New - BCA

ದಾವಣಗೆರೆ ವಿಶ್ವವಿದ್ಯಾನಿಲಯ, ದಾವಣಗೆರೆ

DAVANGERE UNIVERSITY, DAVANGERE

PROGRAM /COURSE STRUCTURE AND SYLLABUS
as per the Choice Based Credit System (CBCS)
designed in accordance with
Learning Outcomes-Based Curriculum Framework (LOCF)
of National Education Policy (NEP) 2020
for
BACHELOR OF COMPUTER APPLICATIONS (BCA)

w.e.f.

Academic Year 2021-22 and onwards

BOS Chairman
Dept. of Computer Science
Davangere University
Shivagangotri, Davangere

ಗಣಕ ವಿಜ್ಞಾನ ವಿಭಾಗ

DEPARTMENT OF STUDIES IN COMPUTER SCIENCE

PREAMBLE

BCA is an excellent academic course in the field of computer applications. For those who want to pursue a successful and rewarding career in the fields of computers and information technology, BCA comes out as a compelling course option. Aspirants study various aspects of computer science, apart from developing sound knowledge and understanding of the latest advancements specific to this field. By delivering theoretical and practical knowledge, BCA course prepares students to interact with real life situations and build systems.

Owing to the unprecedented growth in the field of information and technology, demand for candidates with BCA degree has risen considerably. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers and its applications, but finding a solution requires both computer science expertise and knowledge of the particular application domain. Computer science has a wide range of specialties. These include Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Mathematical and Statistical Analysis, Data Science, Computational Science, and Software Engineering. Universities and other HEIs introduced programmes of computer application. Information Technology is growing rapidly. Increasing applications of computers in almost all areas of human endeavour has led to vibrant industries with concurrent rapid change in technology. Unlike other basic disciplines, developing core competency in this discipline that can be reasonably stable becomes a challenge.

In India, it was initially introduced at the Master (postgraduate) level as MCA and M.Tech. Later on, engineering programmes such as B.Tech and B.E in Computer Science & Engineering and in Information Technology were introduced in various engineering College/Institutions to cater to the growing demand for trained engineering manpower in IT industries. Parallelly, BCA, BSc and MSc programmes with specialization in Computer Science were introduced to train manpower in this highly demanding area. BCA and BCA (Hons) are aimed at undergraduate level training facilitating multiple career paths. Students so graduated, can take up postgraduate programmes in CS or MCA leading to research as well as R&D, can be employable at IT industries, or can pursue a teaching profession or can adopt a business management career.

BCA and BCA (Hons) aims at laying a strong foundation of computer application at an early Curriculum for BCA Program of Davangere University, Davangere (DUD) as per NEP 2020 w.e.f. 2021-22. There are several employment opportunities and after successful

completion of BCA, graduating students can fetch employment directly in companies as programmer, Web Developer, Software Engineer, Network Administrator, Data Scientist, or AI/ML personnel.

The Program outcomes in BCA are aimed at allowing flexibility and innovation in design and development of course content, in method of imparting training, in teaching learning process and in assessment procedures of the learning outcomes. The emphasis in BCA courses, in outcome-based curriculum framework, help students learn solving problems, accomplishing IT tasks, and expressing creativity, both individually and collaboratively. The proposed framework will help Students learn programming techniques and the syntax of one or more programming languages. All students must, therefore, have access to a computer with a modern programming language installed. The computer science framework does not prescribe a specific language. The teacher and students will decide which modern programming languages students will learn. More importantly, students will learn to adapt to changes in programming languages and learn new languages as they are developed.

The current Curriculum Framework for BCA degrees is intended to assist the students to achieve the following.

- To develop an indulgent and knowledge of the basic theory of Computer Science and Information Technology with good foundation on theory, systems and applications such as algorithms, data structures, data handling, data communication and computation
- To develop the capability to use this knowledge to analyse new situations in the application domain
- To attain necessary and state-of-the-art skills to take up industry challenges. The objectives and outcomes are carefully designed to suit to the abovementioned purpose.
- The ability to synthesize the acquired knowledge, understanding and experience for a better and improved comprehension of the real-life problems Curriculum for BCA Program of DUD as per NEP 2020 w.e.f. 2021-22.
- To learn skills and tools like mathematics, statistics and electronics to find the solution, interpret the results and make predictions for the future developments
- To formulate, to model, to design solutions, procedure and to use software tools to solve real world problems and evaluate Curriculum for BCA Program of DUD as per NEP 2020 w.e.f. 2021-22.

PROGRAM OUTCOMES:

At the end of the BCA programme, the students will be able to understand, analyze and develop computer programs/applications using efficient data structures and algorithms, web designs and networking. Few of the extended outcomes will be achieved by the students are listed below:

1. **Discipline knowledge:** Acquiring knowledge on basics of Computer Science and ability to apply to design principles in the development of solutions for problems of varying complexity

2. **Problem Solving:** Improved reasoning with strong mathematical ability to Identify, formulate and analyse problems related to computer science and exhibiting a sound knowledge on data structures and algorithms.

3. **Design and Development of Solutions:** Ability to design and development of algorithmic solutions to real world problems and acquiring a minimum knowledge on statistics and optimization problems. Establishing excellent skills in applying various design strategies for solving complex problems.

4. **Programming a Computer:** Exhibiting strong skills required to program a computer for various issues and problems of day-to-day applications with thorough knowledge on programming languages of various levels.

5. **Application Systems Knowledge:** Possessing a sound knowledge on computer application software and ability to design and develop app for applicative problems.

6. **Modern Tool Usage:** Identify, select and use a modern scientific and IT tool or technique. for modelling, prediction, data analysis and solving problems in the area of Computer Science and making them mobile based application software.

7. **Communication:** Must have a reasonably good communication knowledge both in oral and writing.

8. **Project Management:** Practicing of existing projects and becoming independent to launch own project by identifying a gap in solutions.

9. **Ethics on Profession, Environment and Society:** Exhibiting professional ethics to maintain the integrality in a working environment and also have concern on societal impacts due to computer-based solutions for problems. Curriculum for BCA Program of DUD as per NEP 2020 w.e.f. 2021-22.

10. **Lifelong Learning:** Should become an independent learner. So, learn to learn ability.

11. **Motivation to take up Higher Studies:** Inspiration to continue educations towards advanced studies on Computer Science.

By the end of the program the students will be able to:

The Bachelor of Computer Application (BCA (Hons)) program enables students to attain following additional attributes besides the afore-mentioned attributes, by the time of graduation:

1. Apply standard Software Engineering practices and strategies in real -time software project development.
2. Enabling Design and Development of computer programs/computer - based systems in the areas related to AI, algorithms, networking, web design, cloud computing, IoT and data analytics.
3. Acquaint with the contemporary trends in industrial/research settings and thereby innovate novel solutions to existing problems.
4. The ability to apply the knowledge and understanding noted above to the analysis of a given information handling problem.
5. The capability to work independently on a substantial software project and as an effective team member.

PROGRAM STRUCTURE

Proposed Curricular and Credits Structure under Choice Based Credit System [CBCS] of BCA with Discipline Scheme for the Four Years Computer Application BCA Undergraduate Honors Programme with effect from 2021-22

SEMESTER-1											
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L-1	21BCA1L1LK1	Kannada	40	60	100	4	0	0	3	3
	21BCA1L1LFK1	Functional Kannada								
L-2	21BCA1L2LEN2	English	40	60	100	4	0	0	3	3
	21BCA1L2LHI2	Hindi								
	21BCA1L2LSN2	Sanskrit								
	21BCA1L2LTE2	Telugu								
	21BCA1L2LUR2	Urdu								
DSC1	21BCA1C1L	Programming in C	40	60	100	3	0	0	3	3
	21BCA1C1P	C Programming Lab	25	25	50	0	0	4	2	3
DSC2	21BCA1C2L	Fundamentals of Computers	40	60	100	3	0	0	3	3
	21BCA1C2P	Computer Fundamentals Lab	25	25	50	0	0	4	2	3
DSC3	21BCA1C3LMF	Mathematical Foundation*	40	60	100	3	0	0	3	3
	21BCA1C3LAC	Accountancy*								
OEC1	21BCA1O1CPL	C Programming Concepts	40	60	100	3	0	0	3	3
SEC1	21BCA1S1FD	Digital Fluency**	25	25	50	1	0	2	2	2
VBC1	21BCA1V1PE1	Physical Education - Yoga	25	-	25	-	-	2	1	-
VBC2	21BCA1V2HW	H&W/NCC/NSS/R&R/ Cultural	25	-	25	-	-	2	1	-
Total Marks					**800**	**Semester Credits**			**26**	

Note: 1. *The students who have studied Mathematics at PUC or Diploma have to opt Accountancy and who have studied Accountancy at PUC and ITI students have to opt Mathematics.

Registrar
Davangere University
Shivagangotri, Davangere

SEMESTER-2											
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L-3	21BCA2L3LK2	Kannada	40	60	100	4	0	0	3	3
	21BCA2L3FKL2	Functional Kannada								
L-4	21BCA2L4EN2	English	40	60	100	4	0	0	3	3
	21BCA2L4HI2	Hindi								
	21BCA2L4SN2	Sanskrit								
	21BSC2L4TE2	Telugu								
	21BCA2L4UR2	Urdu								
DSC4	21BCA2C4L	Data Structures using C	40	60	100	3	0	0	3	3
	21BCA2C4P	Data Structures Lab	25	25	50	0	0	4	2	3
DSC5	21BCA2C5L	Object Oriented Concepts using Java	40	60	100	3	0	0	3	3
	21BCA2C5P	JAVA Lab	25	25	50	0	0	4	2	3
DSC6	21BCA2C6L	Discrete Mathematics	40	60	100	3	0	0	3	3
OEC2	21BCA2O2MPL	Web Designing	40	60	100	3	0	0	3	3
AECC1	21BCA2AE1L	Environmental Studies	20	30	50	1	0	2	2	2
VBC3	21BCA2V3PE2	Physical Education – Sports	25	-	25	-	-	2	1	-
VBC4	21BCA2V4NC1	H&W/NCC/NSS/R&R/ Cultural	25	-	25	-	-	2	1	-
Total Marks					**800**	**Semester Credits**			**26**	

Exit Option

Choice Based Credit System [CBCS] of BCA with Discipline Scheme for the Four Years Computer Application BCA

Undergraduate Honors Programme with effect from 2021-22

Sl. No	Years	After Completion of	Exit Option
1.	First	I and II Semesters	UG Certificate Course in Computer Applications
2.	Second	III and IV Semesters	UG Diploma in Computer Applications
3.	Third	V and VI Semesters	BCA
4.	Fourth	VII and VIII Semesters	BCA (Honors)

Concept Note, Abbreviation Explanation and Coding:

Concept Note:

1. CBCS is a mode of learning in higher education which facilitates a student to have some freedom in selecting his/her own choices, across various disciplines for completing a UG/PG program.
2. A credit is a unit of study of a fixed duration. For the purpose of computation of workload as per UGC norms the following is mechanism be adopted in the university:
   One credit (01) = One Theory Lecture (L) period of one (1) hour.
   One credit (01) = One Tutorial (T) period of one (1) hour.
   One credit (01) = One practical (P) period of two (2) hours.
3. Course: paper/subject associated with AECC, DSC, DSEC, SEC, VBC, OEC, VC, IC, MIL.
4. Wherever there is a practical there will be no tutorial and vice-versa.
5. Vocational course is a course that enables individual to acquire skills set that are required for a particular job.
6. Internship is a designated activity that carries some credits involving more than 25 days of working in an organization (either in same organization or outside) under the guidance of an identified mentor. Internship shall be an integral part of the curriculum.
7. OEC: For non- Computer Science students. Computer Science students have to opt for OEC from departments other than their disciplines.

Abbreviation Explanations:

1. AECC: Ability Enhancement Compulsory Course.
2. DSC: Discipline Specific Core Course.
3. DSEC: Discipline Specific Elective Course.
4. SEC: Skill Enhancement Course.

5. VBC: Value Based Course.
6. OEC: Open/Generic Elective Course.
7. VC: Vocational Course.
8. IC: Internship Course
9. L1: Language One.
10. L2: MIL.
11. L= Lecture; T= Tutorial; P=Practical.
12. MIL= Modern Indian Language; English or Hindi or Telugu or Sanskrit or Urdu.

Program Coding:

1. Code 21: Year of Implementation.
2. Code BCA: BCA Program under the faculty of Applied Science of the University.
3. Code 1: First Semester of the Program, (2 to 6 represent higher semesters).
4. Code A: AECC, (C for DSC, S for SEC, V for VBC and O for OEC).
5. Code 1: First "AECC" Course in semester, similarly in remaining semester for such other courses.
6. Code LK: Language Kannada, FK for Functional Kannada, similarly Language English, Language Hindi, Language Sanskrit, &Language Urdu.
7. Code 1: Course in that semester.

COURSE-WISE SYLLABUS

Year	I	**Course Code:**21BCA1C1L	**Credits**	03
Sem.	I	**Course Title:** Programming in C	**Hours**	42
Course Pre-requisites, if any		NA		
Formative Assessment Marks: 40		Summative Assessment Marks: 60	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course the student should be able to: 1. Read, understand and trace the execution of programs written in C language 2. Apply programming control structures for a given problem to create C code 3. Understand derived datatypes and develop C code using arrays/ strings 4. Understand user defined functions and datatypes to develop C code		

Unit No.	Course Content	Hours
Unit I	**Introduction to C Programming:** Overview of C; History and Features of C; Structure of a C Program with Examples; Creating and Executing a C Program; Compilation process in C. **C Programming Basic Concepts:** C Character Set; C tokens - keywords, identifiers, constants, and variables; Data types; Declaration & initialization of variables; Symbolic constants. **Input and output with C:** Formatted I/O functions - printf and scanf, control stings and escape sequences, output specifications with printf functions; Unformatted I/O functions to read and display single character and a string - getchar, putchar, gets and puts functions.	10
Unit II	**C Operators & Expressions:** Arithmetic operators; Relational operators; Logical operators; Assignment operators; Increment & Decrement operators; Bitwise operators; Conditional operator; Special operators; Operator Precedence and Associatively; Evaluation of arithmetic expressions; Type conversion. **Control Structures:** Decision making Statements - Simple if, if_else, nested if_else, else_if ladder, Switch Case, goto, break & continue statements; Looping. Statements - Entry controlled and exit controlled statements, while, do-while, for loops, Nested loops.	10

Unit III	**Derived data types in C:** Arrays: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays - Declaration, Initialization and Memory representation. **Strings:** Declaring & Initializing string variables; String handling functions - strlen, strcmp, strcpy and strcat; Character handling functions - toascii, toupper, tolower, isalpha, isnumeric etc. **User Defined Functions:** Need for user defined functions; Format of C user defined functions; Components of user defined functions - return type, name, parameter list, function body, return statement and function call; Categories of user defined functions - With and without parameters and return type.	10
Unit IV	**User defined data types:** Structures - Structure Definition, Advantages of Structure, declaring structure variables, accessing structure members, Structure members initialization, comparing structure variables, Array of Structures; Unions - Union definition; difference between Structures and Unions. **Pointers in C:** Understanding pointers - Declaring and initializing pointers, accessing address and value of variables using pointers; Pointers and Arrays; Pointer Arithmetic; Advantages and disadvantages of using pointers;	12
Recommended Learning Resources		
Print Resources	1. C: The Complete Reference, By Herbert Schildt. 2. C Programming Language, By Brain W. Kernighan 3. Kernighan & Ritchie: The C Programming Language (PHI) 4. P. K. Sinha &Priti Sinha: Computer Fundamentals (BPB) 5. E. Balaguruswamy: Programming in ANSI C(TMH) 6. Kamthane: Programming with ANSI and TURBO C (Pearson Education) 7. V. Rajaraman: Programming in C (PHI –EEE) 8. S. Byron Gottfried: Programming with C(TMH) 9. Yashwant Kanitkar: Let us C 10. P.B. Kottur: Programming in C (Sapna Book House)	

Year	I	**Course Code:** 21BCA1C1P	**Credits**	02
Sem.	I	**Course Title:** Lab: C Programming	**Hours**	52
Course Pre-requisites, if any:		NA		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Part A:

1. Program to read radius of a circle and to find area and circumference
2. Program to read three numbers and find the biggest of three
3. Program to compare library functions of math.h with user user defined function.
4. Program to generate the factorial of a given number
5. Program to generate n fibonacii sequence
6. Program to read a number, find the sum of the digits, reverse the number and check it for palindrome
7. Program to read numbers from keyboard continuously till the user presses 999 and to find the sum of only positive numbers
8. Program to read percentage of marks and to display appropriate message (demonstration of switch Case statement)
9. Program to find the roots of quadratic equation (Demonstration of else-if ladder)
10. Program to read marks scored by a student and find the sum, average and result using switch case.
11. Program to remove Duplicate Element in a single dimensional Array
12. Program to find GCD of two integers using function

Part B:

1. Program to perform all bitwise operations on two integers.
2. Program to read a string and to find the number of alphabets, digits, vowels, consonants, spaces and special characters.
3. Program to find length and Reverse of a string without using built in function.
4. Program to read, display and to find the trace and norm of a matrix in order M X N
5. Program to find first and second largest element in an array.
6. Program to perform addition and subtraction of Matrices

	7. Program to read, display and multiply two m x n matrices using functions 8. Program to check a given number is prime or not by user defining isprime( ) function 9. Program to demonstrate student structure to read & display records of n students. 10. Program to demonstrate the concept of nested structure. 11. Program to swap two integers using call-by-value and call-by-reference. 12. Program to implement the concept of dynamic memory allocation(malloc(), calloc(), realloc(), free())

Note: Student has to execute a minimum of 10 programs in each part to complete the Lab course

Semester I

Year	I	Course Code: 21BCA1C2L	Credits	03
Sem.	I	Course Title: Fundamentals of Computers	Hours	42
Course Pre-requisites, if any:		NA		
Formative Assessment Marks: 40		Summative Assessment Marks: 60	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course the student should be able to: 1. Create an awareness of computers its classification and anatomy 2. Understand Number systems , Computer Languages and the steps for problem solving 3. Understand the fundamentals of operating systems and basic commands 4. Understand basic concepts of DBMS and Internet		

Unit No.	Course Content	Hours
Unit I	**Fundamentals of Computers:** Introduction to Computers - Computer Definition, Characteristics of Computers, Evolution and Generations of Computers, Basic Organization of a Digital Computer; Functions & Components of a Computer, Central Processing Unit, Microprocessor, Storage units, Input and output Devices. How CPU and memory works. Classification of Digital Computer Systems: Microcomputers, Minicomputers, Mainframes, Super computers	10
Unit II	**Number Systems** – different types, conversion from one number system to another; Computer Codes – BCD, Gray Code, ASCII; Boolean Algebra – Boolean Operators with Truth Tables; **Computer Languages** - Machine Level, Assembly Level & High Level Languages, Translator Programs – Assembler, Interpreter and Compiler; Planning a Computer Program – Algorithm and Flowchart with Examples.	10
Unit III	**Operating System Fundamentals:** Operating Systems: Introduction, Functions of an operating System, Classification of Operating Systems, System programs, Application programs, Utilities, The Unix Operating System, Basic Commands (cal, date, bc, echo, who, ls, pwd, cd, mkdir, rmdir), Commands to work with file (cat, cp, rm, mv, file, wc, head, tail)	10
Unit IV	**Introduction to Database Management Systems:** Database, DBMS, Why Database -File system vs DBMS, Database applications, Database users, Introduction to	12

	SQL, Classification of SQL-DDL, DML, DCL. **Internet Basics:** Introduction, Features of Internet, Internet application, Services of Internet, Logical and physical addresses, Internet Service Providers, Domain Name System. **Web Basics:** Introduction to web, web browsers, http/https, URL.	
Print Resources	1. Pradeep K. Sinha and Priti Sinha: Computer Fundamentals (Sixth Edition), BPB Publication 2. David Riley and Kenny Hunt, Computational thinking for modern solver, Chapman &Hall/CRC, 3. J. Glenn Brook shear," Computer Science: An Overview", Addision-Wesley, Twelth Edition, 4. R.G. Dromey, "How to solve it by Computer", PHI,	

Year	I	**Course Code:** 21BCA1C2P	**Credits**	02
Sem.	I	**Course Title:** Computer Fundamentals Lab	**Hours**	52
Course Pre-requisites, if any:		NA		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03hrs.	
		Part A: Hardware 1. Identification of the peripherals of a computer, components in a CPU and their functions. 2. Assembling and disassembling the system hardware components of personal computer. 3. Basic Computer Hardware Trouble shooting. 5. LAN and WiFi Basics. 6. Operating System Installation – Windows OS, UNIX/LINUX, Dual Booting. 7. Installation and Uninstallation of Software – Office Tools, Utility Software (like Anti-Virus, System Maintenance tools); Application Software - Like Photo/Image Editors, Audio Recorders/Editors, Video Editors ...); Freeware, Shareware, Payware and Trialware; Internet Browsers, Programming IDEs, 8. System Configuration – BIOS Settings, Registry Editor, MS Config, Task Manager, System Maintenance, Third-party System Maintenance Tools (Similar to CCleaner and Jv16 PowerTools)		
		Part B: Software 1. Activities using Word Processor Software 2. Activities using Spreadsheets Software 3. Activities using Presentation Software 4. Activities involving Multimedia Editing (Images, Video, Audio) 5. Tasks involving Internet Browsing 6. Flow charts: Installation and using of flowgarithms software for different arithmetic tasks like sum, average, product, difference, quotient and remainder of given numbers, calculate area of Shapes (Square, Rectangle, Circle and Triangle), arrays and recursion. **Note:** Use any open source software to execute the above assignments.		

Reference:

1. Computational Thinking for the Modern Problem Solver, By Riley DD, Hunt K.A CRC press, 2014

2. Ferragina P, Luccio F. Computational Thinking: First Algorithms, Then Code. Springer

Web References:

http://www.flowgorithm.org/documentation/

Year	I	Course Code: 21BCA1C3LMF	Credits	03
Sem.	I	Course Title: Mathematical Foundation	Hours	42
Course Pre-requisites, if any	NA			
Formative Assessment Marks: 40	Summative Assessment Marks:60	Duration of ESA: 03 hrs.		
Course Outcomes	At the end of the course the student should be able to: 1. Study and solve problems related to connectives, predicates and quantifiers under different situations. 2. Develop basic knowledge of matrices and to solve equations using Cramer's rule. 3. Know the concept of Eigen values. 4. To develop the knowledge about derivatives and know various applications of differentiation. 5. Understand the basic concepts of Mathematical reasoning, set and functions			

Unit No.	Course Content	Hours
Unit I	**Basic concepts of set theory:** Mathematical logic introduction statements Connectives - negation, Conjunction, disjunction statement formulas and truth tables- conditional and bi Conditional statements- tautology contradiction- equivalence of formulas-duality law-Predicates and Quantifiers, Arguments.	10
Unit II	**Operations on sets**: power set- Venn diagram Cartesian product-relations - functions- types of functions - composition of functions.	10
Unit III	**Matrix algebra:** Introduction-Types of matrices-matrix operations- transpose of a matrix - determinant of matrix - inverse of a matrix-Cramer's rule. **Matrix:** finding rank of a matrix - normal form-echelon form Cayley Hamilton theorem-Eigen values	12
Unit IV	**Differential calculus:** Functions and limits - Simple Differentiation of Algebraic Functions – Evaluation of First and Second Order Derivatives – Maxima and Minima	10

Recommended Learning Resources

Print Resources	1. P. R. Vittal-Business Mathematics and Statistics, Margham Publications, Chennai B. S. Vatsa-Discrete Mathematics –New Age International Limited Publishers, New Delhi

Year	I	Course Code: 21BCA1C3LAC	Credits	03
Sem.	I	Course Title: Accountancy	Hours	42
Course Pre-requisites, if any	NA			
Formative Assessment Marks: 40	Summative Assessment Marks: 60		Duration of ESA: 03 hrs.	
Course Outcomes	At the end of the course the student should be able to: 1. Study and understand Accounting, systems of Book, Branches of accounting advantage and limitations 2. Know the concept of accounting, financial accounting process and Journalization 3. Maintenance different account book and reconciliations 4. Preparations of different bills, and trial balance.			

Unit No.	Course Content	Hours
Unit I	**Introduction:** History and Development of Accounting, Meaning, Objectives and functions of Accounting, Book keeping V/s Accounting, Users of accounting data, systems of book keeping and accounting, branches of accounting, advantages and limitations of accounting	10
Unit II	**Accounting Concepts and Convention:** Meaning, need and classification, accounting standards meaning, need and classification of Indian accounting standards. Accounting principles V/s accounting standard.Financial Accounting Process: Classification of accounting transactions and accounts, rules of debit and credit as per Double Entry System. Journalization and Ledger posting.	10
Unit III	**Preparation of Different Subsidiary Books:** Purchase Day book Sales Day Book, Purchase Returns Day Book, Sales Returns Day Book, Cash Book. Bank Reconciliation Statement: Meaning, Causes of Difference, Advantages, Preparation of Bank Reconciliation Statements.	10
Unit IV	**Account Procedure:** Honor of the Bill, Dishonor of the Dill, Endorsement, Discounting, Renewal, Bill for collection, Retirement of the Bill, Accommodation Bills, Bill Receivable Book and Payable Book. Preparation of Trial Balance: Rectification of errors and Journal Proper.**Preparation of Final Accounts:** Meaning, need and classification, Preparation of Manufacturing, Trading, Profit and loss account and Balance – Sheet of sale-traders and partnership firms.	12

Recommended Learning Resources

Print Resources	**Reference Books:** 1. S. Ramesh, B.S. Chandrashekar, A Text Book of Accountancy. 2. V.A. Patil and J.S. Korlahalli, Book – keeping and accounting, (R. Chand and Co.Delhi). 3. R.S.Singhal, Principles of Accountancy,(Nageen Prakash pvt.Lit.Meerut). 4. M.B.Kadkol, Book–Keeping and Accountancy, (Renuka Prakashan, Hubil) 5. Vithal, Sharma: Accounting for Management, Macmillan Publishers, Mumbai. 6. B B.S. Raman, Accountancy, (United Publishers, Mangalore). 7. Tulsian, Accounting and Financial Management – I:Financial Accounting – Person Education

Open Elective Courses offered by the Department of Computer Science for other discipline

Open Elective 1: C Programming Concepts

Year	I	Course Code: 21BCA1O1CPL	Credits	03
Sem.	1	Course Title: C Programming Concepts	Hours	42
Course Pre-requisites, if any		NA		
Formative Assessment Marks: 40		Summative Assessment Marks: 60	Duration of ESA:.03 hrs.	

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- Confidently operate Desktop Computers to carry out computational tasks
- Understand working of Hardware and Software and the importance of operating systems
- Understand programming languages, number systems, peripheral devices, networking, multimedia and internet concepts
- Read, understand and trace the execution of programs written in C language
- Write the C code for a given problem
- Perform input and output operations using programs in C
- Write programs that perform operations on arrays

Course Content

Content	Hours
Unit - 1	
Fundamentals of Computers: Introduction to Computers -Hardware, software- System software, Application software, Utility software, Operating System; Computer Languages - Machine Level, Assembly Level & High-Level Languages, Translator Programs – Assembler, Interpreter and Compiler; Planning a Computer Program – Algorithm and Flowchart with Examples.	6Hrs
Unit - 2	

Introduction to C Programming: Over View of C; History and Features of C; Structure of a C Program with Examples; Creating and Executing a C Program; Compilation process in C. **C Programming Basic Concepts:** C Character Set; C tokens - keywords, identifiers, constants, and variables; Data types; Declaration & initialization of variables; Symbolic constants. **Input and output with C:** Formatted I/O functions - *printf* and *scanf*, control stings and escape sequences, output specifications with *printf* functions; Unformatted I/O functions to read and display single character and a string - *getchar, putchar, gets* and *puts* functions.	10 Hrs
Unit - 3	
C Operators & Expressions: Arithmetic operators; Relational operators; Logical operators; Assignment operators; Increment & Decrement operators; Bitwise operators; Conditional operator; Special operators; Operator Precedence and Associatively; Evaluation of arithmetic expressions; Type conversion. **Control Structures:** Decision making Statements - *Simple if, if_else, nested if_else, else_if ladder,* Switch-*case, goto, break* & *continue* statements; Looping Statements - Entry controlled and Exit controlled statements, *while, do-while, for* loops, Nested loops.	8 Hrs
Unit - 4	
User Defined Functions: Need for user defined functions; Format of C user defined functions; Components of user defined functions - return type, name, parameter list, function body, return statement and function call; Categories of user defined functions - With and without parameters and return type.	8 Hrs
Unit 5:	
Arrays: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays - Declaration, Initialization and Memory representation. Strings: Declaring & Initializing string variables; String handling functions - *strlen, strcmp, strcpy and strcat;* Character handling functions - *toascii, toupper, tolower, isalpha, isnumeric* etc. **Basics of Pointers in C:** Understanding pointers - Declaring and initializing pointers, accessing address and value of variables using pointers; Pointer Arithmetic; Advantages and disadvantages of using pointers;	10Hrs

Text Books:

1. Pradeep K. Sinha and Priti Sinha: Computer Fundamentals (Sixth Edition), BPB Publication
2. E. Balgurusamy: Programming in ANSI C (TMH)

References:

1. Kamthane: Programming with ANSI and TURBO C (Pearson Education)
2. V. Rajaraman: Programming in C (PHI –EEE)
3. S. ByronGottfried: Programming with C (TMH)
4. Kernighan & Ritche: The C Programming Language (PHI)
5. Yashwant Kanitkar: Let us C
6. P.B. Kottur: Programming in C (Sapna Book House)

Semester: II

Year	I	**Course Code:** 21BCA2C4L	**Credits**	03
Sem.	2	**Course Title:** Data Structures using C	**Hours**	42
Course Pre-requisites, if any		Knowledge of Programming		
Formative Assessment Marks: 40		Summative Assessment Marks: 60	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course the student should be able to: * Understand the classification of data structures and dynamic memory allocation * Understand the difference between iteration and recursion and apply recursive definition for problem solving * Understand and evaluate the applications of stacks and queues * Understand and evaluate the applications of linked lists and tree		

Unit No.	**Course Content**	**Hours**
Unit I	**Introduction to data structures:** Definition; Types of data structures - Primitive & Non-primitive, Linear and Non-linear; Operations on data structures. **Dynamic memory allocation:** Static & Dynamic memory allocation; Memory allocation and de-allocation functions - malloc, calloc, realloc and free.	10
Unit II	**Stacks:** Basic Concepts – Definition and Representation of stacks; Operations on stacks – Push, Pop; Applications of stacks; Infix, postfix and prefix notations; Conversion from infix to postfix using stack; Evaluation of postfix expression using stack; Application of stack in function calls. **Recursion:** Definition; Types of recursions; Recursion Technique Examples - GCD, Binomial coefficient nCr, Towers of Hanoi; Comparison between iterative and recursive functions.	10
Unit III	**Queues:** Basic Concepts – Definition and Representation of queues; Types of queues, - Simple queues, Circular queues, Double ended queues, Priority queues; Operations on Simple queues;	10

	Sorting and Searching: Arrays as abstract data types, Representation of linear arrays in memory, Traversing linear arrays; Inserting and deleting elements; Sorting – Selection sort, Bubble sort, Quick sort, Selection sort, Insertion sort; Searching - Sequential Search, Binary search; Iterative and Recursive searching	
Unit IV	**Linked list:** Basic Concepts – Definition and Representation of linked list, Types of linked lists - Singly linked list, Doubly linked list, Circular linked list Doubly Circular Linked list; Representation of Linked list in Memory; Operations on Singly linked lists – Traversing, Searching, Insertion, Deletion; **Trees:** Definition; Tree terminologies –node, root node, parent node, ancestors of a node, siblings, terminal & non-terminal nodes, degree of a node, level, edge, path, depth; **Binary tree:** Type of binary trees - strict binary tree, complete binary tree, binary search tree and heap tree; Array representation of binary tree. Traversal of binary tree; preorder, in order and post order traversal.	12
Recommended Leaning Resources		
Print Resources	**Reference Books:** 1. Ellis Horowitz and Sartaj Sahni: Fundamentals of Data Structures 2. Tanenbaum: Data structures using C (Pearson Education) 3. Kamathane: Introduction to Data structures (Pearson Education) 4. Y. Kanitkar: Data Structures Using C(BPB) 5. Kottur: Data Structure Using C 6. Padma Reddy: Data Structure Using C	

Year	I	**Course Code:** 21BCA2C4P	**Credits**	02
Sem.	II	**Course Title:** Lab: Data Structures	**Hours**	52
Course Pre-requisites, if any:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Part A:

1. Program to find GCD using recursive function
2. Program to generate binomial coefficient using recursive function.
3. Program to implement Towers of Hanoi using recursion.
4. Program to read the names of cities and arrange them alphabetically.
5. Program to sort the given list using selection sort technique.
6. Program to sort the given list using bubble sort technique.
7. Program to sort the given list using insertion sort technique.
8. Program to implement the stack operations.
9. program to implement the operations of linear queue

Part B:

1. Program to sort the given list using quick sort technique.
2. Program to sort the given list using merge sort technique.
3. Program to search an element using linear search technique.
4. Program to search an element using binary search technique.
5. Program to convert an infix expression to postfix.
6. Program to implement circular queue.
7. Program to implement the operations of singly linked list.
8. Program to implement the operations of circular linked list
9. Program to construct BST and implement tree traversal.

Year	I	**Course Code:** 21BCA2C5L	**Credits**	03
Sem.	II	**Course Title:** Object Oriented Programming with JAVA	**Hours**	42
Course Pre-requisites, if any		Knowledge of Programming		
Formative Assessment Marks: 40		Summative Assessment Marks: 60	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course the student should be able to: 1. Understand the features of Java and the architecture of JVM 2. Write, compile, and execute Java programs that may include basic data types and control flow constructs and how type casting is done 3. Identify classes, objects, members of a class and relationships among them needed for a specific problem and demonstrate the concepts of polymorphism and inheritance 4. The students will be able to demonstrate programs based on interfaces and threads and explain the benefits of JAVA's Exceptional handling mechanism compared to other Programming Language 5. Write, compile, execute Java programs that include GUIs and event driven programming and also programs based on files		

Unit No.	Course Content	Hours
Unit I	**Introduction to Java:** OOPs concepts, Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Arrays in java. **Objects and Classes:** Basics of objects and classes in java, Methods and objects, Instance of operator, Visibility modifiers, Method Overloading, Constructors, Static Members, Inbuilt classes like String, Character, String Buffer, this reference.	12
Unit II	**Inheritance and Polymorphism:** Inheritance in java, Super and sub class, Types of inheritance, Overriding, Polymorphism, Dynamic binding, Abstract class, Interface in java, Packages in java - defining and importing user defined packages.	10
Unit III	**Exception handling:** Exception mechanism with try catch-finally.	10

	Multithreading in java: Thread life cycle and methods, Runnable interface, Thread priorities	
Unit IV	**Event and GUI programming:** Event handling in java, Event types, Mouse and key events, GUI Basics, Panels, Frames, Layout Managers: Flow Layout, Border Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, TextFields, Text Areas, Combo Boxes, Lists, Windows, Menus.	10
Recommended Learning Resources		
Print Resources	**Reference Books:** 1. Java, By E Balagurusamy – A Primer, Fourth Edition, Tata McGraw Hill Education Private Limited. 2. Core Java Volume I – Fundamentals, By Cay S. Horstmann, Prentice Hall 3. Object Oriented Programming with Java : Somashekara, M.T., Guru, D.S., Manjunatha,K.S 4. Java 2 - The Complete Reference – McGraw Hill publication. 5. Java - The Complete Reference, 7th Edition, By Herbert Schildt– McGraw Hill publication.	

Year	I	**Course Code:** 21BCA2C5P	**Credits**	02
Sem.	II	**Course Title:** Lab: JAVA	**Hours**	52
Course Pre-requisites, if any:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Part A: Programming Lab – Java Fundamentals – OOPS in JAVA

1. Program to assign two integer values to X and Y. Using the 'if' statement the output of the program should display a message whether X is greater than Y.
2. Program to list the factorial of the numbers 1 to 10. To calculate the factorial value, use while loop. (Hint: Fact of 4 = 4*3*2*1)
3. Program to find the area and circumference of the circle by accepting the radius from the user.
4. Program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.
5. Program to perform mathematical operations. Create a class called AddSub with methods to add and subtract. Create another class called MulDiv that extends from AddSub class to use the member data of the super class. MulDiv should have methods to multiply and divide A main function should access the methods and perform the mathematical operations.
6. Program with class variable that is available for all instances of a class. Use static variable declaration. Observe the changes that occur in the object's member variable values.
7. Program to create a student class with following attributes; Enrollment No: Name, Mark of sub1, Mark of sub2, mark of sub3, Total Marks. Total of the three marks must be calculated only when the student passes in all three subjects. The passing mark for each subject is 50. If a candidate fails in any one of the subjects his total mark must be declared as zero. Using this condition write a constructor for this class. Write separate functions for accepting and displaying student details. In the main method create an array of three student objects and display the details.

	8. Write a program to demonstrate multiple inheritance and use of Implementing Interfaces 9. Illustrate creation of thread by a) Extending Thread class. b) Implementing Runnable Interfaces 10. Write a program to demonstrate multilevel inheritance using abstract class. 11. Create a package 'BCA' in your current working directory. a. Create a class student in the above package with the following attributes: Name, age, gender. Include appropriate constructor and a method for displaying the details. b. Import above package and access the member variables and function contained in a package.
	PART B: Exception Handling & GUI Programming 1. Program to catch Negative Array Size Exception. This exception is caused when the array size is initialized to negative values. 2. Program to demonstrate exception handling with try, catch and finally. 3. Program which create and displays a message on the window 4. Program which creates a frame with two buttons father and mother. When we click the father button the name of the father, his age and designation must appear. When we click mother button similar details of mother also appear. 5. Create a frame which displays your personal details with respect to a button click 6. Program to create a window with TextFields and Buttons. The "ADD" button adds the two integers and display the result. The "CLEAR" button shall clear all the text fields. 7. Program to create a window, when we press M or m, the window displays "good morning", A or a, the window display's Good Afternoon" , E or e, the window displays "good morning", N or n, the window displays "good morning" 8. Demonstrate the various mouse handling events using suitable example. 9. Program to create menu bar with label name 10. Program to create menu and pull-down menus.

Year	I	**Course Code:** 21BCA2C6L	**Credits**	03
Sem.	II	**Course Title:** Discrete Mathematics	**Hours**	42
Course Pre-requisites, if any		NA		
Formative Assessment Marks: 40		Summative Assessment Marks: 60	Duration of ESA: 03 hrs.	
Course Outcomes		1. At the end of the course the student should be able to: 2. To understand the basic concepts of Mathematical reasoning, set and functions. 3. To understand various counting techniques and principle of inclusion and exclusions. 4. Understand the concepts of various types of relations, partial ordering and 5. Equivalence relations. 6. Apply the concepts of generating functions to solve the recurrence relations. 7. Familiarize the fundamental concepts of graph theory and shortest path algorithm		

Unit No.	Course Content	Hours
Unit I	**The Foundations:** Logic and proofs: Propositional Logic, Applications of Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy. **Basic Structures:** Sets, Functions, Sequences, Sums, and Matrices: Sets, set operations, Functions, Sequences and Summations, matrices.	10
Unit II	**Counting:** Basics of counting, Pigeonhole principle, Permutation and combination, Binomial Coefficient and Combination, Generating Permutation and Combination. **Advanced Counting Techniques:** Applications of Recurrence Relations, Solving Linear Recurrence, Relations, Divide and Conquer Algorithms and Recurrence Relations, Generating functions, Inclusion-Exclusion, Applications of Inclusion-exclusion.	10
Unit III	**Induction and Recursion:** Mathematical Induction, Strong Induction and Well- Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Corrections.	12

	Relation: Properties of relation, Composition of relation, Closer operation on relation, Equivalence relation and partition. Operation on relation, Representing relation.	
Unit IV	**Graphs**: Graphs and Graph models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring..	10
Recommended Learning Resources		
Print Resources	**Reference Books:** 1. Discrete Mathematics and Its Applications, Kenneth H. Rosen: Seventh Edition, 2012. 2. Discrete Mathematical Structure, Bernard Kolman, Robert C, Busby, Sharon Ross, 2003. 3. Graph Theory with Applications to Engg and Comp. Sci: Narsingh Deo-PHI1986. 4. Discrete and Combinatorial Mathematics Ralph P. Grimaldi, B. V. Ramatta, Pearson, Education, 5Edition. 5. Discrete Mathematical Structures, Trembley and Manobar.	

Open Elective Courses offered by the Department of Computer Science for other discipline

Open Elective 2: Web Designing

Year	I	Course Code: 21BCA2O2MPL	Credits	03
Sem.	II	Course Title: Web Designing	Hours	42
Course Pre-requisites, if any		NA		
Formative Assessment Marks: 40		Summative Assessment Marks: 60	Duration of ESA:.03 hrs.	

Course Outcomes (COs):

- Be familiar with different web design theories and terminology.
- Analyze a web page and identify its elements and attributes.
- Create web pages using XHTML and Cascading Style Sheets.
- Build dynamic web pages using JavaScript (Client-side programming).

Unit – 1	
Fundamentals: Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, the Web Programmers Toolbox. Introduction to XHTML: Basic syntax, Standard structure, Basic text markup, Images, Hypertext Links, Lists, Tables, Forms, Frames	8 Hrs
Unit-2	
Cascading Style Sheet (CSS): Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The and tags, Conflict resolution.	8 Hrs
Unit-3	
The Basics of JavaScript: Overview of JavaScript, Object orientation and JavaScript, Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions, Errors in scripts, Examples.	10Hrs
Unit-4	

JavaScript and HTML Documents: The JavaScript execution environment, The Document Object Model, Element access in JavaScript, Events and event handling, handling events from the Body elements, Button elements, Text box and Password elements, The DOM 2 event model, the navigator object, DOM tree traversal and modification.	8 Hrs
Unit-5	
Dynamic documents with JavaScript: Introduction, positioning elements, Moving elements, Element visibility, Changing colors and fonts, Dynamic content, Stacking elements, locating the mouse cursor, Reacting to a mouse click, slow movement of elements, Dragging and dropping elements.	8 Hrs

Text Books:

1. Robert W. Sebesta: Programming the World Wide Web,4th Edition, Pearson Education, 2008.

References:

1. M. Deitel, P.J. Deitel, A. B. Goldberg: Internet & World Wide Web How to Program, 4th Edition, Pearson Education, 2004.
2. Chris Bates: Web Programming Building Internet Applications,3rd Edition, Wiley India, 2007.
3. Xue Bai et al: The web Warrior Guide to Web Programming, Cengage Learning, 2003.
4. M Srinivasan: Web

Evaluation Scheme for Lab Examination

Assessment Criteria		25 marks
Program – 1 from Part A	Writing the Program	05
	Execution and Formatting	05
Program -2 from Part B	Writing the Program	05
	Execution and Formatting	05
Viva Voice		05
Total		**25**

ASSESSMENT METHODS

Evaluation Scheme for Internal Assessment:

Theory:

Assessment Criteria	40 marks
1st Internal Assessment Test for30 marks 1 hr 30 min after 8 weeks and 2nd Internal Assessment Test for 30 marks 1 hr 30 min after 15 weeks. Average of two tests should be considered.	30
Attendance >75%	05
Assignment	05
Total	**40**

Practical:

Assessment Criteria	25 marks
Semester End Internal Assessment Test for 15 marks 2 hrs	15
Attendance >75%	05
Journal (Practical Record)	05
Total	**25**

BCA Question Paper Pattern

Time: 3 Hrs **PART-A** **Max. Marks. 60**

Answer any Five questions. **5X2=10**

1.

2.

3.

4

5

6

7

8

Note: Two questions from each unit.

PART-B

Answer any Five of the following questions. 5X4=20

1

2

3

4

5

6

7

8

Note: Two questions from each unit.

PART-C

Answer any Three of the following questions. 3X10=30

1

2

3

4

5

Note: Minimum One question from each unit.

BOS Chairman
Dept. of Computer Science
Davangere University
Shivagangotri, Davangere