PROGRAMME PROJECT REPORT

Bachelor of Computer Application

(3 Year Programme in accordance with NEP-2020)

SCHOOL OF SCIENCES
U. P. Rajarshi Tandon Open University
Prayagraj

Contents

1. Bachelor's Degree Programme

The National Education Policy (NEP) 2020 envisions a new vision that enable an individual to study one or more specialized areas of interest at a deep level, and also develop capabilities across a range of disciplines including sciences, social sciences, arts, humanities, languages, as well as professional, technical, and vocational subjects. The NEP 2020 focuses on the formulation of expected learning outcomes for all higher education programmes. It states that "National Higher Education Qualifications Framework (NHEQF)" shall be align with the National Skills Qualifications Framework (NSQF) to ease the integration of vocational education into higher education. It also points out that higher education qualifications leading to a degree/diploma/certificate shall be described by the NHEQF in terms of Outcome Based Education (OBE).

The design of Bachelor of Computer Application programme in line with NHEQF offers opportunities and avenues to learn core subjects but also to explore additional avenues of learning beyond the core subjects for holistic development of a learner.

The uniform grading system will also enable potential employers in assessing the performance of the learner. In order to bring uniformity in evaluation system and computation of the Cumulative Grade Point Average (CGPA) based on learner's performance in examinations, guidelines framed by the UGC are followed. Hence, adoption of NHEQF helps to overcome the gap between university degree and employability by introducing skills and competencies in the graduates.

2. BCA Programme

The structure and duration of undergraduate programme of Bachelor of Computer Application in accordance with NEP 2020 includes multiple exit options within this period, with appropriate certifications:

- Level 5: a **certificate** after completing 1 year (2 semesters) of study in the chosen discipline or field, including vocational and professional areas;
- Level 6: a **diploma** after 2 years (4 semesters) of study;
- Level 7: a **Bachelor's** degree after a 3-year (6 semesters) programme.

2.1 Programme Mission & Objectives

In line with the mission of the University to provide flexible learning opportunities to all, particularly to those who could not join regular colleges or universities owing to social, economic and other constraints, the 3-year Undergraduate Programme in BCA aims at providing holistic and value based knowledge and guidance to promote scientific temper in everyday life. The program offers a platform to the learners to fulfill the eligible criteria in various scientific jobs in government and private sector.

The programme aims at the following objectives:

1. Produce knowledgeable and skilled human resources which are employable in IT and ITES.
2. Impart knowledge required for planning, designing and building complex Application Software Systems as well as provide support to automated systems or application.
3. Produce entrepreneurs who can develop customized solutions for small to large Enterprises.
4. To develop academically competent and professionally motivated personnel, equipped with objective, critical thinking, right moral and ethical values that compassionately foster the scientific temper with a sense of social responsibility.
5. To develop students to become globally competent.
6. To inculcate Entrepreneurial skills among students

2.2 Relevance of the Programme with Mission and Goals

The 3-year Undergraduate Programme in BCA is designed with the objective of equipping learners to cope with the emerging trends and challenges in the scientific domain. In congruence with goals of the University the Programme also focuses to provide skilled manpower to the society to meet global demands. The Programme is designed with three major subjects so that a successful learner can go for higher studies in any one of the major subjects of his/ her choice. The Programme also aims at making the learners fit for taking up various jobs.

2.3 Nature of Prospective Target Group of Learners

The Program is targeted to all individuals looking to earn a graduation degree for employment, further higher education, promotion in career and professional development.

2.4 Appropriateness of Programme to be conducted in ODL mode to acquire specific skills & competence

Learning outcomes after Level 5		
Learning Outcomes	**Elements of the descriptor**	**Level 5 (Undergraduate Certificate)**
LO 1	Knowledge and understanding	The graduates should be able to demonstrate the acquisition of: • knowledge of facts, concepts, principles, theories, and processes in broad multidisciplinary learning contexts within the chosen fields of learning in a broad multidisciplinary learning, • understanding of the linkages between the learning areas within and across the chosen fields of study, • procedural knowledge required for performing skilled or paraprofessional tasks associated with the chosen fields of learning.
LO 2	Skills required to perform and accomplish tasks	The graduates should be able to demonstrate the acquisition of: • a range of cognitive and technical skills required for accomplishing assigned tasks relating to the chosen fields of learning in the context of broad multidisciplinary contexts. • cognitive skills required to identify, analyse and synthesize information from a range of sources. • cognitive and technical skills required for selecting and using relevant methods, tools, and materials to assess the appropriateness of approaches to solving problems associated with the chosen fields of learning.
LO 3	Application of knowledge and skills	The graduates should be able to demonstrate the ability to: • apply the acquired operational or technical and theoretical knowledge, and a range of cognitive and practical skills to select and use basic methods, tools, materials, and information to generate solutions to specific problems relating to the chosen fields of learning
LO 4	Generic learning outcomes	The graduates should be able to demonstrate the ability to: • listen carefully, read texts related to the chosen fields of study analytically and present information in a clear and concise manner to different groups/audiences.

		• express thoughts and ideas effectively in writing and orally and present the results/findings of the experiments carried out in a clear and concise manner to different groups. • meet one's own learning needs relating to the chosen fields of learning. • pursue self-directed and self-managed learning to upgrade knowledge and skills required for higher level of education and training. • gather and interpret relevant quantitative and qualitative data to identify problems, • critically evaluate principles and theories associated with the chosen fields of learning. • make judgement and take decision, based on analysis of data and evidence, for formulating responses to issues/problems associated with the chosen fields of learning, requiring the exercise of some personal responsibility for action and outputs/outcomes.
LO 5	Constitutional, humanistic, ethical and moral values	The graduates should be able to demonstrate the willingness to: • practice constitutional, humanistic, ethical, and moral values in one's life, and practice these values in real-life situations, • put forward convincing arguments to respond to the ethical and moral issues associated with the chosen fields of learning.
LO 6	Employment ready skills, and entrepreneurship skills and mindset	The graduates should be able to demonstrate the acquisition of: • knowledge and a basket of essential skills, required to perform effectively in a defined job relating to the chosen fields of study, • ability to exercise responsibility for the completion of assigned tasks and for the outputs of own work, and to take some responsibility for group work and output as a member of the group.

Learning outcomes after Level 6		
Learning Outcomes	**Elements of the descriptor**	**Level 6 (Undergraduate Diploma)**
LO 1	Knowledge and understanding	The graduates should be able to demonstrate the acquisition of: • theoretical and technical knowledge in broad multidisciplinary contexts within the chosen fields of learning, • deeper knowledge and understanding of one of the learning areas and its underlying principles and theories, • procedural knowledge required for performing skilled or paraprofessional tasks associated with the chosen fields of learning.
LO 2	Skills required to perform and accomplish tasks	The graduates should be able to demonstrate the acquisition of: • cognitive and technical skills required for performing and accomplishing complex tasks relating to the chosen fields of learning, • cognitive and technical skills required to analyse and synthesize ideas and information from a range of sources and act on information to generate solutions to specific problems associated with the chosen fields of learning.

LO 3	Application of knowledge and skills	The graduates should be able to demonstrate the ability to: • apply the acquired specialized or theoretical knowledge, and a range of cognitive and practical skills to gather quantitative and qualitative data, • select and apply basic methods, tools, materials, and information to formulate solutions to problems related to the chosen field(s) of learning.
LO 4	Generic learning outcomes	The graduates should be able to demonstrate the ability to: • listen carefully, read texts related to the chosen fields of learning analytically and present complex information in a clear and concise manner to different groups/audiences, • communicate in writing and orally the information, arguments, and results of the experiments and studies conducted accurately and effectively to specialist and non-specialist audience. • meet one's own learning needs relating to the chosen field(s) of learning, work/vocation, and an area of professional practice, • pursue self-paced and self-directed learning to upgrade knowledge and skills required for pursuing higher level of education and training. • critically evaluate the essential theories, policies, and practices by following scientific approach to knowledge development. • make judgement and take decision, based on the analysis and evaluation of information, for determining solutions to a variety of unpredictable problems associated with the chosen fields of learning, taking responsibility for the nature and quality of outputs.
LO 5	Constitutional, humanistic, ethical and moral values	The graduates should demonstrate the willingness and ability to: • embrace the constitutional, humanistic, ethical, and moral values, and practice these values in life, and take a position regarding these values, • formulate arguments in support of actions to address issues relating to the ethical and moral issues relating to the chosen fields of learning, including environmental and sustainable development issues, from multiple perspectives.
LO 6	Employment ready skills, and entrepreneurship skills and mindset	The graduates should be able to demonstrate the acquisition of knowledge and essential skills set that are necessary to: • take up job/employment or professional practice requiring the exercise of full personal responsibility for the completion of tasks and for the outputs of own work. • exercise self- management within the guidelines of study and work contexts. • supervise the routine work of others, taking some responsibility for the evaluation and improvement of work or study activities.

Learning outcomes after Level 7		
Learning Outcomes	**Elements of the descriptor**	**Level 7 (Bachelor of Computer Application)**
LO 1	Knowledge and understanding	The graduates should be able to demonstrate the acquisition of: • comprehensive, factual, theoretical, and specialized knowledge in broad multidisciplinary contexts with depth in the underlying principles and theories relating to one or more fields of learning.

		• knowledge of the current and emerging issues and developments within the chosen field(s) of learning. • procedural knowledge required for performing and accomplishing professional tasks associated with the chosen fields of learning.
LO 2	Skills required to perform and accomplish tasks	The graduates should be able to demonstrate the acquisition of: • cognitive and technical skills required for performing and accomplishing complex tasks relating to the chosen fields of learning. • cognitive and technical skills required to evaluate and analyse complex ideas, • cognitive and technical skills required to generate solutions to specific problems associated with the chosen fields of learning.
LO 3	Application of knowledge and skills	The graduates should be able to demonstrate the ability to: • apply the acquired specialized technical or theoretical knowledge, and cognitive and practical skills to gather and analyse quantitative/ qualitative data to assess the appropriateness of different approaches to solving problems, • employ the right approach to generate solutions to problems related to the chosen fields of learning.
LO 4	Generic learning outcomes	The graduates should be able to demonstrate the ability to: • listen carefully, to read text related to the chosen fields of learning analytically and present complex information in a clear and concise manner to different groups/audiences. • communicate in writing and orally the constructs and methodologies adopted for the studies undertaken relating to the chosen fields of learning, • make coherent arguments to support the findings/results of the study undertaken to specialist and non-specialist audience. • meet one's own learning needs relating to the chosen field(s) of learning, • pursue self-paced and self-directed learning to upgrade knowledge and skills that will help adapt to changing demands of workplace and pursue higher level of education and training. • critically evaluate evidence for taking actions to generate solutions to specific problems associated with the chosen fields of learning based on empirical evidence. • make judgement and take decisions based on the analysis and evaluation of information for formulating responses to problems, including real-life problems, • exercise judgement across a broad range of functions based on empirical evidence, for determining personal and/or group actions to generate solutions to specific problem s associated with the chosen fields of learning.
LO 5	Constitutional, humanistic, ethical and moral values	The graduates should be able to demonstrate the willingness and ability to: • embrace the constitutional, humanistic, ethical, and moral values, and practice these values in life. • identify ethical issues in science, • formulate coherent arguments about ethical and moral issues, including environmental and sustainable development issues. • follow ethical practices in all aspects of research and development

LO 6	Employment ready skills, and entrepreneurship skills and mindset	The graduates should be able to demonstrate the acquisition of: • knowledge and essential skills set and competence that are necessary to: take up a professional job and professional practice, • entrepreneurship skills and mindset required for setting up and running an economic enterprise or pursuing self-employment • exercise management and supervision in the contexts of work or study activities involving unpredictable work processes and working environment

2.5 Instructional Design

2.5.1 3-year BCA Programme Structure

The University follows the credit system in all its programmes. One credit is equal to 30 hours of learner's study time which is equivalent to 15 lectures in conventional system. To earn a bachelor's degree, a learner must earn 120 credits in a minimum of six semesters (three years) with 20 credits per semester. For earning 120 credits, a learner must go through the following Programme Structure:

Programme Structure of BCA under NHEQF

Level	Year	Sem	Core Course 1	Core Course 2	Core Course 3	Core Course 4	Ability Enhancement Compulsory Course (AECC)	Discipline Specific Elective Course (DEC)	Practical Lab/ Dissertation with viva voce	Total credit
5	1	1st	4	4	4	4	2		2	**20**
		2nd	4	4	4	4	2		2	**20**
6	2	3rd	4	4	4	4	2		2	**20**
		4th	4	4	4	4	2		2	**20**
7	3	5th	4	4	4	--	--	4	4	**20**
		6th	4	4	--	--	--	4	8	**20**
			Total credit							**120**

Explanation of terms used for categorization of courses:

A. **Course 1 to 4:** A course, which should compulsorily be studied by a learner as a core requirement is termed as a Core course.

B. **Ability Enhancement Compulsory Courses (AECC):** "AECC" courses are the courses based upon the content that leads to knowledge enhancement.

Semester	Ability Enhancement Courses (AECC)
1	Ability Enhancement Course in English [AECEG] OR Ability Enhancement Course in Hindi [AECHD]
2	Ability Enhancement Course in Human Rights and Duties [AECHRD] OR Ability Enhancement Course in Health & Hygiene [AECHH]
3	Ability Enhancement Course in Environment Awareness [AECEA] OR Ability Enhancement Course in Solid Waste Management [AESWM]
4	Ability Enhancement Course in Nutrition for Community [AECNC] OR Ability Enhancement Course in Disaster Management [AECDM]

C. **Practical Lab:** Lab based on theory courses for implementing the algorithms discussed in theory papers.
D. The learner has to choose any one course from **Discipline Specific Elective Course** in fifth and sixth semester.
E. **Industrial Training/ Survey/ Research Project/ Field Work/Apprenticeship/ Dissertation/Internship:** An elective course designed to acquire special/advanced knowledge, such as supplement study/support study to a project work, and a learner studies such a course on his own with an advisory support by a counsellor/faculty member. Currently, Dissertation is offered under code; **BCA-130RP**.

2.5.2 Course curriculum: The detail of syllabus is given in Appendix-I

2.5.3 Language of Instruction: English. However, learner can write assignment and give Term End Examination (TEE) either in Hindi or English.

2.5.4 Duration of the Programme
Minimum duration in years: 03 Maximum duration in years: 06

2.5.5 Faculty & Support Staff
Professor (1), Assistant Professor (4) and support staff (3)

2.6 Instructional Delivery Mechanisms

The Open University system is more learner-oriented, and the student is an active participant in the teaching-learning process. Most of the instructions are imparted through distance rather than face-to-face communication.

The University follows a multi-media approach for instruction. It comprises of:

- self-instructional printed material (Self Learning Material)
- audio and video lectures
- face-to-face counselling
- assignments
- laboratory work
- Project work in some courses
- teleconference/web conference
- Web Enabled Academic Support Portal

2.6.1 Self-Learning Material

The Self Learning Material (SLMs) are prepared in line with the UGC guidelines on preparation of SLMs. The prepared study materials are self-instructional in nature.

The course material is divided into blocks. Each block contains a few units. Lessons, which are called Units, are structured to facilitate self-study. The units of a block have similar nature of contents. The first page of each block indicates the numbers and titles of the units comprising the block. In the first block of each course, we start with course introduction. This is followed by a brief introduction to the block. After the block introduction, emphasis is given on contribution of ancient Indian knowledge into that specific course. Next, each unit begins with an introduction totalk about the contents of the unit. The list of objectives are outlined to expect the learning based outcome after working through the unit. This is followed by the main body

of the unit, which is divided into various sections and sub-sections. Each unit is summarized with the main highlights of the contents.

Each unit have several "Check Your Progress" Questions and Terminal Questions /exercises. These questions help the learner to assess his/her understanding of the subject contents. At the end of units,additional references/books/suggested online weblink for MOOCs/Open Educational Resources for additional reading are suggested.

2.6.2 Audio and Video lectures

Apart from SLM, audio and video lectures have been prepared for some courses. The audio-video material is supplementary to print material. The video lectures are available at YouTube channel of university

2.6.3 Counselling Classes

The face to face (F2F) counselling classes are conducted at head quarter and study centers. The purpose of such a contact class is to answer some of questions and clarify the doubts of learner which may not be possible through any other means of communication. Well experienced counsellors at study centers provide counselling and guidance to the learner in the courses that (s)he has chosen for study. The counselling sessions for each of the courses will be held at suitable intervals throughout the whole academic session. The time table for counselling classes are displayed at head quarter as well as by the coordinator of study center, however, attending counselling sessions is not compulsory. It is noted that to attend the counselling sessions, learner has to go through the course materials and note down the points to be discussed as it is not a regular class or lectures.

2.6.4 Assignments

The purpose of assignments is to test the comprehension of the learning material that learner receives and also help to get through the courses by providing self-feedback to the learner. The course content given in the SLM will be sufficient for answering the assignments.

Assignments constitute the continuous evaluation component of a course. The assignments are available at the SLM section of the home page of university website. In any case, learner has to submit assignment before appearing in the examination for any course. The assignments of a course carry 30% weightage while 70% weightage is given to the term-end examination (TEE). The marks obtained by learner in the assignments will be counted in the final result. Therefore, It is advised to take assignments seriously. However, there will be no written assignments for Lab courses.

2.6.5 Laboratory Work

Laboratory courses are an integral component of the Bachelor of Computer Application programme. While designing the curricula for laboratory courses, particular care has been taken to weed out experiments not significant to the present-day state of the discipline. Importance has been given to the utility of an experiment with respect to real life experience, development of experimental skills, and industrial applications. It is planned to phase the laboratory courses during suitable periods (such as summer or autumn vacations) so that in-service persons can take them without difficulty. Laboratory courses worth 2 credits will require full-time presence of the student at the Study Centre for one week continuously. During this time a student has to work for around 60 hours. Around 40 hours would be spent on experimental work and the remaining time will be used for doing calculations, preparations of records, viewing or listening to the video/audio programmes.

2.6.6 Teleconference/Web conference

Teleconference/web conference, using done through ZOOM/webex in form of online special counselling sessions is another medium to impart instruction to and facilitate learning for a distance learner. The students concerned would be informed about the teleconferencing schedule and the place where it is to be conducted by sending bulk SMS.

2.6.7 Web Enabled Academic Support Portal

The University also provide Web Enabled Academic Support Portal to access the course materials, assignments, and other learning resources.

2.6.8 Learner Support Service Systems

(a) Study Centre

A Study Centre has following major functions:

(i) **Counselling:** Counselling is an important aspect of Open University System. Face to face contact-cum-counselling classes for the courses will be provided at the Study Centre. The detailed programme of the contact-cum-counselling sessions will be sent to the learner by the Coordinator of the Study Centre. In these sessions learner will get an opportunity to discuss with the Counsellors his/her problems pertaining to the courses of study.

(ii) **Evaluation of Assignments:** The evaluation of Tutor Marked Assignments (TMA) will be done by the Counsellors at the Study Centre. The evaluated assignments will be returned to the learner by the Coordinator of Study Centre with tutor comments and marks obtained in TMAs. These comments will help the learner in his/her studies.

(iii) **Library:** Every Study Centre will have a library having relevant course materials, reference books suggested for supplementary reading prepared for the course(s).

(iv) **Information and Advice:** The learner will be given relevant information about the courses offered by the University. Facilities are also provided to give him/her guidance in choosing courses.

(v) **Interaction with fellow-students:** In the Study Centre learner will have an opportunity to interact with fellow students. This may lead to the formation of self-help groups.

(b) Learner Support Services (LSS)

The University has formed an LSS cell at the head quarter. The LSS cell coordinate with the Study Centre to get rid of any problem faced by the learner.

2.7 Procedure for admissions, curriculum transaction and evaluation

2.7.1 Admission Procedure

(a) The detailed information regarding admission will be given on the UPRTOU website and on the admission portal. Learners seeking admission shall apply online.

(b) Direct admission to 3-year Bachelor of Computer Application program is offered to the interested candidates.

(c) Eligibility:

10+2

OR

3-years diploma from Board of Technical Education / equivalent

OR

Two year ITI programme of any trade after 10 standard.

2.7.2 Programme Fee: Rs. 13000/- year. The fee is deposited through online admission portal only.

2.7.3 Evaluation

The evaluation consists of two components: (1) continuous evaluation through assignments, and (2) term-end examination. Learner must pass both in continuous evaluation as well as in the term-end examination of a course to earn the credits assigned to that course. For each course there shall be one written Terminal Examination. The evaluation of every course shall be in two parts that is 30% internal weightage through assignments and 70% external weightage through terminal exams.

(a) Theory course	**Max. Marks**
Terminal Examination	70
Assignment	30
Total	**100**

(b) Practical course:	**Max. Marks**
Terminal Practical Examination	100

Marks of Terminal Practical Examination shall be awarded as per following scheme:

i.	Write up /theory work	30
ii.	Viva-voce	30
iii.	Execution/Performance/Demonstration	20
iv.	Lab Record	20

The following 10-Point Grading System for evaluating learners' achievement is used for CBCS programmes:

10-Point Grading System in the light of UGC-CBCS Guidelines

Letter Grade	Grade Point	% Range
O (Outstanding)	10	91-100
A+ (Excellent)	9	81-90
A (Very Good)	8	71-80
B+ (Good)	7	61-70
B (Above Average)	6	51-60
C (Average)	5	41-50
P (Pass)	4	36-40
NC (Not Completed)	0	0-35
Ab (Absent)	0	
Q	Qualified	Applicable only for Non-Credit courses
NQ	Not Qualified	

Learner is required to score at least a 'P' grade (36% marks) in both the continuous evaluation (assignments) as well as the term-end examination. In the overall computation also, learner must get at least a 'P' grade in each course to be eligible for the B. Sc. degree.

Computation of CGPA and SGPA

(a) Following formula shall be used for calculation of CGPA and SGPA

For jth semester SGPA (Sj) = Σ (Ci *Gi)/ Σ Ci	where, Ci = number of credits of the ith course in jth semester Gi= grade point scored by the learner in the ith course in jth semester.
CGPA = Σ (Cj *Sj)/ Σ Cj	where, Sj = SGPA of the jth semester Cj = total number of credits in the jth semester

The CGPA and CGPA shall be rounded off up to the two decimal points. (For e.g., if a learner obtained 7.2345, then it will be written as 7.23 or if s(he) obtained 7.23675 then it be will written as 7.24)

CGPA will be converted into percentage according to the following formula:

Equivalent Percentage = CGPA * 9.5

(b) Award of Division

The learner will be awarded division according to the following table:

Division	Classification
1st Division	6.31 or more and less than 10 CGPA
2nd Division	4.73 or more and less than 6.31 CGPA
3rd Division	3.78 or more and less than 4.73 CGPA

2.7.4 Multiple Entry and Multiple Exit options

The 3-year BCA programme is an Outcome-Based Education (OBE) for qualifications of different types. The qualification types and examples of title/nomenclature for qualifications within each type are indicated in Table 1.

Table 1

Level	Qualification title	Programme duration	Entry Option	Exit option
5	Certificate in Computer Application	Programme duration: First year (first two semesters) of the BCA programme	10+2 **OR** 3-years diploma from Board of Technical Education / equivalent **OR** Two year ITI programme of any trade after 10 standard.	Awarded with Certificate in Computer Application
6	Undergraduate Diploma in Computer Application	Programme duration: First two years (first four semesters) of the BCA programme	Undergraduate Certificate obtained after completing the first year (two semesters) of the BCA programme	Awarded with Diploma in Computer Application
7	Bachelor in Computer Application	Programme duration: First three years (first six semesters) of the of the BCA programme	Undergraduate diploma obtained after completing two years (four semesters) of the BCA programme	Awarded with Bachelor of Computer Application

2.8 Requirement of the laboratory support and Library Resources

The practical sessions are held in the science laboratories of the Study Centre. In these labs, the learner will have the facility to use the equipment andconsumables relevant to the syllabus. The SLM, supplementary text audio and video material of the various courses of the program is available through the online study portal of the University. The University also have a subscription of National Digital Library to provide the learners with the ability to enhance access to information and knowledge of various courses of the programme.

2.9 Cost estimate of the programme and the provisions

3-year Bachelor of Computer Application programme consists of 25 theory courses and 05 laboratory and 01 research project. Each course is of 4 credits which consists of approx. 10 units. The total approximated expenditure on the development of 25 courses is:

S. No.	Item	Cost per Unit (writing & editing)	Total cost (Rs.)
1	Total no. of units in 25 courses = 250	4500	11,25,000
2	BOS Meetings etc.	100000	100000
		Total	12,25,000

2.10 Quality assurance mechanism and expected programme outcomes

(a) **Quality assurance mechanism:** The program structure is developed under the guidance of the Board of studies comprising external expert members of the concerned subjects followed by the School board. The program structure and syllabus is approved by the Academic Council of the University. The course structure and syllabus is reviewed time to time according to the feedback received from the stakeholders and societal needs.

The Centre for Internal Quality Assurance will monitor, improve and enhance effectiveness of the program through the following:

- ✓ Annual academic audit
- ✓ Feedback analysis for quality improvement
- ✓ Regular faculty development programs
- ✓ Standardization of learning resources
- ✓ Periodic revision of program depending upon the changing trends by communicating to the concerned school

(b) **Expected programme outcomes (POs)**

PO1	An ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline.
PO 2	An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
PO 3	An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
PO 4	An ability to function effectively on teams to accomplish a common goal
PO 5	An understanding of professional, ethical, legal, security and social issues and responsibilities
PO 6	An ability to analyze the local and global impact of computing on individuals, organizations, and society.
PO 7	Recognition of the need for and an ability to engage in continuing professional development.
PO 8	An ability to use current techniques, skills, and tools necessary for computing practice.

APPENDIX-I

Academic Year: 2023-2024

Year wise Structure & Syllabi of Bachelor of Computer Application

Year	Semester	Course Code	Paper Title	Type of Course	Max. Marks	Credits
First	1	**Core Courses**				
		BCA-101N	Computer Fundamental & PC Software	Theory	100	4
		BCA-102N	C Programming	Theory	100	4
		BCA-103N	Data Structures	Theory	100	4
		BCA-104N	Basic Mathematics	Theory	100	4
		BCA-105P	Practical Based on BCA -102 & 103	Practical	100	2
		Ability Enhancement Compulsory Courses				
		AECEG OR AECHD	Ability Enhancement Course in English [AECEG] OR Ability Enhancement Course in Hindi [AECHD]	Theory	100	2
	2	**Core Courses**				
		BCA-106N	Numerical Analysis	Theory	100	4
		BCA-107N	Multimedia	Theory	100	4
		BCA-108N	Discrete Mathematics	Theory	100	4
		BCA-109N	C++ and Object Oriented Programming	Theory	100	4
		BCA-110N	Practical Based on BCA -106 & BCA-109	Practical	100	2
		Ability Enhancement Compulsory Courses				
		AECHRD OR AECHH	Ability Enhancement Course in Human Rights and Duties [AECHRD] OR Ability Enhancement Course in Health &Hygiene[AECHH]	Theory	100	2
Second	3	**Core Courses**				
		BCA-111N	Database Management System	Theory	100	4
		BCA-112N	Operating System	Theory	100	4
		BCA-113N	Software Engineering	Theory	100	4
		BCA-114N	Principle of Programming Languages	Theory	100	4
		BCA-115P	Practical Based on BCA -111 & BCA-112	Practical	100	2
		Ability Enhancement Compulsory Courses				
		AECEA OR AECSWM	Ability Enhancement Course in Environment Awareness [AECEA] OR Ability Enhancement Course in Solid Waste Management [AESWM]	Theory	100	2
	4	**Core Courses**				
		BCA-116N	Computer Network	Theory	100	4
		BCA-117N	Java Programming	Theory	100	4
		BCA-118N	Windows Programming	Theory	100	4
		BCA-119N	Computer Organization	Theory	100	4
		BCA-120P	Practical Based on BCA-117 & BCA-118	Practical	100	2
		Ability Enhancement Compulsory Courses				
		AECNC OR AECDM	Ability Enhancement Course in Nutrition for Community [AECNC] OR Ability Enhancement Course in Disaster Management [AECDM]	Theory	100	2
	5	**Core Courses**				
		BCA-121N	Information and Network Security	Theory	100	4

Third		BCA-122N	Design and Analysis of Algorithm	Theory	100	4
		BCA-123N	Computer Graphics	Theory	100	4
		BCA-124P	Practical Based on BCA-122 & BCA-123	Practical	100	4
		Discipline Elective Course (select any one)				
		BCA-EA OR BCA-EB	Web Technology OR Client Server Technology	Theory	100	4
	6	**Core Courses**				
		BCA-127N	Python Programming	Theory	100	4
		BCA-128N	Soft Computing	Theory	100	4
		BCA-130RP	Project with Viva Voce	Project	200	8
		Discipline Elective Course (select any one)				
		BCA-EC OR BCA-ED	Computer Architecture OR Microprocessor and its applications	Theory	100	4
				Total Max. Marks/Credit	**3400**	**120**

Programme: **BCA**	Year: **First**	Semester: **I**
Subject: BCA		
Course Code: **BCA-102N**	Course Title: **Computer Fundamental & PC Software**	
Course Objectives: This course deals with fundamentals of computer. This includes generations of computer, evolution and development of microprocessor, input and output devices, primary and secondary storage devices, programming languages etc. It offers understanding of the hardware and software aspects of the computer like operating system, application software and system software. It provides an overview of functions and working of central processing unit, motherboard and other peripherals.		
Course Outcomes: **CO1** Understand hardware components of computer system such as memory system organization, input/output devices. **CO2** Aware of software components of computer system, component of programming languages and operating system concepts. **CO3** Explain data communication and networking related technology. **CO4** Analyze of computer security and viruses. **CO5** Describe concepts related to graphical user interface. **CO6** Familiarize with word processing application and presentation software: MS Word & MS PowerPoint.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	
Unit 1	**Computer Basics:** Algorithms, A Simple Model of a Computer, Characteristics of Computers, Problem-solving Using Computers. Data Representation: Representation of Characters in computers, Representation of Integers, Representation of Fractions, Hexadecimal Representation of Numbers, Decimal to Binary Conversion, Error-detecting codes. Input & Output Devices: Description of Computer Input Units, Other Input Methods, Computer Output Units (Printers, Plotters) Operating Systems: History and Evolution. Main functions of OS Multitasking, Multiprocessing. Time Sharing, Real Time OS with Examples Database Management System: Purpose and Organization of Database, Introduction to Data Models.	
Unit 2	**Disk operating system(DOS):** Introduction, history & versions of DOS, DOS basics- Physical structure of disk, drive name, FAT, file & directory structure and naming rules, booting process, DOS system files, DOS commands- internal & external.	
Unit 3	**Windows Operating System:** Windows concepts. Features, Windows Structure, Desktop, Taskbar, Start Menu, My Computer, Recycle Bin, Windows Accessories- Calculator, Notepad, Paint, Wordpad, Character Map, Windows Explorer, Entertainment, Managing Hardware & Software- Installation of Hardware & Software, Using Scanner, System Tools, Communication, Sharing Information between programs.	
Unit 4	**Word Processing; MS-Word:** Features, Creating, Saving and Opening Documents in Word, Interface, Toolbars, Ruler, Menus, Keyboard Shortcut, Editing, Previewing, Printing,& Formatting a Document, Advanced Features of MS Word, Find & Replace, Using Thesaurus, Using Auto- Multiple Functions, Mail Merge, Handling Graphics, Tables & Charts, Converting a word document into various formats like- Text, Rich. Text format, Word perfect, HTML etc.	
Unit 5	**Worksheet- MS-Excel:** Worksheet basics, creating worksheet, entering into worksheet, heading information, data, text, dates, alphanumeric values, saving & quitting worksheet, Opening and moving around in an existing worksheet, Toolbars and Menus, Keyboard shortcuts, Working with single and multiple workbook, working with formulae & cell referencing, Auto sum, Coping formulae, Absolute & relative addressing, Worksheet with	

	ranges, formatting of worksheet, Previewing & Printing worksheet, Graphs and charts. Database, Creating and Using macros, multiple worksheets- concepts, creating and using.

Suggested Readings:

1. Tanenbaum A.S.: Structured Computer Organization, EEE, Prentice hall India, 5th Edition.
2. Stallings W.: Computer Organization & Architecture, Prentice hall India, 5th Edition.
3. Rajaraman V.: Fundamentals of Computers, EEE, Prentice Hall India.

Suggested online courses (MOOCs)

1. Computer Fundamentals By Prof. Sanjay Tanwani, Devi Ahilya Viswavidyalaya, Indore https://onlinecourses.swayam2.ac.in/cec19_cs06/preview

This course can be opted as an elective by the students of following subjects: **B.Sc.(Computer Science), B.Sc. (Statistics)**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **First**	Semester: **I**
Subject: BCA		
Course Code: **BCA-102N**	Course Title: **C Programming**	
Course Objectives: Briefly glimpse the basics of software engineering practices like modularization, commenting, and naming conventions which help in collaborating and programming in teams. This course emphasizes translating any algorithm into C code and writing efficient and maintainable C code.		
Course Outcomes: **CO1** Develop a sound approach to problem solving using C programming language. **CO2** Implement techniques like recursion and iteration are learnt to solve a problem. **CO3** Demonstrate programming concepts like pointers, structures. **CO4** Reading, understanding and modifying code written by others.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Introduction to algorithms and program design**
Unit 1	**Introduction to Algorithms**: Problem solving techniques, Algorithm
Unit 2	**Pseudo-codes and Flowcharts**: Tools of Algorithm, Pseudo codes, Flowchart
Unit 3	**Program design principles**: Introduction to computer programming, Program design principles, Programming techniques, Program Errors
Block 2	**Introduction to the 'C' programming language**
Unit 4	**Introduction:** History of C Language, Structure of a 'C' program, Creating and Executing a 'C' program,
Unit 5	**Data Types in 'C':** Character Set of 'C' language, Trigraph characters, Tokens, Identifiers, Keywords, Constants, Data types, Variables
Unit 6	**Storage Classes:** Scope and lifetime of variable, Storage classes, Automatic storage class, Register storage class, Static storage class, External storage class
Unit 7	**Input and Output Functions:** Reading a single character, Writing a single character, Formatted Input-Output, Formatted Input, Formatted Output
Block 2	**Operator and Control Structures**
Unit 8	**Operators and Expressions:** Arithmetic operators, Relational operators, Logical operators, Assignment operators, Increment and decrement operators, Conditional operators, Bitwise operators, Special operators, Operator Precedence and Associativity, lvalue and rvalue, Type casting: Promotion and Demotion of variable types
Unit 9	**Decision Structures in 'C':** if statement, if else statement, nested if … else statement, switch statement, goto statment
Unit 10	**Loop Structures in 'C':** for statement, while statement, do while statement, break statement, continue statement
Unit 11	**Arrays:** One dimensional array, Two dimensional array, Multidimensional arrays, Strings, String handling functions, Character functions
Block 3	**Advanced Features of C**
Unit 12	**Pointers:** Pointers and Address (&) operator, Pointer declaration and Initialization , Indirection operator, Pointer Arithmetic, Arrays and Pointers, Character strings and Pointers, Array of Pointers, Pointer to Pointer
Unit 13	**Functions:** Functions, user-defined functions, categories of function, returning non-integer values, function arguments, recursion, arrays as function arguments

Unit 14	**Structures, Unions, enum and typedef:** Structure definition, Structures within structures, Structures as function arguments, Pointers to structures, Unions, Enumerated data type, Type definition
Unit 15	**File and Memory Management in 'C':** Files, File Pointer Variable, Opening a file, Reading and writing to files, File Status Functions, Random Access to files, Command Line Arguments, Memory management
Unit 16	**Preprocessor Directives and Error reporting:** Macro directives, Conditional directives, Control directives, Error reporting

Suggested Readings:

1. Kanetkar, Yashavant. *Let us C*. BPB publications, 2018.
2. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India

Suggested online courses (MOOCs)

1. Introduction to programming in C By Prof. Satyadev Nandakumar, IIT Kanpur
   https://onlinecourses.nptel.ac.in/noc23_cs02/preview
2. Problem Solving Through Programming In C By Prof. Anupam Basu, IIT Kharagpur
   https://onlinecourses.nptel.ac.in/noc23_cs53/preview

This course can be opted as an elective by the students of following subjects: **B.Sc.(Computer Science), B.Sc.(Statistics), M.Sc. (Mathematics)**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **First**	Semester: **I**
Subject: BCA		
Course Code: **BCA-103N**	Course Title: **Data Structures**	
Course Objectives: The objective of the course is to familiarize students with basic data structures and their use in fundamental algorithms.		
Course Outcomes: **CO1:** Understand basic data structures such as arrays, strings, and linked lists. **CO2:** Study linear data structures such as stacks and queues and understand their difference. **CO3:** Describe the hash function and concepts of collision and its resolution methods. **CO4:** Study tree, heap and graphs along with their basic operations. **CO5:** Study different techniques for solving problems like sorting and searching		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	
Block 1	**BLOCK - 1**	
Unit 1	**Introduction to data structure:** Algorithm, Basic criteria for algorithms, Data type, Data structure, Data representation, linear and nonlinear data structure.	
Unit 2	**Basics of algorithm:** Algorithm, Basics of complexity of algorithm	
Unit 3	**Array:** Definition, Representation of array, Single and multi-dimensional array, address calculation (one dimensional, two dimensional, multidimensional), sparse matrices	
Block 2	**BIOCK – 2**	
Unit 4	**Stack:** Definition, Operations on stacks, Array representation and implementation of stack; infix, prefix and postfix representation of expression and evaluation multiple stacks, Application of stacks.	
Unit 5	**Recursion:** Recursive definition and processes, some named problems of recursion, principle of recursion: designing recursive algorithm, how recursion works, tail recursion.	
Unit 6	**Queue:** Definition, operation on queues, circular queue, dequeue, priority queue, Application of queue.	
Block 3	**BLOCK 3**	
Unit 7	**Linked List:** Representation and implementation of single linked list, Operations in the singly linked list, stack and queue as a linked list, circularly linked list, doubly linked list, circularly doubly linked list, Application of linked list: polynomial representation and addition, garbage collection	
Unit 8	**Tree:** Basic terminology, binary tree, binary tree representation, complete binary tree, extended binary tree, array and linked list representations, traversing binary tree, threaded binary tree, binary search tree, Operations on BST, AVL tree, Operations on AVL tree, B-tree Insertion and deletion in B tree.	
Unit 9	**Graph:** Basic terminology Graph representation Depth first search, breadth first search, topological sort, connected components, spanning tree, minimum cost spanning tree, Kruskal's and prim's algorithm, Shortest path algorithms: Bellman Ford Algorithm, Dijkstra’s algorithm, Floyd-Warshall algorithm.	
Block 4	**BLOCK- 4**	
Unit 10	**Searching and sorting:** Sequential search, binary search, comparison and analysis, Selection sort, Bubble sort, Insertion sort, Heap sort, Quick Sort, Merge sort, Shell sort, radix sort.	
Unit 11	**Hashing:** Hash table, hash function, collision resolution strategies, hash table implementation.	

Unit 12	**File Structure:** Terminology, File organization, Sequential files, Direct File organization, Indexed Sequential file organization.

Suggested Readings:

1. E Horowitz and S. Sahni: Fundamentals of Data Structures in C, Second Edition, Universities Press, Hyderabad.
2. R.L. Kruse: Data Structures & Program Design in C, PHI.

Suggested online courses (MOOCs)

1. Programming and Data Structure, IIT Kharagpur by Dr. P.P.Chakraborty https://nptel.ac.in/courses/106105085
2. NOC:Programming and Data structures (PDS), IIT Madras by Dr. N S. Narayanaswamy https://nptel.ac.in/courses/106106130
3. NOC:Programming, Data Structures and Algorithms, IIT Madras by Prof. Hema A Murthy, Dr. N S. Narayanaswamy, Prof. Shankar Balachandran https://nptel.ac.in/courses/106106127
4. Data Structures And Algorithms, IIT Delhi by Prof. Naveen Garg https://nptel.ac.in/courses/106102064

This course can be opted as an elective by the students of following subjects: N.A.

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **First**	Semester: **I**
Subject: BCA		
Course Code: **BCA-104N**	Course Title: **Basic Mathematics**	
Course Objectives: The course offers an introduction to basic mathematics which is essentials for a computer science student to go ahead and study any other topics in the subject. The emphasis will be on problem solving as well as proofs. This course focuses on Set Theory, Functions, Limits and Continuity, Quadratic Equation and Calculus.		
Course Outcomes: **CO1** Use logical notation to define and reason about fundamental mathematical concepts such as sets, relations, functions. **CO2** Find limits of functions. **CO3** Analyze and apply the notions of continuity and differentiability to algebraic and transcendental functions.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	
Unit 1	**Sets &. Relations :** Sets and elements, Equal sets, Universal set & Empty set, Subsets, Venn diagrams, Basic operations on sets, Union & Intersection, Complements, Difference, Symmetric Difference, Fundamental Products, Algebra of sets and Duality, Finite Sets, Counting Principle, Classes of sets, Power sets, Partitions, Mathematical Induction, Cartesian Products of Sets, Relations, Pictorial representations of Relations, Composition of relations, Types of relations, Equivalence Relations, Partial ordering relations.	
Unit 2	**Functions, Limits and Continuity** : Functions, Kinds of Functions , Concept of real function, Domain and Range (simple cases), Composition Function, One-to-one, onto, into, invertible functions, Mathematical Functions , Exponential and Logarithmic Functions, Graph of functions (plotting of linear function, absolute value function, parabolic functions, Sin(x), Cos(x), tan(x), reciprocal function, ex, log x, Signum function), Polar coordinates and graph, Limit of variable, Limit of function, Evaluation of limits of various types of functions, Continuity & Discontinuity at a point, Continuity over an interval. Trigonometrical Functions: Definitions, proofs for any angle 0, signs of ratios, ratios of some standard angles.	
Unit 3	**Quadratic Equation:** Solution of Quadratic Equations, Nature of Roots. Co-ordinates and Loci: Cartesian co-ordinate system, Introduction to Polar co-ordinates, distance between two points, section formulae, Area of triangle, Locus and its Equation. Straight Line: Equation of straight line parallel to an Axis, slope form, intercept form, through two point condition of concurrency of three lines. Matrices and Determinants : Definition and Types of Matrices, Addition , Subtraction and Multiplication of a Matrices, Scalar Multiplication, Transpose of Matrix, Determinants, Determinants of square matrix of order I, 2 and 3, Area of a triangle, Solution of system of linear equations by Cramer's Rule, Minors and Cofactors, Adjoint of a Matrix, Inverse of a Matrix(up to order 3).	
Unit 4	**Differential Calculus:** Derivative of a Function, Various Formulae-Product and Quotient Rule of Differentiation, Differentiation of Function of Function(chain rule), Trigonometrical functions, Inverse Trigonometrical functions, Exponential function, Logarithmic function, Implicit functions, Logarithmic Differentiation, Differentiation of function with regard to another function, Higher Derivatives, Successive Differentiation, Liebnitz Theorem, Expansion of functions(up to 3 or 4 terms only) using Maclaurin's and Taylor's 1 theorem, Maxima and Minima (simple cases), Curve tracing (simple cases), Introduction to partial differentiation.	
Unit 5	**integral Calculus :** Anti-Derivatives, Constant of integration, Indefinite integral, Elementary Integration, formulae, Methods of Integration, Integration by Substitution, Integration by parts, integration through partial fractions and rationalization, Concept of	

	Definite integral, properties of definite integral, Integration using Gamma function. Area of Bounded Region, Circle, Parabola, Ellipse in standard form between two coordinates and x- axis.
Suggested online courses (MOOCs) 1. Basic Calculus 1 and 2 By Prof. Parasar Mohanty, IIT Kanpur https://onlinecourses.nptel.ac.in/noc21_ma20/preview	
This course can be opted as an elective by the students of following subjects: N.A.	
Suggested equivalent online courses (MOOCs) for credit transfer: N.A	

Programme: **BCA**	Year: **First**	Semester: **I**
Subject: Computer Science		
Course Code: **BCA-105P**	Course Title: **Data Structures and C Programming Lab**	

Course Objectives:
To enhance programming skills while improving their practical knowledge in data structures.
To strengthen the practical ability to apply suitable data structure for real time applications.

Course Outcomes:
CO1 Implement the abstract data type and reusability of a particular data structure.
CO2 Implement linear data structures such as stacks, queues using array and linked list.
CO3 Understand and implements non-linear data structures such as trees, graphs.
CO4 Implement various kinds of searching, sorting and traversal techniques and know when to choose which technique.

Credits: **04**	Type of Course: **Practical Lab**
Max. Marks: **100**	Min. Passing Marks: 36

List of Practical in Data Structures Lab with C:

1. Implementation of Stacks, Queues (using both arrays and linked lists).
2. Implement a program to evaluate a given postfix expression using stacks.
3. Implement the following operations on singly and circular linked list: i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Implementation of operations on binary tree (delete entire tree, copy entire tree, mirror image, level order, search for a node etc.)
5. Implementation of the following operations on binary search tree (BST): (a) Minimum key (b) Maximum key (c) Search for a given key (d) Delete a node with given key
6. Implementation of graph traversals by applying: (a) BFS (b) DFS
7. Implement the following algorithms to find out a minimum spanning tree of a simple connected undirected graph: (a) Prim‟s algorithm (b) Kruskal‟s algorithm
8. Implement Dijkstra‟s algorithm for solving single source shortest path problem.
9. Implementation of recursive and non-recursive functions to perform the following searching operations for a key value in a given list of integers: i) Linear search ii) Binary search
10. Implement the following sorting algorithms: a) Bubble sort b) Selection sort c) Insertion sort (d) Merge sort (e) Quick sort (f) Heap sort

Suggested Readings:

1. Virtual Lab on Data Structure: https://ds1-iiith.vlabs.ac.in/

Programme: **BCA**	Year: **First**	Semester: **II**
Subject: BCA		
Course Code: **BCA-106N**	Course Title: **Numerical Analysis**	
Course Objectives: The course provides students with an understanding to develop numerical methods for various mathematical problems and calculate the error involved in the numerical solution when compared to their exact solution. These techniques are useful to students while solving various interdisciplinary science problems. It contains solution of system of linear equations, roots of non-linear equations, interpolation, numerical differentiation and integration.		
Course Outcomes: **CO1** Understand basic methods and principles of scientific computing. **CO2** Solve basic and frequently occurring mathematical problems using computers and numerical software.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Solutions of Non-Linear Equations in one Variable**
Unit 1	Review of Calculus, Round off Error, Truncation Error, Some properties of equations, Iteration Methods for finding the roots (zero's) of an equation. Convergence Criterion, Initial Approximation to a Root, Bisection Method,
Unit 2	Fixed Point Iteration Method, Chord Methods for Finding Roots- Regula Falsi Method, Newton Raphson Method. Order of convergence.
Block 2	**Solution of System of Linear Equations**
Unit 3	Direct Methods- Preliminaries, Method of solution using inverse of matrix. Cramer's rule. Gauss Elimination Method, Gauss- Jordon Reduction Method, LU decomposition method. Crout's method.
Unit 4	Iterative Method- General Iteration Method, Jacobi's Iteration Method, Gauss- Seidal Iteration Method, Relaxation method.
Block 3	**Interpolation**
Unit 5	Definition, Finite Differences: Forward differences, Backward differences, Central differences, Other differences operator, Relation between operators. Interpolation at Equally interval; Newton Gregory formula for forward differences and backward difference.
Unit 6	Interpolation at Unequally interval Lagrange's interpolation formula. Divided differences, Properties of divided differences, Newton's Divided difference interpolation formula.
Block 4	**Numerical Differentiation, Integration and Solutions of Differentiation Equations**
Unit 7	Numerical Differentiation, Numerical Integration; Trapezoidal Rule. Simpson's One Third Rule, Simpson's Three Eight's Rule. Weddle's Rule.
Unit 8	Numerical Solution of Ordinary Differential Equations-(first order, second order and simultaneous) by Picard's Iteration Method, Euler's Method, Runge- Kutta Methods- 4^{th} Order.

Suggested Readings:

1. K. E. Atkinson, An Introduction to Numerical Analysis, 2nd Edition, John Wiley, 2008.
2. Numerical Analysis, R. L. Burden and J. D. Faires, 7th ed., Thomson Learning, 2001.

Suggested online courses (MOOCs)

1. NOC:Numerical methods, IIT Roorkee by Prof. Ameeya Kumar Nayak, Prof. Sanjeev Kumar https://nptel.ac.in/courses/111107105
2. NOC:Numerical Methods for Engineers, IIT Madras by Dr. Niket S.Kaisare https://nptel.ac.in/courses/127106019
3. Numerical Analysis By Prof. S. Baskar, IIT Bombay

https://onlinecourses.nptel.ac.in/noc23_ma44/preview
This course can be opted as an elective by the students of following subjects: **B.Sc.(Computer Science), M.Sc. (Statistics) and M.Sc. (Mathematics)**
Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **First**	Semester: **II**
Subject: BCA		
Course Code: **BCA-107N**	Course Title: **Multimedia Technology**	

Course Objectives: Today, Multimedia and web design technology play an essential role in education, agriculture, product launch, science and technology, corporate development and enhanced business opportunities. The increasing variety of hardware and software components in multimedia and website design has escalated the demand for human resources in these fields. This course is designed to inculcate required skills for these activities.

Course Outcomes:

CO1 Visualize scopes of multimedia and understand steps in creation of multimedia applications.

CO2 Understand digital audio, prepare audio required for a multimedia system and Speech synthesis and recognition concept.

CO3 Analyze representation of video, how video work and different video formats.

CO4 Describe different animation techniques and software used for animation.

CO5 Understand various multimedia development and authoring tools.

CO6 Know the different layers of network along with video conferencing technique.

Credits: **04**	Type of Course: **Core**
Max. Marks: **100**	Min. Passing Marks: **36**

Block 1	**Introduction to Multimedia and Its Components**
Unit 1	**Multimedia Technology:** Meaning & scope of Multimedia; Elements of Multimedia; Creating multimedia applications; Multimedia file & I/O functions; Multimedia data structures; Multimedia file formats; Multimedia Protocols
Unit 2	**Multimedia Audio:** Digital sound; Audio compression & decompression; Companding: ADPCM compression; MPEG audio compression; True Speech; Special effects and Digital Signal Processing: Audio synthesis; FM synthesis: Sound blaster card; Special effect processors on sound cards; Wave table synthesis; MIDI functions; Speech synthesis & Recognition
Unit 3	**Multimedia Video:** Representation of Digital video; Video capture: Frame grabbing; Full motion video; Live video in a window; Video processor; Video compression & decompression; Standards for video compression & decompression; Playback acceleration methods
BLOCK-2	**Multimedia Animation, Authoring Tools and Internet**
Unit 4	**Creating Multimedia Animation:** Icon animation; Bit-map animation; Real-time vs Frame by Frame animation; Object modeling in 3D animation; Motion control in 3D animation; Transparency; Texture. Shadows, Anti-aliasing; Human modeling & Animation; Automatic motion control
Unit 5	**Multimedia Authoring Tools:** Project editor; Topic editor; Hot-spot editor; Developing a multimedia title; Multimedia text authoring systems; Usage of authoring tools
Unit 6	**Multimedia on LANs & Internet:** Multimedia on LAN; Fast modems & Digital networks for multimedia; High speed digital networks; Video conferencing techniques; Multimedia interactive applications on Internet: Future Directions.

Suggested Readings:

1. "Li, Ze-Nian, Mark S. Drew, and Jiangchuan Liu. Fundamentals of multimedia. Upper Saddle River (NJ) Pearson Prentice Hall, 2004.
2. Jeffcoate, Judith. Multimedia in practice: technology and applications. Prentice-Hall, Inc., 1995.
3. Vaughan, Tay. Multimedia: Making it work. Tata McGraw-Hill Education, 2006.
4. Melliar-Smith, Peter Michael, and Louise E. Moser. "Multimedia Networking: Technology, Management and Applications. Hershey, PA Idea Group, 2002.

Suggested online courses (MOOCs)

1. Multimedia processing, IIT Kharagpur by Prof. Somnath Sengupta https://nptel.ac.in/courses/117105083 2. CIT-003: Web Based Technologies and Multimedia Applications By Prof. P. V. Suresh | Indira Gandhi National Open University https://onlinecourses.swayam2.ac.in/nou20_cs05/preview
This course can be opted as an elective by the students of following subjects: N.A.
Suggested equivalent online courses (MOOCs) for credit transfer: N.A.

Programme: **BCA**	Year: **First**	Semester: **II**
Subject: **BCA**		
Course Code: **BCA-108N**	Course Title: **Discrete Mathematics**	
Course Objectives: This course provides students understand discrete objects such as proofs, sets, graphs, colorings, algebraic structures and algorithms that arise naturally and frequently in many areas of mathematics and computer science. It develops a sound understanding of these discrete objects to solve problems arising in computer science.		
Course Outcomes: **CO1** Apply mathematical logic to solve problems. **CO2** Use logical notation to define and reason about fundamental mathematical concepts such as sets, relations, and functions. **CO2** Understand and apply counting techniques to the representation and characterization of relational concepts. **CO2** Impart foundations of probabilistic theory which is mostly used in varied applications in engineering and science.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Language of Mathematics and its application**
Unit 1	**Mathematical Logic:** statements, operations, truth values, tautology and quantifiers.
Unit 2	**Arguments:** Rule of Detachment, Validity of a compound statement by using Truth Table, Validity using Simplification Methods, Validity using Rules of Inference, Invalidity of an Argument, Indirect Method of proof and Proof by Counter-Example.
Unit 3	**Boolean Algebra:** Boolean Algbera, Principle of Duality, Isomorphic Boolean Algebras, Boolean Algebra as Lattices, Boolean Functions, Disjunctive Normal Form**,** Conjunctive Normal Form**,** Minimization of Boolean Functions (Karnaugh Map)
Unit 4	**Switching circuits and logical Circuits:** Switching Circuits, Simplification of circuit, Non-Series Parallel Circuits, Relay Circuits, Logic Circuits
Block 2	**Set theory and its application**
Unit 5	**Set theory:** sets, Subsets, Operations on Sets, Complementation, Intersection and Union, Laws Relating Operations, Distributive Laws and De Morgan's Laws.
Unit 6	**Relation:** Relation, binary relations in a Set, Domain and Range of a Relation, Total number of Distinct Relations, Relations as Sets of Ordered Pairs, Types of Relations, Composition of Relations, Equivalence relation in a set, Partition of a Set, Equivalence Class and Quotient set of a set.
Unit 7	**Partitions and Distributions:** Equivalence Relations, Equivalence Classes, Properties of Equivalence Classes, Quotient set and Partition.
Unit 8	**Function:** Functions, Direct and Inverse image, Inverse Functions, Operations on Functions, Composite of functions, Types of Functions and Connection between Equivalence relation and mapping.
Block 3	**Counting Process**
Unit 9	**Mathematical Induction:** Principle of Mathematical Induction, Second Principle of Induction and Well ordering property.
Unit 10	**Combinatorics:** Basic counting principles**,** Principle of Disjunctive counting, Principle of Sequential counting and Ordered and Unordered Partitions.
Unit 11	**Permutation**
Unit 12	**Combination**

Block 4	**Block – 04: Probability theory and application**
Unit 13	**Binomial theorem:** Binomial theorem, General term in a binomial expansion,Middle term in a binomial expansion and Binomial expansion for rational exponents.
Unit 14	**Probability:** Definition of Probability, Addition law for counting and Product law for counting.
Unit 15	**General Counting methods:** General Counting method is the extension part of counting process. It discusses Sum and Product Rules and the Pigeonhole Principle**.**
Unit 16	**The Inclusion- Exclusion Principle:** inclusion-exclusion principle, Alternative form of the inclusion-exclusion principle and Onto Functions.

Suggested Readings:

1. C.L.Liu and D.P.Mohapatra, " Elements of Discrete Mathematics: A Computer Oriented Approach", Mcgraw Hill, Third Edition, 2012.
2. Kenneth H. Rosen, "Discrete Mathematics and Its Applications" Mcgraw Hill, Seventh Edition, 2012 (Indian Adaptation by Kamala Krithivasan, Iit Madras).

Suggested online courses (MOOCs)

1. NOC:Discrete Mathematics, IIT Ropar, Prof. Prabuchandran K.J, Prof. Sudarshan Iyengar; https://nptel.ac.in/courses/106106183
2. NOC:Discrete Mathematics, IIT Guwahati, Prof. Benny George K, Prof. Sajith Gopalan https://nptel.ac.in/courses/106103205

This course can be opted as an elective by the students of following subjects: **B.Sc. in Computer Science, B.Sc. in Physics, B.Sc. in Statistics.**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A.

Programme: **BCA**	Year: **First**	Semester: **II**
Subject: BCA		
Course Code: **BCA-109N**	Course Title: **C++ and Object-oriented programming**	
Course Objectives: This course aims to offer a practical mastery of object-oriented concepts such as classes, objects, data abstraction, methods, method overloading, inheritance and polymorphism.		
Course Outcomes: **CO1** Develops a sound approach to problem solving using a middle level programming language. **CO2** Apply techniques like recursion and iteration are learnt to solve a problem. **CO3** Build programming concepts like pointers, structures.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**BLOCK - 1**
Unit 1	**Principles of object-oriented programming:** Object oriented programming paradigm, Comparison with procedural programming, Basic concepts of object-oriented programming, benefits of O0P, object-oriented Languages, advantage of C++.
Unit 2	**Object Orient Programming System:** Class, inheritance, abstraction, encapsulation and information hiding, polymorphism, overloading.
Unit 3	**Advanced concept**: Dynamism (Dynamic typing, dynamic binding, late binding, dynamic loading). Structuring programs, reusability, organizing object-oriented project,
Block 2	**BIOCK - 2**
Unit 4	**Overview of C++:** Tokens, keywords, identifiers and constants basic data types, user-defined and derived Data types, type compatibility, reference, variables type Casting, operator precedence, control structures, structure, function.
Unit 5	**Classes and objects:** Class specification, class objects, accessing class members, scope resolution operator, data hiding, empty classes, Pointers within a class, passing objects as arguments, returning objects from functions, friend Functions and friend classes, constant parameters and member functions, structures and Classes, static members.
Unit 6	**Object initialization and cleanup**: Constructors destructor, constructor overloading, order of construction and destruction, Constructors with default arguments, nameless objects, dynamic initialization through Constructors, constructors with dynamic operations, constant objects and constructor, static Data members with constructors and destructors, nested classes.
Block 3	**BLOCK - 3**
Unit 7	**Operator overloading and type conversion:** Defining operator overloading, overloading unary operators, overloading binary operators, overloading binary operators using friends, manipulation of strings using Operators, rules for overloading operators. type conversions.
Unit 8	**Inheritance: extending classes:** Deriving derived classes, single multilevel, multiple, hierarchical, hybrid inheritance, Constructors & destructors in derived classes, constructors invocation and data members Initialization, virtual base classes, abstract classes, delegation.
Block 4	**BLOCK- 4**
Unit 9	**Pointers, virtual functions and polymorphism:** Pointers to objects, this pointer. pointers to derived classes, virtual functions, Implementation of run-time polymorphism, pure virtual functions.
Unit 10	**Working with files:** Classes for file stream operations. opening and closing a file, file pointers and their Manipulations, sequential input and output operations, error handling during file Operations, command line arguments.

Unit 11	**Object Oriented Modeling:** Need of object-oriented Modeling, Simulation of real-life problems using OOP concept: Example, Representation of problem using object and class diagrams at design level.

Suggested Readings:

1. The C++ Programming Language by Bjarne Stroustrup, 2013.
2. Programming: Principles and Practice Using C++ by Bjarne Stroustrup, 2014
3. Oriented Object-Oriented Programming with C++ by Balaguruswamy, TMH

Suggested online courses (MOOCs)

1. NOC:An Introduction to Programming Through C++, IIT Bombay by Prof. Abhiram G Ranade https://nptel.ac.in/courses/106101208
2. Programming in Modern C++, IIT Kharagpur By Prof. Partha Pratim Das https://onlinecourses.nptel.ac.in/noc23_cs50/preview

This course can be opted as an elective by the students of following subjects: N.A.

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **First**	Semester: **I**
Subject: BCA		
Course Code: **BCA-110N**	Course Title: **Practical Lab Based on Numerical Analysis and OOP with C++**	

Course Objectives:

- To enhance programming skills while improving their practical knowledge in data structures.
- To strengthen the practical ability to apply suitable data structure for real time applications.

Course Outcomes:

CO1 Implement the abstract data type and reusability of a particular data structure.
CO2 Implement linear data structures such as stacks, queues using array and linked list.
CO3 Understand and implements non-linear data structures such as trees, graphs.
CO4 Implement various kinds of searching, sorting and traversal techniques and know when to choose which technique.

Credits: **04**	Type of Course: **Practical Lab**
Max. Marks: **100**	Min. Passing Marks: 36

List of practical in numerical methods and OOP with C++:

1. Find the roots of non-linear equation using bisection method in C++.
2. Find the roots of non-linear equation using newton's method in C++.
3. Solve the system of linear equations using gauss - elimination method in C++.
4. Solve the system of linear equations using gauss -jorden method in C++.
5. Integrate numerically using trapezoidal rule in C++.
6. Integrate numerically using simpson's rules in C++.
7. Find numerical solution of ordinary differential equations by runge- kutta method in C++.
8. Write a C++ program to illustrate the concept of class with method overloading.
9. Write a C++ Program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use String Tokenizer class of java.util)
10. Write a C++ program to illustrate the concept of Single level and Multi level Inheritance.
11. Write a C++ program to demonstrate the Interfaces & Abstract Classes.
12. Write a C++ program to implement the concept of exception handling.

Suggested Readings: N.A

Programme: **BCA**	Year: **Second**	Semester: **III**
Subject: BCA		
Course Code: **BCA-111N**	Course Title: **Data Base Management System**	

Course Objectives: Today databases form the backbone of all major applications – internet, banking, product & sales etc. Relational Database Management Systems (DBMS) have long formed the basis for many leading databases such as Oracle, Microsoft SQL Server and MySQL. This course aim to provide a common set of models and design paradigms which includes:

- Data models, conceptualize and depict a database system using ER diagram.
- Internal storage structures in a physical DB design.
- Database normalization technique that organizes the data within a database in the most efficient manner possible.
- Fundamental concepts of transaction processing techniques**.**

Course Outcomes:

CO1 Students can explain the role of a database management system, basic database concepts, including the structure and operation of the relational data model.

CO2 Apply logical database design principles, including E-R/EE-R diagrams, conversion of ER diagrams to relations.

CO3 Describe the concepts of integrity constraints, relational algebra, relational domain & tuple calculus, data normalization.

CO4 Construct simple and moderately advanced database queries using Structured Query Language (SQL).

CO5 Understand and apply Database Normalization to remove the duplicate data and database anomalies from the relational table

CO6 Understand the concept of a database transaction including concurrency control, backup and recovery.

Credits: **04**	Type of Course: **Core**
Max. Marks: **100**	Min. Passing Marks: **36**

Block 1	**Basic concepts of DBMS**
Unit 1	**Introduction: Database** Management System, Examples, Characteristics of the Database Approach, Advantage of using a Database Approach. Database System concepts and Architecture, Data Models, Schemes and Instances, DBMS Architecture and Data independence, Database Languages, Procedural and Non-procedural languages and Interfaces. Database System Environment, Classification of Database Management Systems.
Unit 2	**ER Model:** Database Modeling using the ER Model., Using High-Level conceptual Data Models for Database design, an example Database Application, Entity types, Entity Sets, Attributes and keys, Relationships, Relationship types, roles and Structural Constraints., Week Entity types, Refining the ER Design for the Company Database, ER Diagrams, naming conventions and design Issues, Conversion of ER Diagram to tables.
Unit 3	**Relational Data Model:** Basic Relational data model Concepts, Relational Databases and Relational Database Schemas, Relational Model Constraints, update Operations and Dealing with Constraint Violations
Block 2	**Query Language and Database Design Concepts**
Unit 4	**Relational Algebra:** Relational Model Concepts, Relational concepts and Relational Database Schemes, Update Operation and Dealing with Constraints Violations, Relational Database Design, Using ER-to-Relational Mapping.
Unit 5	**Structured Query language:** Data definition, Constraints and Schema changes in SQL 2, Basic Quires in SQL, More Complex SQL Quires, Insert, Delete and Update Statements in SQL, views (Virtual Tables) in SQL, Specifying general constraints as Assertion features of SQL. Integrity constraints, Triggers, Functional dependencies.

Unit 6	**Functional Dependency Theory: Functional** Dependencies and Normalization for Relational Database, Informal Design Guidelines for Schemes, Functional Dependencies.
Unit 7	**Normalization:** Normal Forms based on Primary keys, General Definitions of Second and Third Normal forms, Boyce Codd Normal form, Relational Database Design Algorithms and Further Dependencies, Algorithms for Relational Database Schema Design, Multi-valued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form, Inclusion Dependencies, Other Dependencies and Normal Forms.
Block 3	**Transaction Management & Emerging Databases**
Unit 8	**Transaction Processing Concepts**: Introduction to Transaction Processing, Transaction and System Concept, Desirable properties of Transactions, Scheduling and Recoverability, Serializability of Scheduling, Transaction Support in SQL, Concurrency control techniques, Concurrency techniques for concurrency control, concurrency control based on timestamp based protocol, validation based protocol, deadlock handling, Database Recovery Techniques based on Immediate Update, Failure classification, Shadow Paging, Log based recovery, failure with loss of Nonvolatile Storage.
Unit 9	Emerging Trends in DBMS: Emerging Trends in DBMS: Introduction to object-oriented Database Management System, Introduction to client/Server Database, Introduction to Distributed Database, Introduction to Knowledge Databases.

Suggested Readings:

1. R Elmasri, S Navathe, Fundamentals of Database Systems, 6th edition, Addison-Wesley, 2010.
2. R Ramakrishnan, J Gehrke, Database Management Systems, 3rd Ed., McGraw-Hill, 2002.
3. A Silberschatz, H Korth and S Sudarshan, Database System Concepts, 6th Ed., McGraw-Hill, 2010.

Suggested online courses (MOOCs)

1. NOC: Data Base Management System, IIT Kharagpur by Prof. Partha Pratim Das Prof. Samiran Chattopadhyay Prof. Kausik Datta
   https://nptel.ac.in/courses/106105175
2. NOC:Introduction to Database Systems, IIT Madras by Prof. P.Sreenivasa Kumar
   https://nptel.ac.in/courses/106106220
3. NOC:Fundamentals of Database Systems (Course sponsored by Aricent), IIT Kanpur By Dr. Arnab Bhattacharya
   https://nptel.ac.in/courses/106104135

This course can be opted as an elective by the students of following subjects: **B.Sc. in Computer Science, MCA**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Second**	Semester: **III**
Subject: BCA		
Course Code: **BCA-112N**	Course Title: **Operating System**	
Course Objectives: The course will introduce Operating Systems (OS), their design and implementation. We will discuss the goals of an OS and some successful and not-so-successful OS designs. We will also discuss the following OS services in detail: thread scheduling, security, process management, memory management, virtual memory, and disk scheduling.		
Course Outcomes: **CO1** Analyze & classify different types of operating system **CO2** Understand the working of Operating system **CO3** Interpret concepts of thread scheduling, process management, memory management, virtual memory, and disk scheduling.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**An Overview and Process Management**
Unit 1	**Introduction:** Basic definitions, Batch processing, Multi-programming. Time sharing, multiprocessing; Structure and Functions of Operating System
Unit 2	**Process and thread:** Process, Process states, State Transitions, Process Control Block, Context Switching, concept of thread, comparison between process and thread, Thread model, thread usage, implementing thread in kernel and user space.
Unit 3	**Process Scheduling:** Scheduler, Scheduling criteria, Preemptive and non-preemptive scheduling, Process Scheduling, Process scheduling algorithms.
Unit 4	**Concurrent Process:** Process Interaction, Shared Data and Critical Section, Mutual Exclusion, Synchronization, Classical Problems of Synchronization, Semaphores, Monitors.
Block 2	**Memory Management and Unix Case Study**
Unit 5	**UNIT 5: Deadlock:** Concept of deadlock, necessary condition for deadlock, resource allocation graph, deadlock prevention, deadlock avoidance, Banker's algorithm, Deadlock detection, deadlock recovery.
Unit 6	**UNIT 6: Memory management:** Address Binding, Dynamic Loading and Linking Concepts, Logical and Physical Addresses Contiguous and non-contiguous memory allocation, Paging, Segmentation, Virtual Memory, Demand Paging, Page fault,Page replacement algorithms, thrashing.
Unit 7	**UNIT 7: Secondary memory management:** Free Space management, Disk Structure, Disk Scheduling, Formatting, Swap space Management.
Unit 8	**UNIT 8: Case Study of UNIX**

Suggested Readings:

1. Silberschatz, Galvin, Gagne, Operating System Concepts, 8th Edition, Wiley,2008
2. Andrew S. Tanenbaum, Albert S. Woodhull, Operating Systems, Design and Implementation, 3rd Edition, Prentice Hall,2006.
3. William Stallings, Operating Systems: Internals and Design Principles, 6th Edition, Prentice Hall, 2013.
4. Charles Patrick Crowley, Operating Systems-A Design-oriented Approach. 1996

Suggested online courses (MOOCs)

1. NOC:Operating System Fundamentals, IIT Kharagpur by Prof. Santanu Chattopadhyay https://nptel.ac.in/courses/106105214
2. NOC:Introduction to Operating Systems, IIT Madras by Prof. Chester Rebeiro https://nptel.ac.in/courses/106106144
3. Operating Systems, IIT Delhi by Prof. Sorav Bansal

https://nptel.ac.in/courses/106102132
This course can be opted as an elective by the students of following subjects: **B.Sc. (Computer Science) and MCA**
Suggested equivalent online courses (MOOCs) for credit transfer: N.A.

Programme: **BCA**	Year: **Second**	Semester: **III**
Subject: BCA		
Course Code: **BCA-113N**	Course Title: **Software Engineering**	
Course Objectives: Provide the current software engineering techniques and examine the software life-cycle, including software specification, design implementation, testing and maintenance. It presents software engineering methodologies for the development of Quality, cost-effective, schedule meeting software.		
Course Outcomes: **CO1** Describe software engineering layered technology and process framework. **CO2** Introduces theories, models, and techniques that provide a basis for the software development life cycle. **CO3** Introduces software testing approaches including verification and validation, static analysis, reviews, inspections, and audits. **CO4** Understanding of the role of project management including planning, scheduling, risk management, etc. **CO5** Work as an individual and/or in team to develop and deliver quality software.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	
Unit 1	**Software Engineering Fundamentals:** Definition of Software, Software characteristics, Software Applications. Software Process: Software Process Models - Waterfall model, prototyping model, spiral model, incremental model, concurrent development model. Project management Concepts: The Management Spectrum - The People, The Product, The Process, The Project.	
Unit 2	**Software Process and Project Metrics :** Measures , Metrics and Indicators , Software measurement Size -Oriented Metrics , Function - Oriented Metrics , Extended Function point metrics Software Project Planning : Project Planning Objectives , Software Project Estimation , Decomposition Techniques - Problem Based Estimation Process Based Estimation ,Empirical Estimation Models- The COCOMO Model Risk Analysis and Management: Software risks, Risk identification, Risk Projection, Risk Refinement, Risk Mitigation , Monitoring and Management.	
Unit 3	**Software Quality Assurance:** Basic concepts- Quality, Quality Control, Quality Assurance, Cost of Quality, Software Quality Assurance (SQA), Formal Technical Review Software Configuration Management: Baselines, Software Configuration Items, The SCM Process, Version Control, Change Control, Configuration Audit, Status Reporting. Analysis Concepts and Principles: Requirements Elicitation for Software, Analysis Principles. The Information Domain, Modeling, Partitioning, Essential and Implementation Views, Specification: Specification Principles, Representation, The Software Requirement Specification (SRS)	
Unit 4	**Design Concepts and Principles:** Design Principles, Design Concepts — Abstraction, Refinement, Modularity, Software Architecture, Control Hierarchy, Structural Partitioning, Data Structure. Software Procedure, Structure, Information Hiding, Effective Modular Design- Cohesion, Coupling Software Testing: Testing Objectives & principles, Unit Testing, Integration Testing (Top-Down Integration, Bottom. Up Integration, Regression Testing, Smoke Testing), Validation Testing (Alpha and Beta Testing), System Testing (Recovery Testing, Security Testing, Stress Testing, Performance Testing).	
Unit 5	**Reengineering:** Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering CASE Tools: What is CASE, Building Blocks of CASE, A Taxonomy of CASE Tools, Integrated CASE Environments, The integration Architecture, The CASE Repository.	

Suggested Readings:

1. Mall, Rajib. Fundamentals of software engineering. PHI Learning Pvt. Ltd., 2018.
2. R.S. Pressman, Software Engineering – A Practitioner's Approach, 6th Edition, TMH, 2013.
3. Ian Sommerville, Software Engineering, 8th Edition, Addison Wesley, 2009.
4. Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Publishing, 2010.

Suggested online courses (MOOCs)

1. NOC:Software Engineering, IIT Kharagpur by Prof. Rajib Mall https://nptel.ac.in/courses/106105182
2. Software Engineering, IIT Bombay by Prof. Rushikesh K Joshi, Prof. Umesh Bellur, Prof. N.L. Sarda https://nptel.ac.in/courses/106101061

This course can be opted as an elective by the students of following subjects: N.A.

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Third**	Semester: **III**
Subject: BCA		
Course Code: **BCA-114N**	Course Title: **Principle of Programming Language**	
Course Objectives: This course gives an understanding of the evolution of programming languages. It explains the underlying concepts of object-oriented languages, functional languages, logical and scripting languages. The course offers students the principles and techniques involved in the design of modern programming languages.		
Course Outcomes: **CO1** Inculcate notations to describe syntax and semantics of programming languages. **CO2** Analyze semantic issues associated with function implementations, including variable binding, scoping rules, parameter passing, and exception handling.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**BLOCK -1: Programming Languages-1**
Unit 1	Programming Languages Fundamental: Introduction, Programming Language Introduction, Importance of programming , languages , Brief history, Features
Unit 2	Language Translator: Introduction, Attributes of good programming language, Introduction to language translator
Unit 3	Data Types (Elementary And Structured): Introduction, Binding and binding time, Elementary and structured data types, Specifications
Unit 4	Representations And Implementation Of Numbers: Introduction, Representations and Implementation of numbers
Block 2	**Programming Languages-2**
Unit 5	Variable Size Data Structure: Introduction, Vectors , Arrays , Records , Character string, Variable size data structure, Sets
Unit 6	Encapsulation: Introduction, Input files, Encapsulation , Information hiding, Sub programs.
Unit 7	Data Types & Sequence Control: Introduction, Type definition , Data Types , Abstract data types, Sequence control, Explicit and Implicit Sequence Control
Unit 8	Exception Handlers & Co-Routines: Introduction, Subprogram sequence control, Recursive sub-programs , Exception and exception handlers, Co-routines , Scheduled subprograms.
Block 3	**Programming Languages-3**
Unit 9	Task & Exception: Introduction, Tasks and Exceptions, Concurrency and Exceptions, Referencing Environments
Unit 10	Structures (Static, Dynamic & Block): Introduction, Static and dynamic structures, block structure
Unit 11	Local Referencing Environments: Introduction, Local data & local referencing environments
Unit 12	Scope of Shared Data: Introduction, Dynamic and Static scope of shared data, Types of Scopes
Block 4	**Programming Languages-4**
Unit 13	Parameter & Their Transmission: Introduction, Block structure, Parameters and their transmission
Unit 14	Task And Shared Data Storage: Introduction, Task and shard data storage

Unit 15	System Controlled Storage Management: Introduction, Program and system controlled storage management, Storage Management Phases
Unit 16	Storage Management: Introduction, Static based storage management, Stack based storage management, Fixed size heap storage management, Variable size heap storage management

Suggested Readings:

1. Programming Language Pragmatics. Michael Scott, Morgan Kaufmann, 2000.
2. Essentials of Programming Languages. Friedman, Wand and Haynes, Prentice-Hall International (PHI), 1998.
3. Principles of Programming Languages. Tennant. PHI, 1981.

Suggested online courses (MOOCs)

1. P Principles of Programming Languages, IIT Delhi by Prof. S. Arun Kumar https://nptel.ac.in/courses/106102067

This course can be opted as an elective by the students of following subjects: **B.Sc.(Computer Science)**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Second**	Semester: **III**
Subject: Computer Science		
Course Code: **BCA-115P**	Course Title: **DBMS and OS Lab**	

Course Objectives:

- Provide working on existing database systems, designing of database, creating relational database, analysis of table design.
- Practice various DDL commands in SQL
- Write simple and complex queries in SQL
- Familiarize PL/SQL

Course Outcomes:

CO1 Design and implement a database schema for a given problem

CO2 Populate and query a database using SQL and PL/SQL

Credits: **04**	Type of Course: **Practical Lab**
Max. Marks: **100**	Min. Passing Marks: 36

List of Practical in Database Management Systems Lab:

1. Creation of a database (exercising the commands for creation)
2. Simple to complex condition query creation using SQL Plus.
3. Implementation of DDL commands of SQL with suitable examples: Create table, Alter table and Drop Table
4. Implementation of DML commands of SQL with suitable examples: Insert, Update and Delete
5. Implementation of different types of function with suitable examples: Number function, Aggregate Function, Character Function, Conversion Function and Date Function
6. Implementation of different types of operators in SQL: Arithmetic Operators, Logical Operators, Comparison Operator, Special Operator and Set Operation.
7. Implementation of different types of Joins: Inner Join, Outer Join and Natural Join etc.
8. Study and Implementation of Group By, having clause, Order by clause and Indexing.
9. Implementation of Sub queries and Views.
10. Usage of triggers and stored procedures.
11. Writing PL/SQL procedures for data validation.

List of Practical in Operating System Lab:

1. Demonstrate basic UNIX shell commands with UNIX shell programs
2. Given a list of processes with their CPU burst times and arrival times, display the Gantt chart and compute average waiting time and average turnaround time for each of the following scheduling policies with C language.
   a) FCFS b) SJF c) Round Robin d) Priority.
3. Simulate the following page replacement algorithms in C Language a) FIFO b) LRU c) OPTIMAL
4. Simulate following disk scheduling algorithms with C language a) FCFS b) SCAN c) C-SCAN

Suggested Readings:

1. https://www.cdlsiet.ac.in/wp-content/uploads/2022/03/DBMS-LAB-MANUAL.pdf
2. https://mrcet.com/pdf/Lab%20Manuals/CSE%20II-II%20SEM.pdf

Programme: **BCA**	Year: **Second**	Semester: **IV**
Subject: BCA		
Course Code: **BCA-116N**	Course Title: **Computer Networks**	
Course Objectives: This course offers students an understanding of how machines are connected in a network and how data communication takes place between machines at various locations. It provides basic concepts of data communication, layered model, protocols and interworking between computer networks and switching components in telecommunication systems.		
Course Outcomes: **CO1** Understand basics of computer networks and various network topologies. **CO2** Explain basics of OSI Reference Model and TCP/IP Model. **CO3** Understand various protocol of data link layer for flow and error control such as Stop and wait protocols, One bit sliding window protocol, Using Go-Back N. **CO4** Describe different types of network devices Hub, Bridges, Switch, Gateways, and Routers along with their working. **CO5** Aware of different types of IP addresses classes and the need of subnetting. **CO6** Realize how packet is being transferred from source to destination PC. **CO7** Examine different types of routing protocols, flow control, error control and congestion control algorithms in network and transport layer.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	Computer Network Basics and Services
Unit 1	Introduction to Computer Network: Computer networks, Network Hardware—Local Area networks, Metropolitan Area networks,,Wide Area networks, Wireless networks, Internetworks, Network Software: Protocol Hierarchies,
Unit 2	OSI and TCP/IP Model: Design and Issue for layers, Interfaces and services, Connection oriented and Connection less Services. OSI reference model, and its Evolution, TCP/IP model.
Unit 3	The Physical Layer: Physical Layer, Transmission media, twisted pair, Base band and Broadband coaxial cable, Fiber optics, unguided media.
Unit 4	ISDN and Switching Techniques: MODEM, ISDN services, Switching Message, Packet Circuit switching TDM, and FDM, ATM, X.25.
Block 2	Link Layer Issues and Access Protocols
Unit 5	Unit 5: Data Link Layer: Data Link Layer, Error detection and Correction, Protocols: Simplex Stop and wait protocols, One bit sliding window protocol, Using Go-Back N. Flow control, Sliding Window Protocol, Channel Allocation Problem,
Unit 6	Unit 6: Multiple Access Protocol: ALOHA, CSMA protocol, Collision Free protocol, Polling, FDM, TDM,
Unit 7	Unit 7: The Medium Access Sub Layer: Framing, Static and Dynamic Channel Allocation in LANs and MANs, IEEE Standard 802.3, and Ethernet IEEE standard 802.4 and token Ring, IEEE Standard 802.5, Token Bus
Unit 8	Network devices: Hub, Bridges, Switch, Gateways, Routers.
Block 3	IP Addressing and Routing Issues
Unit 9	IP Protocol and Addressing: Network layer design issue, IP Protocol, IP Addresses, subnets,
Unit 10	Connection Management: Internetworking, connection less and connection oriented services, tunneling, Fragmentation, Firewall, Internet Controls Protocols.

Unit 11	Routing in Network Layer: Routing Algorithm, shortest path routing, Flooding, Flow-based routing, Broadcast routing, Congestion Control Algorithm, Congestion control and prevention policies;
Block 4	Transport, Session, Presentation and Application Layer
Unit 12	Transport layer: Transport layer connection management, flow control, error control, congestion control, Establishing and releasing a connection, TCP service Model, TCP protocol
Unit 13	Session and Presentation Layer: Introduction to cryptography and data compression
Unit 14	The Application Layer: Network Security, Domain Name System, Email: Architecture and Services, Message formats, Message transfer.

Suggested Readings:

1. HBehrouz A. Forouzan, Data Communications and Networking, McGraw Hill , 2006
2. A.S. Tanenbaum, Computer Networks, PHI , 2002

Suggested online courses (MOOCs)

1. Data Communication, IIT Kharagpur by Prof. Ajit Pal
   https://nptel.ac.in/courses/106105082
2. NOC:Computer Networks and Internet Protocol, IIT Kharagpur by Prof. Soumya Kanti Ghosh, Prof. Sandip Chakraborty
   https://nptel.ac.in/courses/106105183
3. NOC:Advanced Computer Networks, IIT Indore, IIT Gandhi nagar by Prof. Neminath Hubballi, Prof. Sameer Kulkarni
   https://nptel.ac.in/courses/106106243

This course can be opted as an elective by the students of following subjects: **B.Sc. in computer science, MCA**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Second**	Semester: **IV**
Subject: BCA		
Course Code: **BCA-117N**	Course Title: **Java Programming**	
Course Objectives: This course aims to cover the essential topics of Java programming so that students can improve their skills to cope with the current demand of IT industries and solve many problems in their field of study.		
Course Outcomes: **CO1** Use the characteristics of an object-oriented programming language JAVA in a program. **CO2** Apply JAVA features to program design and implementation. **CO3** Design and implementation programs of Java Script, Applets, Event Handling, AWT Programming, and Interface. **CO4** Implementation of Packages, Swing, and Servlet. **CO5** Design and implementation programs of JSP.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Object Oriented Methodology and Java**
Unit 1	**Object Oriented Programming:** Paradigms of Programming languages, Evolution of Object-Oriented Methodology, Basic Concepts of OO Approach, Comparison of object oriented and procedure - oriented Approaches, Benefits of OOPS, Applications of OOPS. Classes and objects, Abstraction and Encapsulation, Inheritance, Method overriding and Polymorphism.
Unit 2	**Java Language Basics**: Introduction to Java, Primitive Data Type and Variables, Java Operators.
Unit 3	**Expressions Statements and Arrays:** Expressions, Statements, Control Statements, Selection Statements, Iterative Statements, Jump statements, Arrays.
Block 2	**Object oriented concepts and Exceptions Handling**
Unit 4	**Class and objects:** Class Fundamentals, Introducing Methods, this Keyword, Using objects as Parameters, Method overloading, Garbage collection, the ffinalize () Method.
Unit 5	**Inheritance and Polymorphism:** Inheritance Basics, Access, Multilevel, inheritance, Method overriding Abstract classes, Polymorphism, Final Keyword.
Unit 6	**Packages and interfaces:** Package, Accessibility of Packages, using Package members, Interfaces, Implementing interfaces, interface and Abstract classes, Extends and Implements together.
Unit 7	**Exceptions Handling:** Exception, Handling of Exception, Types of Exceptions, Throwing, Exceptions, writing Exception subclasses.
Block 3	**Multithreading, I/O, and Strings Handling**
Unit 8	**Multithreaded Programming:** Multithreading, The Main thread, JAVA Thread Model, Thread Priorities, Synchronization in JAVA, Inter thread Communication.
Unit 9	**I/O In Java:** I/O Basics, Streams and stream, Classes, the predefined streams, Reading from and writing to console, reading and writing files, the transient and volatile Modifiers, using instance of Native Methods.
Unit 10	**Strings and Characters:** Fundamental of Characters and Strings, the String class, String operations, Data Conversion using value of () Methods, Strings Buffer and Methods.
Unit 11	**Exploring Java I/O:** Java I/O classes and interfaces, Stream classes, Text streams, Stream Tokenizer, Serialization, Buffered stream, print stream, Random Access file.
Block 4	**Graphics and user interfaces**

Unit 12	**Applets:** The applet class, Applet architecture, An applet Skeleton: Initialization and Termination, Handling events, HTML Applet TAG.
Unit 13	**Graphics and user interfaces:** Graphics contests and Graphics objects, user interface components, Building user interface with AWT, Swing - Based GUI, Layouts and layouts and layout Manager, Container.
Unit 14	**Networking Features:** Socket overview, reserved parts and proxy servers, Internet Addressing: Domain Naming Services (DNS), Java and The Net: URL, TCP/IP Sockets, Datagrams.

Suggested Readings:

1. Java: The Complete Reference Hebert Schildt, Mc Graw Hill
2. Object-Oriented Programming with C++ and Java Debasis Samanta, Prentice Hall India.

Suggested online courses (MOOCs)

1. NOC:Programming in Java, IIT Kharagpur by Prof. Debasis Samanta: https://nptel.ac.in/courses/106105191

This course can be opted as an elective by the students of following subjects: **B.Sc. in computer science**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Second**	Semester: **IV**
Subject: BCA		
Course Code: **BCA-118N**	Course Title: **Windows Programming**	
Course Objectives: This course intended to provide Windows programming and its associated concepts like traditional programming, programming resources, and visual C++ programming. It discusses about the Visual Basic programming concepts including the controls available in control/tool box and other custom controls. It imparts understanding of the document view architecture concepts using MFC, Single Document Interface (SDI), Multiple Document Interface (MDI), Database Management System (DBMS), Network programming concepts, ActiveX Controls, COM, DCOM, and COM+.		
Course Outcomes: **CO1** Understand basics of visual basic and its various components. **CO2** Apply event driven model and object oriented methodology. **CO3** Use basic programming skills using GUI interfaces to develop various applications.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Introduction to Windows Programming**
Unit 1	Windows Programming, Traditional Programming Paradigms, Event Driven Programming, Handles and Data Types, Windows Messages, Device Contexts, Document Interfaces, Dynamic Linking Libraries, Software Development Kit Tools, Context Help
Unit 2	Programming Resources: Accelerators, Bitmaps, Dialog Boxes, Icons, Menus, String Tables, Toolbars.
Unit 3	Visual C++ Programming: Basic Concepts of VC++, Object Oriented Programming, Objects and Classes, VC++ Components, Resources, Event Handling, Menus, Dialog Boxes, MFC File Handling, MFC and VC++
Block 2	**Visual Basic Programming**
Unit 4	Visul Basic Programming: Introduction to Visual Basic, Important Windows, Variables, Data Types, Decision Making, Operators, Loops, Procedures in Visual Basic, Visual Basic Code Module
Unit 5	Working with Controls: What is Control, What is Custom Control, Control Properties, The Intrinsic Controls, RichTextBox Controls, Working with Menu Items, Adding and Removing Control, Naming a Control
Unit 6	Dialog Boxes and Internet: Introduction to Dialog Boxes, Modal Dialog Box, Modeless Dialog Box, Modal Vs Modeless Dialog Box, Common Dialog Box, Visual Basic and Internet
Block 3	**Working with Graphics**
Unit 7	Document View Architecture: Microsoft Foundation Class, View Document Architecture Using MFC, Serialization, Separating documents from view, Visual C++ Resources, Application Wizard, Accelerators, Menus, Toolbars
Unit 8	Graphics and Multimedia: Working with Graphics, Consoles, Multitasking Process and Threads, Drawing Graphics in Windows, Clipboards, Printing Graphics and Text, Creating Animations with Picture Clip control
Block 4	**Block 4 Interfacing and Database Application**
Unit 9	Interfacing Other Applications: Single Document Interface (SDI), Multiple Document Interface (MDI), Difference between SDI & MDI, Explorer Style-Interface, Splitter Windows, Exception Handling, Debugging, Object Linking and Embedding (OLE)
Unit 10	Database Application: History of DBMS, Introduction to DBMS, DBMS Architecture, Components of DBMS, Need of DBMS, Advantages of DBMS, Disadvantages of DBMS,

	Database Administrator (DBA), Open Database Connectivity (ODBC), Database Access, Structured Query Language (SQL), Database Access with Data Control, Recordset, Applications of DBMS
Unit 11	Network Programming: Introduction to Winsock, Windows Socket in General, Creating Sockets, Miscellaneous API, Winsock Catalog, Windows Objects, Access Control Story, Security Descriptors
Unit 12	Advance Topics and Case Study: ActiveX Control, Component Object Model (COM), COM+, Distributed Component Object Model (DCOM), Application using Visual Basic, Example - Customer Database Input Screen

Suggested Readings:

1. Starting Out with Visual Basic, 7th Edition. Gaddis. Addison-Wesley. ISBN: 978-0134400150.
2. Programming Windows with MFC, Second Edition, by Jeff Prosise
3. Wiley India VB.Net Step By Step, Michael Halvorson, PHI.

This course can be opted as an elective by the students of following subjects: **B.Sc.(Computer Science), M.Sc. (Statistics) and M.Sc. (Mathematics)**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Second**	Semester: **IV**
Subject: BCA		
Course Code: **BCA-119N**	Course Title: **Computer Organization**	
Course Objectives: The course aims to provide an understanding of the basic structure of a digital computer and to study the operations of internal components.		
Course Outcomes: **CO1** Assess basics components of computer hardware. **CO2** Understand how Boolean algebra is related to designing computer logic, through simple combinational and sequential logic circuits. **CO3** Realize a simple computer with hardware design including data format, instruction format, instruction set, addressing modes, bus structure, input/output, memory, Arithmetic/Logic unit, control unit, and data, instruction and address flow. **CO4** Design combinational and sequential logic circuits, flip-flops, counters, shift registers, adders, substractor, multiplexer, demultiplexer, Arithmetic/Logic unit. **CO5** Develop concept of memory unit and input/output architecture. **CO6** Build basics of Instruction Set Architecture (ISA).		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Introduction to Digital Electronics**
Unit 1	**Introduction to number system:** binary, octal, hexadecimal, Inter-conversion to different number system.
Unit 2	**Boolean algebra and Logic Gates:** De Morgan's theorem, Boolean Identity. OR, AND NOT NAND, NOR and Ex OR gates and their Truth Tables, Positive and Negative logic.
Unit 3	**Reduction Techniques:** Standard representation of Boolean expressions, SOP and POS forms, Combinational and sequential circuits, Minterm and Maxterm expressions, Map reduction techniques, K- tap. Code Conversions: Binary to Gray, BCD to decimal etc.
Unit 4	**Binary Arithmetic:** Half and Full Adder, Substractor, Multiplexer, Demultiplxer, Decoder, Encoders, Comparators.
Unit 5	**Sequential Circuit:** Flip Flops: S/R, J/K, D and T Latches, Digital Counters, Registers.
Block 2	**Basic building blocks**
Unit 6	**Building blocks:** I/O, Memory, ALU and its components, Control Unit and its functions
Unit 7	**Instruction** — word, Instruction and Execution cycle, branch, skip, jump and shift instruction, Operation of control. registers; Controlling of arithmetic operation.
Unit 8	**Addressing techniques —** Direct, Indirect, Immediate, Relative, Indexed addressing and paging. Registers —Indexed, General purpose, Special purpose, overflow, carry, shift, scratch, Memory Buffer register; accumulators; stack pointers; floating point; status information and buffer registers.
Block 3	**Memory & I/O**
Unit 9	**Memory**: Main memory, RAM, static and dynamic, ROM, EPROM, EEPROM, EAROM, Cache and Virtual memory.
Unit 10	**I/O System:** Buses, Interfacing buses, Bus formats- address, data and control, Interfacing keyboard, display, auxiliary storage devices and printers.
Unit 11	**Introduction to Microprocessors and microcontrollers**; Introduction to 8085 microprocessor, example of few instructions to understand addressing techniques,

	differences between microprocessors and microcontrollers. Interlocution to different processor families.

Suggested Readings:

1. William Stallings, "Computer Organization and Architecture", 9th Edition, PHI,2012
2. M. Morris Mano, Michael D. Ciletti, "Digital Design", 4th Edition, Pearson Education, 2011.
3. Hennessy J. and Patterson D., "Computer Architecture: A Quantitative Approach", 5th Edition, Morgan Kaufmann, 2011.

Suggested online courses (MOOCs)

1. Digital Computer Organization, IIT Kharagpur by Prof. P.K. Biswas
   https://nptel.ac.in/courses/117105078
2. NOC:Computer architecture and organization, IIT Kharagpur by Prof. Indranil Sengupta, Prof. Kamalika Datta
   https://nptel.ac.in/courses/106105163
3. NOC:Computer Organization and Architecture, IIT Madrasby Prof. V. Kamakoti
   https://nptel.ac.in/courses/106106166
4. Computer Organisation and Architecture, IIT Kanpurby Prof. Bhaskaran Raman
   https://nptel.ac.in/courses/106104073

This course can be opted as an elective by the students of following subjects: **B.Sc. in computer science**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Second**	Semester: **IV**
Subject: BCA		
Course Code: **BCA-120P**	Course Title: **Java and Windows Lab**	

Course Objectives:

Prepare students to excel in object-oriented programming and to succeed as a Java developer.

- Provide students with a solid foundation in OOP fundamentals required to solve programming problems.
- Inculcate multidisciplinary approach and an ability to relate java programming issues to broader application context.

Course Outcomes:

CO1 Understand the necessity for Object Oriented Programming paradigm over structured programming.
CO2 Develop java programs, analyze, and interpret object-oriented data and report results.
CO3 Demonstrate an ability to design an object-oriented system, AWT components.
CO3 Apply event driven model programming skills using GUI interfaces to develop various applications.

Credits: **04**	Type of Course: **Practical Lab**
Max. Marks: **100**	Min. Passing Marks: 36

List of Practical in Java Programming:

1. Write a java program for Method overloading and Constructor overloading.
2. Write a java program to display the employee details using Scanner class.
3. a) Write a java program to represent Abstract class with example.
   b) Write a java program to implement Interface using extends keyword.
4. Write a java program to implement method overloading, method overriding, dynamic method dispatch.
5. Write a java program to implement single, multilevel, hierarchal, multiple, hybrid inheritances.
6. Write java programs that demonstrate the use of abstract, this, super, static, final keywords.
7. a) Write a java program for creating a package and using a package.
   b) Write a java program to demonstrate the use of wrapper classes.
8. a) Write a java program using all five keywords of exception handling mechanism.
   b) Write a java program for creating customized (user) exception
9. a) Write a java program to create the following AWT components: Button, Checkbox, Choice, and List.
   b) Write java programs to create AWT application using containers and layouts.
10. a) Write a java program to create a file, write the data and display the data.
    b) Write a java program that reads a file name from user and displays its information.

List of Practical in Windows Programming:

1. Design an application using form to demonstrate the use of textbox, label, command button and scroll bars.
2. Demonstrate the use of file list box, directory list box and drive list box with a suitable application.
3. Implement a font dialog box using combo/list, text, option buttons, and check box control.
4. Design a simple application to demonstrate OLE control.
5. Develop a simple application like Notepad (eg. File , Edit , format, open, font, save, copy and cut) using menu editor, MDI, common dialog box.
6. Demonstrate database concepts (Data control,DAO,RDO,ADO, DB-list, DB combo) using a simple database application.

Suggested Readings:

https://mrcet.com/pdf/Lab%20Manuals/Lab%20Manual%20Object%20Oriented%20Programming%20through%20JAVA.pdf

Programme: **BCA**	Year: **Third**	Semester: **V**
Subject: BCA		
Course Code: **BCA-121N**	Course Title: **Information and Network Security**	
Course Objectives: This course aims to provide a basic understanding of the existing algorithms used to protect users online and understand some of the design choices behind these algorithms. The course offers a workable knowledge of the mathematics used in cryptology. The course emphasizes giving a basic understanding of previous attacks on cryptosystems to prevent future attacks.		
Course Outcomes: **CO1** Identify information security goals, classical encryption techniques and acquire fundamental knowledge on the concepts of finite fields and number theory. **CO2** Understand, compare and apply different encryption and decryption techniques to solve problems related to confidentiality and authentication **CO3** Apply the knowledge of cryptographic checksums and evaluate the performance of different message digest algorithms for verifying the integrity of varying message sizes **CO4** Apply different digital signature algorithms to achieve authentication and create secure applications **CO5** Apply network security basics, analyze different attacks on networks and evaluate the performance of firewalls and security protocols like SSL, IPSec, and PGP. **CO6** Apply the knowledge of cryptographic utilities and authentication mechanisms to design secure applications.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Information security and Symmetric Ciphers**
Unit 1	**Introduction:** History, what is Information Security; Characteristics of Information; Information Security Model; Components of an Information Security; Aspects of Information security: Security attacks, Security Mechanism, and Security Services (X.800), Model for Network Security.
Unit 2	**Classical Encryption Techniques: Historical** background, symmetric cipher model, Substitution techniques, Transposition techniques, steganography.
Unit 3	**Block ciphers and DES:** Block cipher principles, Data encryption standard, strength of DES, differential and cryptanalysis, block cipher design principles, block cipher mode of operation.
Unit 4	**Confidentiality Using Symmetric Ciphers:** Placement of encryption function, traffic confidentiality, key distribution, random number generation.
Block 2	**Public key Encryption and Hash Functions**
Unit 5	**Introduction to Number Theory:** Prime numbers, Fermat's and Euler's theorem, discrete logarithm
Unit 6	**Public Key Cryptography:** Public-Key Cryptography Principles, RSA, Key Management: Diffi-Hellman key exchange.
Unit 7	**Message Authentication and Hash Functions:** Authentication requirements, Authentication Functions, Message Authentication codes, Hash Functions, SHA-1, MD5.
Unit 8	**Digital Signatures:** Digital signatures, Authentication protocols, Digital Signature standard
Block 3	**Network Security Applications**
Unit 9	**Authentication Applications:** Kerberos Motivation, X.509 authentication service
Unit 10	**Electronic Mail Security:** PGP: PGP Notation, PGP Operational Description, S/MIME
Unit 11	**IP Security: IP** Security Overview, IP Security Architecture, Authentication Header

Unit 12	**Web Security:** Web Security Threats, Web Traffic Security Approaches, Overview of Secure Socket Layer and Transport Layer Security, Overview of Secure Electronic Transaction
Block 4	**Intruders and Viruses**
Unit 13	**Intruders:** Intruders, Intrusion Techniques, Password Protection, Password Selection Strategies, Intrusion Detection,
Unit 14	**Malicious Programs:** Malicious Programs, Nature of Viruses, Types of Viruses, Macro Viruses, Antivirus Approaches
Unit 15	**Firewall**: Firewall Characteristics, Types of Firewalls, Firewall Configuration

Suggested Readings:

1. Douglas Stinson, "Cryptography Theory and Practice", 2nd Edition, Chapman & Hall/CRC.
2. B. A. Forouzan, "Cryptography & Network Security", Tata Mc Graw Hill.
3. W. Stallings, "Cryptography and Network Security", Pearson Education.

Suggested online courses (MOOCs)

1. NOC:Cryptography And Network Security, IIT Kharagpur by Prof. Sourav Mukhopadhyay https://nptel.ac.in/courses/106105162
2. Cryptography and Network Security, IIT Kharagpur by Dr. Debdeep Mukhopadhyay https://nptel.ac.in/courses/106105031

This course can be opted as an elective by the students of following subjects: **MCA**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A.

Programme: **BCA**	Year: **Third**	Semester: **V**
Subject: Computer Science		
Course Code: **BCA-122N**	Course Title: **Design And Analysis Of Algorithms**	

Course Objectives: This course provide the common paradigms to design efficient algorithms for real world problem solving. It gives an understanding of how to analyze the asymptotic performance of algorithm; write rigorous correctness proofs for algorithms; important algorithmic design paradigms and methods of analysis; efficient algorithms in common engineering design situations.

Course Outcomes:

CO1 Understand that various problem solving methods exist such as; iterative technique, divide and conquer, dynamic programming, greedy algorithms.

CO2 Analyze the strengths and weaknesses of an algorithm theoretically as well as practically.

CO3 Identify and apply an appropriate technique to design an efficient algorithm for simple problems.

CO4 Demonstrate correctness and efficiency of the algorithm.

CO5 Apply various searching and sorting algorithms.

Credits: **04**	Type of Course: **Core**
Max. Marks: **100**	Min. Passing Marks: **36**

Block 1	**Introduction and Design Strategies-I**
Unit 1	**Introduction:** Algorithm, Psuedo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Growth of functions: Asymptotic Notation, Recurrences: substitution method, master method.
Unit 2	**Divide and Conquer:** General method, applications-Binary search, Finding the maximum and minimum, Quick sort, Heapsort, Strassen's Matrix Multiplication.
Unit 3	**Sorting in Linear Time:** Lower bounds for sorting, Counting sort, Radix sort, Bucket sort, Medians and Order Statistics, Minimum and maximum.
Block 2	**Algorithm Design Strategies-II**
Unit 4	**Greedy method:** General method, applications- Knapsack problem, Job sequencing with deadlines, optimal two way merge patterns, Huffman codes, Minimum cost spanning trees: Prims and Kruskal's algorithm, Single source shortest paths: The Bellman-Ford algorithm, Dijkstra's algorithm.
Unit 5	**Dynamic Programming:** General method, applications, capital budgeting problem, Multistage graphs, Matrix chain multiplication, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem.
Block 3	**Algorithm design strategies & Completeness**
Unit 6	**Graph Algorithms:** Introduction, representation of graphs, Breadth first search, depth first search, topological sort, strongly connected component, flow networks, ford-fulkerson method.
Unit 7	**Backtracking:** General method, applications, 8-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.
Unit 8	**Branch-And-Bound:** The method, travelling salesperson problem, 15 puzzle problem.
Unit 9	**NP-Hard and NP-Complete problems:** Basic concepts, non deterministic algorithms, NP - Hard and NP Complete classes, satisfiability problem, reducibility.

Suggested Readings:

1. Cormen, Leiserson, Rivest,and Stein, "Introduction to Algorithms", MIT Press ,Third Edition, 2009.
2. Dasgupta, Papadimitrou and Vazirani, "Algorithms", McGraw-Hill Education, 2006. Horowitz, Sahni, and Rajasekaran, "Computer Algorithms" Silicon Press, 2007

Suggested online courses (MOOCs)

1. NOC:Design and Analysis of Algorithms, Chennai Mathematical Institute By Prof. Madhavan Mukund

https://nptel.ac.in/courses/106106131 2. NOC:Introduction to algorithms and analysis, IIT Kharagpur by Prof. Sourav Mukhopadhyay https://nptel.ac.in/courses/106105164 3. Design and Analysis of Algorithms, IIT Bombay By Prof. Abhiram Ranade https://archive.nptel.ac.in/courses/106/101/106101060/#
This course can be opted as an elective by the students of following subjects: **MCA**
Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Third**	Semester: **V**
Subject: Computer Science		
Course Code: **BCA-123N**	Course Title: **Computer Graphics**	
Course Objectives: The primary role of computer graphics is to render the digital content (0's and 1's) in a human-comprehensible form on the computer screen. This course introduces various object representation techniques along with 2D and 3D transformation, clipping, splines, objects modeling, colour modeling, lighting, textures and visible surface detection.		
Course Outcomes: **CO1** Demonstrate an understanding of contemporary graphics hardware. **CO2** Draw graphics using line & polygon and ability to perform operations on computer graphics. **CO3** Understand and demonstrate geometrical transformations, Segment, Windowing and Clipping, Interaction. **CO4** Demonstrate Hidden Surfaces & Lines; Light, Colour & Shading; Curves and Fractals		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	
Block 1	**Raster Graphics and Clipping**	
Unit 1	Unit 1: Introduction to Computer Graphics: What is Computer Graphics?, Application of Computer Graphics, Presentation Graphics, Painting and Drawing, Photo Editing, Scientific Visualization, Image Processing, Digital Art, Education, training, Entertainment and CAD Simulation, Animation and Games, Graphics Hardware, Input and Output Devices, Touch Panel, Light Pens, Graphic Tablets, Plotters, Film Recorders, Display Devices, Refreshing Display Devices: Raster-Scan, Random-Scan, Plasma Panel and LCD panels	
Unit 2	Unit 2: Graphics Primitives: Points and Lines, Line-drawing Algorithms: DDA Algorithm, Bresenham's line Algorithm, Circle-generating Algorithm: Properties of Circles, Midpoint Circle of Algorithm, Polygon Filling Algorithm: Scan-Line	
Unit 3	Unit 3: 2-D Viewing and Clipping: Point Clipping, Line Clipping: Cohen-Sutherland Line Clippings, Cyrus-Beck Line Clipping Algorithm, Polygon Clipping: Sutherland Hodgman Algorithm, Windowing Transformation	
Block 2	**Transformations**	
Unit 4	Unit 4: 2-D and 3-D Transformations: Basic Transformations: Translation, Rotation, Scaling, Shear, Composite Transformations: Rotations about a point, Reflection about a line, Homogeneous Coordinate Systems, 3-D Transformations	
Unit 5	Unit 5: Viewing Transformation: Projections: Parallel Projection, Orthographic & Oblique Projections, Isometric Projections, Perspective Projections	
Block 3	**Modeling & Rendering**	
Unit 6	Unit 6: Curves and Surfaces: Polygon Representation Methods: Polygon Surfaces, Polygon Tables, Plane Equations, Polygon Meshes, Bezier Curves and Surfaces: Bezier Curves, Properties of Bezier Curves, Bezier Surfaces, Surface of Revolution	
Unit 7	Unit 7: Visible – Surface Detection: Depth Buffer Method, Scan-Line Method, Area-Subdivision Method	
Unit 8	Unit 8: Polygon Rendering and Ray Tracing Methods: Illumination Model: Ambient Reflection, Diffuse Reflection, Specular Reflection, Shading: Gouraud Shading, Phong Shading, Ray Tracing: Basic Ray-Tracing Algorithm	
Suggested Readings: 1. J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes, Computer Graphics - Principles and Practice, Second Edition in C, Pearson Education, 2003. 2. D. Hearn and M. Pauline Baker, Computer Graphics (C Version), Pearson Education,		

2nd Edition, 2004.
3. Edward Angel, Interactive Computer Graphics A Top-Down Approach with OpenGL 5th Edition, Addison-Wesley, 2008.
4. Prabat K Andleigh and KiranThakrar, "Multimedia Systems and Design", PHI, 2003.

Suggested online courses (MOOCs)

1. Computer Graphics, IIT Madras by Prof. Sukhendu Das
   https://nptel.ac.in/courses/106106090
2. Introduction to Computer Graphics, IIT Delhi by Prof. Prem K Kalra
   https://nptel.ac.in/courses/106102065
3. NOC:Computer Graphics, IIT Guwahati by Prof. Samit Bhattacharya
   https://nptel.ac.in/courses/106103224

This course can be opted as an elective by the students of following subjects: This course can be opted as an elective by the students of following subjects: **B.Sc. (Computer Science) and BCA**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A.

Programme: **BCA**	Year: **Third**	Semester: **V**
Subject: BCA		
Course Code: **BCA-101EA**	Course Title: **Web Technology**	
Course Objectives: This course intended to also expose students to the basic tools and technologies used in the development of a web application for the World Wide Web. This includes – Internet technologies, HTML, XML, JavaScript, and JSP.		
Course Outcomes: **CO1** Implement interactive web page(s) using HTML, CSS and JavaScript. **CO2** Design a responsive web site using HTML5 and CSS **CO3** Build Dynamic web site using server side PHP Programming and Database connectivity. **CO4** Describe and differentiate different Web Extensions and Web Services. **CO5** Understand development of web application.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	
Block 1	History of the Internet and World Wide Web -III ML 4 protocols - RCM, SMTP, POP), MIME, IMAP. Introduction to JAVA Scripts - Object Based Scripting for the web, Structures - Functions - Arrays - Objects.	
Block 2	Introduction - Object refers, Collectors all and Children. Dynamic style, Dynamic position, frames. navigator, Event Model - On check - On load - - Form process - Event Bubblers- filters -Transport with the Filter - Creating Images Adding shadows - Creating Gradients - Creating Motion with Bar-Data Binding - Simple Data Binding - Moving with a record set - Sorting table data, binding of an image and table.	
Block 3	database, Relational Database model - Overview, SQL - ASP - Working of ASP - Objects - File System Objects - Session tracking and cookies - ADO - Access a Database from ASP - Server side Active-X Components - Web Resources - XMIL - Structure in Data Name spaces - D7D- Vocabularies - DOM methods.	
Block 4	Introduction, Servlet, Overview Architecture - Dandling II P Request - Go and post request - redirecting request multi-tier applications - ISP - Overviews - Objects - scripting - Standard Actions - Directives. Brief survey of Web 2.0 technologies, introduction to Semantic web and other current technologies.	
Suggested Readings: 1. Deitel, Deitel, Goldberg, "Internet & World Wide Web How to Program", Third Edition, Pearson Education, 2006. 2. Raj Kamal, "Internet and Web Technologies", Tata McGraw-Hill.		
This course can be opted as an elective by the students of following subjects: **B.Sc.(Computer Science), MCA**		
Suggested equivalent online courses (MOOCs) for credit transfer: N.A		

Programme: BCA	Year: **Third**	Semester: **V**
Subject: BCA		
Course Code: **BCA-124P**	Course Title: **Algorithm and Computer Graphics Lab**	

Course Objectives:
The aim of this course is to enhance programming skills while improving their practical knowledge of efficient algorithmic design practices. It strengthens the practical ability to apply suitable data structures for real-time applications.

Course Outcomes:
CO1 Implement linear data structures such as stacks, queues using array and linked list.
CO2 Understand and implements non-linear data structures such as trees, graphs.
CO3 Implement various kinds of searching, sorting and traversal techniques and know when to choose which technique.
CO4 Draw graphics using line & polygon and demonstrate various geometrical transformations.

Credits: **04**	Type of Course: **Practical Lab**
Max. Marks: **100**	Min. Passing Marks: 36

List of Practical in Algorithm Lab with C language:

1. Implementation of Stacks, Queues (using both arrays and linked lists).
2. Implement a program to evaluate a given postfix expression using stacks.
3. Implement the following operations on singly and circular linked list: i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Implementation of operations on binary tree (delete entire tree, copy entire tree, mirror image, level order, search for a node etc.)
5. Implementation of the following operations on binary search tree (BST): (a) Minimum key (b) Maximum key (c) Search for a given key (d) Delete a node with given key
6. Implementation of graph traversals by applying: (a) BFS (b) DFS
7. Implement the following algorithms to find out a minimum spanning tree of a simple connected undirected graph: (a) Prim‟s algorithm (b) Kruskal‟s algorithm
8. Implement Dijkstra‟s algorithm for solving single source shortest path problem.
9. Implementation of recursive and non-recursive functions to perform the following searching operations for a key value in a given list of integers: i) Linear search ii) Binary search
10. Implement the following sorting algorithms: a) Bubble sort b) Selection sort c) Insertion sort (d) Merge sort (e) Quick sort (f) Heap sort

List of Practical in Computer Graphics Lab with C language:

1. Implementation of line generation DDA and Bresenhem's algorithms.
2. Implementation of circle generation using Mid-point method and Bresenhem's algorithm.
3. Implementation of polygon filling using Scan-line algorithms.
4. Implementation of 2D transformation: Translation, Scaling, Rotation, Mirror Reflection and Shearing.
5. Implementation of Line Clipping using Cohen-Sutherland algorithm and Bisection Method.
6. Implementation of Polygon Clipping using Sutherland- Hodgeman algorithm.
7. Implementation of 3D geometric transformations: Translation, Scaling and rotation.
8. Implementation of Curve generation using Interpolation methods, B-spline and Bezier curves.
9. Implementation of any one of Back face removal algorithms such as Depth-Buffer algorithm, Painter's algorithm, Warnock's algorithm, Scan-line algorithm.

Suggested Readings: **N.A.**

Programme: **BCA**		Year: **Third**	Semester: **V**
Subject: BCA			
Course Code: **BCA-101N**		Course Title: **Client Server Technology**	
Course Objectives: This course gives fundamental principles of Client-Server technology. The course revolves around the knowledge of ASP.NET framework and its related components like form and controls, state management, and configuration. It discusses ASP.NET web services, HTML and JavaScript, DHTML, AJAX, and a small application.			
Course Outcomes: **CO1:** Design web applications using ASP.NET. **CO2:** Use ASP.NET controls in the web applications. **CO3:** Create database driven ASP.NET web applications. **CO4:** Incorporate session state, application state and cookies in the web application.			
Credits: **04**		Type of Course: **Core**	
Max. Marks: **100**		Min. Passing Marks: **36**	
Block 1	**Introduction to Client-Server Computing**		
Unit 1	Introduction to Client-Server Computing: Introduction to Client-Server Architecture, Client-Server computing and its uses, historical development, downsizing and client server computing, mainframe computing, client-server technology and heterogeneous computing, advantages of client server computing.		
Unit 2	Distributed Computing: Distributed Computing, File Server versus Client/Server Database, Computing platforms, Microprocessor integration and client server computing, implementations and scalability.		
Unit 3	Designing Client-Server Applications: Fundamentals of client server design, division of labor, Transition to client-server programming; Interaction of client and server communication Techniques and protocols, implementing client server applications.		
Block 2	**Introduction to ASP.NET**		
Unit 4	Introduction to .NET Framework: Introduction, The Origin of .Net Technology, Common Language Runtime (CLR), Common Type System (CTS), Common Language Specification (CLS), Microsoft Intermediate Language (MSIL), Just-In –Time Compilation, Framework Base Classes.		
Unit 5	Traditional ASP Basics: Introduction to ASP, How ASP Works, ASP Objects, Installing IIS on Windows 7 & Windows 8, Sample Programs, Importance's of Form tag and how it works.		
Unit 6	ASP.NET Introduction & Controls: ASP.NET Introduction, First ASP.NET Application, Auto Postback Property, Event Handler, Parameters, Dynamically intializing Controls, IsPostBack property of Page class, ListControls, Comparison between HtmlControls and WebControls, Control Properties and Methods, FileUpload Control		
Block 3	**Working with Forms and Controls**		
Unit 7	Working with Forms and Controls: Life Cycle of ASP.NET Page, Creating an ASP.NET Web Application Project, Creating Web Forms, Using Server Controls, Using Code-Behind Pages, Web Server Controls, Using Validation controls usage of skins and themes.		
Unit 8	ADO.Net: Introduction to ADO.NET, .NET Framework data providers, Data Binding, Connecting to the Database, Accessing Data with DataSets, Displaying a DataSet in a List-Bound Control, Using Multiple Tables, Accessing Data with DataReaders, Disconnected operations with Data tables and Data sets, Connection pooling, Working with LINQ.		

Unit 9	ASP.NET State Management: Application and Session Variables, Cookies, Storing Session Variables in a Database, Cleaning the session state, Types of Assemblies, Private vs. Shared assemblies, Creating and placing strongly named assemblies.
Unit 10	Configuration: Windows configuration, .net configuration, cashing, Types of Caching, SQL Cache Invalidation
Block 4	**Client Side and Server Side Login Services**
Unit 11	HTML & JavaScript: Understanding HTML Form Tag and elements within it, Javascript using Sample Programs, Working with CSS, Use Themes to Customize a Site,Web based security, ASP.NET authentication service, managing user,asp.net login controls, authorizing users.
Unit 12	ASP.Net Web Services: Introduction to web services, creating web services, invoking web services,
Unit 13	AJAX: Introduction to AJAX, AJAX.NET, Script Manager, Update Panel, Update Progress, Timer, AJAX Control Toolkit, server side support for AJAX, AJAX client support.
Unit 14	Developing a small application using ASP.NET for any case study.
Suggested equivalent online courses (MOOCs) for credit transfer: N.A	

Programme: **BCA**	Year: **Third**	Semester: **VI**
Subject: BCA		
Course Code: **BCA-127N**	Course Title: **Python Programming**	
Course Objectives: Provide students with hands-on experience on python programming for solving data science problems.		
Course Outcomes: **CO 1:** The students develops a sound approach to problem solving using a high level programming language. **CO 2:** Use Python data structures – lists, tuples & dictionaries for representing compound data. **CO 3:** The students master the good programming practices like modularity and documentation, and use of named constants. **CO 4:** The student learns the use of object oriented framework using the concept of classes, inheritance, and encapsulation.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Basics of Python**
Unit I	Introduction: History of Python, Need of Python Programming, Applications Basics of Python Programming Using the REPL (Shell), Running Python Scripts, Python IDLE.
Unit II	Tokens and Statements: Variables, Constants, Assignment, Multiple Assignment, Keywords, Punctuators, Identifiers, Input-Output, Indentation, Statements, Comments, Single Comment and Multiline Comment.
Unit III	Data Types, Operators & Expressions: Types – Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Operators precedence, Expressions and order of evaluations Control Flow- if, if-else, if-elif-else, for, while, break, continue, pass.
Block 2	**Data Structure in Python**
Unit IV	Data Structures: Stack & Queue, Lists – Operations, Slicing, Methods; Tuples – Operations, Methods , Sets– Operations , Methods, Dictionaries– Operations , Methods, Sequences– Operations, Methods. Comprehensions– Operations, Methods.
Unit V	Functions – Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions (Function Returning Values), Scope of the Variables in a Function- Global and Local Variables.
Unit VI	Modules & Packages: Modules, Creating modules, import statement, from. Import statement, name spacing, Python packages, Introduction to PIP, Installing Packages via PIP, Using Python Packages.
Block 3	**Oops in Python**
Unit VII	Object-Oriented Programming OOP in Python: Classes, 'self-variable', Methods, Constructor Method, Inheritance, Overriding Methods, Data hiding.
Unit VIII	Exception Handling :Error, and Exceptions: Difference between an error and Exception, Handling Exception, try except for block, Raising Exceptions, User Defined Exceptions
Unit IX	Python Libraries: Brief Tour of the Standard Library – Operating System Interface – String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression.
Unit X	GUI Programming and Testing : Multithreading, GUI Programming, Turtle Graphics Testing: Why testing is required ?, Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.
Block 4	**Machine Learning in Python**

Unit XI	Machine Learning Using Python: Machine Learning Basics, Features and Labels, Supervised and Unsupervised Learning.
Unit XII	Regression and Classification in Machine Learning: Simple Linear Regression, Multiple Regression, Data Collection for Machine Learning, Classification – Features and Types.

Suggested Readings:

1. Python Programming: A Modern Approach, VamsiKurama, Pearson
2. Learning Python, Mark Lutz, Orielly
3. Think Python, Allen Downey, Green Tea Press
4. Core Python Programming, W.Chun, Pearson.
5. Introduction to Python, Kenneth A. Lambert, Cengage

Suggested online courses (MOOCs)

1. NOC:Programming, Data Structures and Algorithms using Python, Chennai Mathematical Institute by Prof. Madhavan Mukund
https://nptel.ac.in/courses/106106145
2. NOC:The Joy of Computing using Python, IIT Ropar by Prof. Sudarshan Iyengar
https://nptel.ac.in/courses/106106182
3. Python for Data Science By Prof. Ragunathan Rengasamy, IIT Madras
https://onlinecourses.nptel.ac.in/noc22_cs32/preview

This course can be opted as an elective by the students of following subjects: **B.Sc.(Computer Science), M.Sc. (Statistics) and M.Sc. (Mathematics)**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A.

Programme: **BCA**	Year: **Third**	Semester: **VI**
Subject: BCA		
Course Code: **BCA-128N**	Course Title: **Soft Computing**	
Course Objectives: Expose students to Neural Network, Fuzzy Logic and Genetic Algorithms, which are the major building blocks of Intelligent Systems.		
Course Outcomes: **CO1** Discuss the ideas of fuzzy sets, fuzzy logic and use of heuristics based on human experience. **CO2** Understand how neural networks learn from available examples and generalize to form appropriate rules for inference systems. **CO3** Provide the mathematical background for carrying out the optimization associated with neural network learning. **CO4** Apply genetic algorithms and other random search procedures for finding global optimum of optimization problems.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block 1	**Artificial Intelligence & Soft Computing:** Introduction of Artificial Intelligence, Problem domain of AI, AI techniques, Rule based system, monotonic reasoning, non-monotonic reasoning, Uncertainty reasoning & Inference, Bayesian theory and dependency network, Limitation of AI, Soft computing paradigms, pattern classification, association and mapping, Pattern recognition techniques.
Block 2	**Fuzzy Set Theory**: Introduction to Neuro – Fuzzy and Soft Computing – Fuzzy Sets – Basic Definition and Terminology – Set-theoretic Operations – Member Function Formulation and Parameterization – Fuzzy Rules and Fuzzy Reasoning – Extension Principle and Fuzzy Relations – Fuzzy If-Then Rules – Fuzzy Reasoning – Fuzzy Inference Systems.
Block 3	**Neural Network:** Structure and Function of a single neuron: Biological neuron, artificial neuron, definition of ANN, Taxonomy of neural net, Difference between ANN and human brain, characteristics and applications of ANN, single layer network, Perceptron training algorithm, Linear separability, Widrow & Hebb's learning rule/Delta rule, ADALINE, MADALINE, AI v/s ANN. Introduction of MLP, different activation functions, Error back propagation algorithm, derivation of BBPA, momentum, limitation, characteristics and application of EBPA, Deep Learning: Convolution Neural Network, Recurrent Neural Network.
Block 4	**Genetic Algorithm:** Fundamentals, basic concepts, working principle, encoding, fitness function, reproduction, Genetic modeling: Inheritance operator, cross over, inversion & deletion, mutation operator, Bitwise operator, Generational Cycle, Convergence of GA, Applications &advances in GA, Differences & similarities between GA & other traditional method.

Suggested Readings:

Text Books

1. J.S.R.Jang, C.T.Sun and E.Mizutani, "Neuro-Fuzzy and Soft Computing", PHI, 2004, Pearson Education 2004.
2. S. Rajasekaran and G.A.VijaylakshmiPai.. Neural Networks Fuzzy Logic, and Genetic Algorithms, Prentice Hall of India.
3. Timothy J.Ross, "Fuzzy Logic with Engineering Applications", McGraw-Hill, 1997.
4. Davis E.Goldberg, "Genetic Algorithms: Search, Optimization and Machine Learning", Addison Wesley,N.Y.,1989.
5. S. Rajasekaran and G.A.V.Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", PHI, 2003.
6. R.Eberhart, P.Simpson and R.Dobbins, "Computational Intelligence - PC Tools", AP Professional, Boston, 1996.

Suggested online courses (MOOCs) 1. NOC:Introduction to Soft Computing, IIT Kharagpur by Prof. Debasis Samanta https://nptel.ac.in/courses/106105173
This course can be opted as an elective by the students of following subjects: **M.Sc. (Statistics) and M.Sc. (Mathematics)**
Suggested equivalent online courses (MOOCs) for credit transfer: N.A.

Programme: **BCA**	Year: **Third**	Semester: **VI**
Subject: BCA		
Course Code: **BCA-101N**	Course Title: **Computer Architecture**	
Course Objectives: The course aims to impart understanding of fundamental and advanced concepts of parallel computing and design architecture. It illustrates concepts of memory and input-output subsystems, pipelining and vector processing, microprocessor algorithms and systems and control mechanisms.		
Course Outcomes: **CO1** Familiarizes the students with basics of computer hardware and how software interacts with computer hardware. **CO2** Introduces how computers represent and manipulate data, computer arithmetic and conversion between different number systems. **CO3** Introduces how Boolean algebra is related to designing computer logic, through simple combinational and sequential logic circuits. **CO4** Introduces basics of Instruction Set Architecture (ISA). **CO5** Familiarize students with a simple computer with hardware design including data format, instruction format, instruction set, addressing modes, bus structure, input/output, memory, Arithmetic/Logic unit, control unit, and data, instruction and address flow. **CO6** Design combinational and sequential logic circuits, flip-flops, counters, shift registers, adders, substractor, multiplexer, demultiplexer, Arithmetic/Logic unit. **CO7 I**ntroduces concept of memory unit and input/output architecture.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	
Block 1	**Processor Basics**	
Unit 1	**CPU organization:** Fundamentals, additional features	
Unit 2	**Data representation:** Basic formats, fixed point numbers, floating point numbers	
Unit 3	**Instruction sets:** Instruction formats, instruction types, programming considerations	
Block 2	**Data path Design**	
Unit 4	Fixed point arithmetic Addition and subtraction, multiplication and division	
Unit 5	**Arithmetic Logic Unit:** Combinational ALUs, sequential ALUs	
Unit 6	**Advanced Topics:** Floating point arithmetic, pipeline processing	
Block 3	**Control Design**	
Unit 7	**Basic concepts:** Introduction, hardwired control, design examples	
Unit 8	**Micro programmed control:** Basic concepts, multiplier control unit, CPU control unit	
Unit 9	**Pipeline control:** Instruction pipelines, pipeline performance, super scalar processing	
Block 4	**Memory Organization**	
Unit 10	**I/O and System:** Control Programmed IO, DMA and Interrupts, 10 processors	
Unit 11	Parallel processing: Processor-level parallelism, multiprocessor	

Suggested Readings:

1. Computer Organization & Architecture - Designing for Performance by William Stallings, Eighth Edition, Pearson, 2010
2. Computer Architecture: A Quantitative Approach by John L. Hennessy and David A. Patterson, Fourth Edition, Morgan Kaufmann Publishers
3. Computer System Architecture by M. Morris Mano, Third Edition, Pearson Education Inc

Suggested online courses (MOOCs)

1. Computer Architecture, IIT Delhi by Prof. Anshul Kumar
   https://nptel.ac.in/courses/106102062

2. NOC:Computer architecture and organization, IIT Kharagpur by Prof. Indranil Sengupta, Prof. Kamalika Datta https://nptel.ac.in/courses/106105163 3. NOC:Computer Architecture, IIT Delhi by Prof. Smruti R.Sarangi https://nptel.ac.in/courses/106102157 4. NOC:Computer Architecture(Course sponsored by Aricent), IIT Madras by Prof.Madhu Mutyam https://nptel.ac.in/courses/106106134
This course can be opted as an elective by the students of following subjects: **M.Sc.(Computer Science), MCA**
Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Third**	Semester: **VI**
Subject: BCA		
Course Code: **BCA-ED**	Course Title: **Microprocessor and its applications**	
Course Objectives: This course provides architecture and organization of microprocessor along with instruction set format. It discusses modes and functional block diagram of 8085 AND 8086 along with pins and their functions; describe memory and addressing modes; explains use different types of instructions, directives and interrupts; illustrates assembly language programs using various programming tools.		
Course Outcomes: **CO1** Apply basic binary math operations using the microprocessor. **CO2** Demonstrate programming using various addressing modes and data transfer instructions of the target microprocessor and microcontroller. **CO3** Compare different Microprocessors (8085 & 8086) and Microcontroller to meet specified performance requirements. **CO4** Analyze and use assembly language programs to solve real-world control problems.		
Credits: **04**	Type of Course: **Core**	
Max. Marks: **100**	Min. Passing Marks: **36**	

Block-1	**Introduction to Microprocessor**
Unit 1	**Introduction of Microcomputer System:** CPU, I/O devices, clock, memory, bussed architecture, tristate logic, address bus, data bus and control bus.
Unit 2	**Semiconductor Memories :** Development of semiconductor memory, internal structure and decoding, memory read and write timing diagrams, MROM, ROM, EPROM, EEPROM, DRAM,
Unit 3	**Architecture of 8-bit Microprocessor:** Intel 8085A microprocessor, Pin description and internal architecture.
Unit 4	**Operation and Control of Microprocessor:** Timing and control unit, op-code fetch machine cycle, Memory read/write machine cycles, I/O read/write machine cycles, interrupt acknowledge machine cycle, state- transition diagram.
Block-2	**Operations, Instruction Set and Assembly Language Programming**
Unit 5	**Instruction Set:** Addressing modes; Data transfer, arithmetic, logical, branch, stack and machine control groups of instruction set, macro RTL and micro RTL flow chart of few typical instructions; Unspecified flags and instructions.
Unit 6	**Assembly Language Programming:** Assembler directives, simple examples; Subroutines, parameter passing to subroutines.
Block-3	**Interface, Interrupt and Programmable Interface**
Unit 7	**Interfacing:** Interfacing of memory chips, address allocation technique and decoding; Interfacing of I/O devices, LEDs and toggle-switches as examples, memory mapped and isolated I/O structure; Input/Output techniques: CPU initiated unconditional and conditional I/O transfer, device initiated interrupt I/O transfer.
Unit 8	**Interrupts:** Interrupt structure of 8085A microprocessor, processing of vectored and non- vectored interrupts, latency time and response time; Handling multiple interrupts

Unit 9	**Programmable Peripheral Interface:** Intel 8255, pin configuration, internal structure of a port bit, modes of operation, bit SET/RESET feature, programming; ADC and DAC chips and their interfacing.
Block-4	**Timer, Controllers and Applications**
Unit 10	**Programmable Interval Timer:** Intel 8253, pin configuration, internal block diagram of counter and modes of operation, counter read methods, programming, READ-BACK command of Intel 8254.
Unit 11	**Programmable Interrupt Controller 8253/8254:** Pin configuration, Timer or counter, Internal structure, Interfacing with system, Mode (0,1,2,3,4,5), Reading timer, Read back command feature.
Unit 12	**Programmable Interrupt Controller 8259A:** Priority interrupt structure, Intel 8259, Pin configuration, Functional Block Diagram, Interrupt sequence, Initiallization control words ICW1, ICW2, ICW3, ICW4, Operation Control Words(OCWs), Fully nested mode, EOI mode, Poll command, Reading status registers, Special fully nested mode, Cascade mode.
Unit 13	**Application of Microprocessor:** Various applications of Intel 8085 Microprocessor, Microprocessor based stepper motor control system using 8085, traffic light controller using 8085, ADC interface using 8085, DAC interface using 8085

Suggested Readings:

1. 'Fundamentals of Microprocessors and microcontrollers' by B. Ram, Eighth Revised Edition, Dhanpat Rai Publications.
2. Fundamentals of Microprocessors and Microcontrollers, B. Ram, Dhanpat Rai Publications.
3. Microprocessors and Microcontrollers, S K Mandal. WBUT Series by TMH

Suggested online courses (MOOCs)

1. NOC:Microprocessors And Microcontrollers, IIT Kharagpur by Prof. Santanu Chattopadhyay https://nptel.ac.in/courses/108105102

This course can be opted as an elective by the students of following subjects: **M.Sc.(Computer Science), MCA**

Suggested equivalent online courses (MOOCs) for credit transfer: N.A

Programme: **BCA**	Year: **Third**	Semester: **VI**
Subject: **BCA**		
Course Code: **BCA-130RP**	Course Title: **Project with viva voce**	
Course Objectives: ➢ To facilitate the learner to independently formulate and solve a social, philosophical, commercial, or technological problem and present the results in written and oral form. ➢ To render learners to real-life problems. ➢ To provide opportunities for learners to interact with people and present them confidently.		
Course Outcomes: **CO1** Investigate and evaluate a research topic relevant to environment and society. **CO2** Learn systematic discovery and critical review of appropriate and relevant information sources. **CO3** Apply qualitative and/or quantitative evaluation processes to original data. **CO4** Communicate research concepts and contexts clearly and effectively both in writing and orally		
Credits: **08**	Type of Course: **Application**	
Max. Marks: **100**	Min. Passing Marks:	