using different standards.
- To familiarize with routers & bridges
- Use commands like ping, ipconfig for trouble shooting network related problems.
- NIC Installation & Configuration (Windows/Linux)
- TCP/UDP Socket Programming
- Multicast & Broadcast Sockets
- Develop a program to compute the Hamming Distance between any two code words.
- Develop a program to compute checksum for an _m' bit frame using a generator polynomial.
- IPC (Message queue)
- Implementation of a Prototype Multithreaded Server
    - Implementation of o Data Link Layer Flow Control Mechanism (Stop & Wait, Sliding Window)
    - Data Link Layer Error Detection Mechanism (Cyclic Redundancy Check)
    - Data Link Layer Error Control Mechanism (Selective Repeat, Go Back N)

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3 * 20 NCH = 30 NCH	2 * 15 NCH = 30 NCH	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

Text Books:

1. *Data and Computer Communication*, William Stallings, 10th Edition, 2013, PHI.
2. *Data Communications and Networking*, Behrouz A Forouzan, 4th Edition, 2017, Tata McGraw Hill
3. *Computer Networks*, Tannenbaum, 3rd Edition, 1996, Pearson Education.

Reference Books:

1. L.L. Peterson & B.S. Davie, *Computer Networks: A Systems Approach*, 5thEdition, 2011, Morgan Kaufmann
2. Anuranjan Misra, *Computer Networks*, 2006, Acme Learning, Morgan Kaufman Publication, New Delhi
3. Bhushan Trivedi, *Computer Networks*, Reprint Edition, 2011, Oxford press

Paper III/Subject Name: Indian Mathematics in Computer Science		**Subject Code:** INT052M403
Course Type: Major		**Course Level:200**
L-T-P-C – 3-1-0-4	**Credit Units: 04**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are to provide a comprehensive knowledge on Vaidic mathematics and uses of IKS in Computer Science

Prerequisites: Concepts of Computer Programming and Basic Mathematics

Course Outcomes

On successful completion of the course the students will be able to:		
Sl No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	To help students to **Define the** significance Indian mathematic and astronomical knowledge	**BT 1**
CO 2	**Demonstrate** understanding in Indian mathematic and astronomical knowledge to solve problems.	**BT 2**
CO 3	**Apply** Indian Mathematics to solve problems	**BT 3**
CO 4	**Examine** the use of **ancient** knowledge to computer science discipline.	**BT 4**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Introduction to Ancient Mathematics & Astronomy**	Brief introduction to inception of Mathematics & Astronomy from vedic periods. Details of different authors who has given mathematical & astronomical sutra (e.g. arytabhatta, bhaskara, brahmagupta, varamahira, budhyana, yajanvlkya, panini, pingala.	**15**
II	**Ancient Mathematics –I**	Veda & Sulvasutras (Pythagoras theorem, Square root & Squaring Circle) (baudhayana sulbhasutra, apastamba sulbhasutra, katyayana sulbhasutra, manava sulbhasutra, maitrayana sulbhasutra, varaha sulbhasutra, vadhula sulbhasutra Pingala's chandasutras, sunya, yaat-tavat, Aryabhata (Aryabhatiya, Asanna, ardha-jya, kuttaka,) bhaskara (trigonometory,shridhara, mahavira), Bhaskara Acharya (Sidhantashiromani), Varamahira panchasiddhantika.	**15**
III	**Ancient Mathematics – II**	Brahamgupta (vargaprakrati, bhramasphuta siddhanta, bhavana), ayatavrtta, ganitasarasamgraha, lilavathi, ganesadaivajna, randavantika, suryasidhhanta, grahalaghava, sadratnamala, mandavrtta, sighrartta, Bijaganita, Bakshali manuscript Golavada, Madhyamanayanaprakara, Mahajyanayanaprakara (Method of Computing Great Sines), Lagnaprakarana, Venvaroha, Sphutacandrapti, Aganita-grahacara , Chandravakyani (Table of Moon-mnemonics)	**15**
IV	**Use of Indian Knowledge in Computer Science**	NLP and Sanskrit, Use of Vaidic Mathematics as Programming, Ancient Indian Geometric Concept in Computer Graphics . Case Studies	**15**
Total			**60**

Indian Mathematics in Computer Science Lab

Detailed Syllabus:

Total Lab Hours for the semester = (4 hours per week)

Minimum 20 Laboratory experiments based on the following-

- **Sutras (Mathematical Formulas):** Vedic mathematics is based on a set of concise and easily memoizable aphorisms called "sutras," which serve as guiding principles for performing mathematical operations. Write Programs using any language to demonstrate these Sutras and check its performance with traditional programs.
- **Mental Calculation Techniques**: Vedic mathematics emphasizes mental calculation techniques that allow for rapid computation without the need for pen and paper. These techniques are particularly useful for performing arithmetic operations such as addition, subtraction, multiplication, and division. Write Programme using Mental Calculation Techniques and check its effectiveness.
- **Vertically and Crosswise Method:** One of the fundamental techniques in Vedic mathematics is the vertically and crosswise multiplication method, which provides a systematic approach to multiplying large numbers quickly and efficiently. Create a Program to apply this technique .
- **Digit Sums and Casting Out Nines**: Vedic mathematics employs techniques such as digit sums and casting out nines to check the accuracy of calculations and detect errors. Create a Programm to apply this technique
- **Algebraic Techniques**: Vedic mathematics includes algebraic techniques for solving equations, factorization, and simplification of expressions, which are based on the underlying principles of the sutras. Create programming solutions based on these .
- **Project-Based Learning with Python:** Encourage students to undertake programming projects that explore real-world applications of Vedic mathematics principles in computer science. For example, they could develop programs that use Vedic mathematics techniques in cryptography, algorithm design, or optimization problems.

Text Book:

1. "Indian Mathematics: Engaging with the World from Ancient to Modern Times," by GG Josheph, speaking Tiger, 2016 .
2. "Ancient Indian Leaps into Mathematics",BS Yadav, , Brikausher publication, 2010

Reference Books:

1. R P Kulkarni, Glimpese of Indian Engineering and Technology (Ancient & Medieval period, Munshiram Manoharlal Publishers Pvt. Ltd. 2018

National Credit Hours		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3 * 20 NCH = 60 NCH	2 * 15 NCH = 30 NCH	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Projects)

- **Detailed Syllabus of Minor Course**

Paper III/Subject Name: Front-End Development with Angular		**Subject Code:** CAP 052N40
Course Type: Minor		**Course Level: 200**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Objective:

The objectives of the course are to teach the students about Angular to enable them to create web pages.

Prerequisites: Fundamentals of Web Development and Server Programming

Course Outcomes

On successful completion of the course the students will be able to:		
Sl No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define** Angular framework and its use .	**BT 1**
CO 2	**Understand** the fundamentals of the Angular framework and its architecture.	**BT 2**
CO 3	**Develop** Angular applications using components, modules, services, and dependency injection.	**BT 3**
CO 4	**Implement** routing and navigation to create single-page applications (SPAs).	**BT 4**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Introduction to Angular**	Overview of front-end development and Angular framework Setting up Angular development environment (Node.js, npm) Creating and running a basic Angular application	**15**
II	**Angular Components & Modules**	Understanding Angular components and component architecture, Creating and using components, templates, and data binding, Component communication using input/output properties and event emitters Introduction to Angular modules and their role in application organization, Creating and importing modules in Angular applications, Implementing services for data sharing and business logic	**15**
III	**Angular Routing and Navigation**	Configuring routes and route parameters in Angular applications Implementing navigation and routing guards Lazy loading modules for optimized application loading	**15**
IV	**Reactive Programming with Angular**	Understanding reactive programming concepts and observables, Managing application state with RxJS operators Building reactive forms for user input and validation Making HTTP requests and handling responses in Angular Integrating external APIs and services into Angular applications Error handling and authentication with HTTP interceptors	**15**
Total			**60**

Textbooks:

1. *Angular Development with TypeScript by Yakov Fain and Anton Moiseev*
2. *Full-Stack React, TypeScript, and Node: Build cloud-ready web applications using React 17 with Hooks and GraphQL* , David Choi, 2020, Packt Publishing Limited.

Reference Books:

2. Frank Zammetti, *Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, Python, Django, and Docker*, 2nd Edition, 2022, APress

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3*20 NCH = 60 NCH	-	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

- **Detailed Syllabus of Minor Course**

Paper III/Subject Name: Server-Side Programming with Node JS		**Subject Code:** CAP052N402
Course Type: Minor		**Course Level: 200**
L-T-P-C – 3-0-0-3	**Credit Units: 03**	**Scheme of Evaluation: T**

Objective:

The objectives of the course are to teach the students about Node JS and it's frame work to enable them to create back end of web sites.

Prerequisites: Fundamentals of Web Development and Server Programming

Course Outcomes

On successful completion of the course the students will be able to:		
SI No	**Course Outcome**	**Blooms Taxonomy Level**
CO 1	**Define basic** semantics and syntax of Node JS to work in server-side programming	**BT 1**
CO 2	**Understand** the principles and advantages of server-side programming with Node.js.	**BT 2**
CO 3	**Apply** asynchronous programming techniques using callbacks, promises, and async/await	**BT 3**
CO 4	**Implement** data storage and retrieval using databases (e.g., MongoDB, MySQL) with Node.js.	**BT 4**

Detailed Syllabus:

Modules	**Topics**	**Course content**	**Periods**
I	**Introduction to Node JS & Asynchronous Programming**	Overview of server-side programming and Node.js Installing Node.js and npm, Writing and running basic Node.js applications Understanding asynchronous programming and non-blocking I/O, using callbacks, promises, and async/await for handling asynchronous operations, Error handling and best practices for asynchronous programming	**15**
II	**Express.js framework**	Introduction to Express.js framework for building web applications and APIs Creating routes, handling requests, and sending responses with Express Middleware concept and implementation in Express.js	**15**
III	**Data Storage & Retrieval**	Working with databases in Node.js (MongoDB, MySQL) Connecting to databases and executing CRUD operations Implementing data validation and error handling, Database Integration, Authentication and Authorization	**15**
IV	**Error Handling and Middleware**	Implementing error handling middleware in Express.js Using third-party middleware for request processing and logging, Best practices for middleware development and management Deployment Strategies and Production Considerations	**15**
Total			**60**

Textbooks:

1. *"Node.js Design Patterns" by Mario Casciaro*.
2. *Full-Stack React, TypeScript, and Node: Build cloud-ready web applications using React 17 with Hooks and GraphQL* , David Choi, 2020, Packt Publishing Limited.

Reference Books:

3. Frank Zammetti, *Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, Python, Django, and Docker*, 2nd Edition, 2022, APress

Credit Distribution		
Lecture/ Tutorial	**Practicum**	**Experiential Learning**
3*20 NCH = 60 NCH	-	30 NCH (Problem Solving, Seminar, Case Study, Discussion, Internship, Project)

- **Detailed Syllabus of Ability Enhancement Compulsory Courses (AECC-III/IV)**

Paper VI/Subject Name: Communication and Presentation Skills		**Subject Code: CEN982A401**
L-T-P-C – 1-0-0-1	**Credit Units: 01**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are to prepare students to develop report writing skills, deliver effective presentation and be informed about technology-enabled communication in the 21st century.

Prerequisites: Basic writing skills in English.

Paper VII/Subject Name: Behavioral Science-IV		**Subject Code: BHS982A404**
L-T-P-C – 1-0-0-1	**Credit Units: 01**	**Scheme of Evaluation: TP**

Objective:

The objectives of the course are: