



***KOLHAN UNIVERSITY***

Chaibasa, Jharkhand, India

Proposed Syllabus for  
Four Year Undergraduate Programme (FYUGP)  
of

***Bachelor of Computer Application  
(BCA)***

***Semester - 1***

With Effect From  
Academic Year 2022 - 2023

As Per Revised Curriculum and Credit Framework for the  
FYUGP under the provisions of NEP - 2020

## Table of Contents

Serial Number	Topics	Page Number
1	Course Structure (Semester – I) for Four Year Undergraduate Programme (FYUGP) of Bachelor of Computer Applications (BCA)	03 – 03
2	Programme Outcomes and Programme Specific Outcomes of Bachelor of Computer Application (BCA)	04 – 04
3	MN–1A (Th): Introduction to Computer Science	05 – 06
4	MN–1A (Pr): Introduction to Computer Science Lab	07 – 08
5	MJ–1 (Th): C Programming Language	09 – 10
6	MJ–1 (Pr): C Programming Language Lab	11 – 12

**Course Structure (Semester – I) for Four Year Undergraduate Programme (FYUGP)  
of  
Bachelor of Computer Applications (BCA)**

Sem.	Paper Code	Paper Title	L–T–P	Credits	Contact Hours
<b>I</b>	AEC–1	Language and Communication Skills: Hindi Composition		2	
	VAC–1	Value Added Course–1 <i>(Two papers are to be selected by the students from the available options)</i>		2+2=4	
	SEC–1	Digital Education		3	
	MDC–1	Multi-Disciplinary Course–1 <i>(To be selected by the students from the available options)</i>		3	
	MN–1A	Th: Introduction to Computer Science	3–0–1	3+1=4	45+30=75
		Pr: Introduction to Computer Science Lab			
MJ–1	Th: C Programming Language	3–0–1	3+1=4	45+30=75	
	Pr: C Programming Language Lab				

**Abbreviations:**

**Th** (Theory), **Pr** (Practical), **L–T–P** (Lecture–Tutorial–Practical), **AEC** (Ability Enhancement Course), **VAC** (Value Added Course), **SEC** (Skill Enhancement Course), **MDC** (Multi Disciplinary Course), **MN–1** (Minor From Discipline–1), **MN–2** (Minor From Vocational Studies/Discipline–2), **IAP** (Internship/Apprenticeship/Project), **MJ** (Major Disciplinary/Interdisciplinary Courses)

## **Programme Outcomes and Programme Specific Outcomes of Bachelor of Computer Application (BCA)**

### **Programme Outcomes:**

1. Acquire knowledge of Computer application theory and algorithm principles in the design and modeling of computer based system.
2. Understand the computing concepts and their applications using the acquired board based knowledge.
3. To provide thorough understanding of nature, scope and application of computer and computer languages.
4. Identify and analyze software application problems in multiple aspect including coding, testing and implementation in industrial applications.
5. The program prepares the young professional for a range of computer applications, computer organization, and techniques of Computer Networking, Software Engineering, Web development, Database management and Advance Java.

### **Programme Specific Outcomes:**

1. To pursue further studies to get specialization in Computer Science and Applications, Economics, Mathematics, Business Administration.
2. To pursue the career in corporate sector can opt for MBA, MCA, etc.
3. To Work in the IT sector as programmer, system engineer, software tester, junior programmer, web developer, system administrator, software developer, etc.
4. To work in public sector undertakings and Government organizations.
5. Ability to understand the changes or future trends in the field of computer application.
6. Encouraging students to convert their start-up idea to reality by implementing.
7. Students will able to understand, analyze and develop computer programs in the areas related to algorithm, system software, web design and networking for efficient design of computer-based system.

**MN-1A (Th): Introduction to Computer Science**

3 Credits | 45 Minimum Class Hours | Semester I

**Objectives:**

This course provides an overview of introductory concepts about computers, number systems and components of computer system. It builds the foundation of the computer application courses.

**Learning Outcomes:**

After completion of this course, a student will be able to–

- Handle a computer system for day-to-day use.
- Enumerate different types of input/output devices and types of memory.
- Perform conversion between different number systems including binary addition and subtraction.
- Familiarize Operating Systems, Programming languages, Network and Internet
- Differentiate between system and application software.
- Prepare documents, spreadsheets, and presentations.
- Use Email services.

**Outline of the Course**

Minimum Class Hours		Exam Time (Hours)		Credits		Marks								
						Semester Internal		End Semester		Full Mark		Pass Marks		Total Marks
Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th + Pr
45	30	3	3	3	1	15	N/A	60	25	75	25	30	10	75+25=100

Unit	Topic	Minimum Class Hours
I	Computer Basics and Number System	12
II	Input/Output Devices	05
III	Computer Software and Programming Language	16
IV	Network and Internet	12
<b>Total</b>		<b>45</b>

**Detailed Syllabus****Unit I: Computer Basics and Number System**

**Computer Basics:** Introduction, Characteristics of Computers, Generation of Computers, Classification of Computers, Applications of Computers, Functional Block Diagram of Computer {Central Processing Unit, Input Unit, Output Unit, Memory Unit (Primary Storage Unit and Secondary Storage Unit), Bus Structure}.

**Number System:** Binary Number System, Octal Number System, Decimal Number System, Hexadecimal Number System, Conversion from One Number System to Another, Binary addition and subtraction.

**Unit II: Input/Output Devices**

Introduction, Input Devices (Keyboard, Pointing Devices, Speech Recognition, Digital Camera, Scanners, Optical Scanners), Output Devices {Hard Copy Output Devices (Printers, Plotters, Computer Output Microfilm (COM)), Soft Copy Output Devices (Monitors, Speakers, Projectors)}.

### Unit III: Computer Software and Programming Language

**Computer Program:** Introduction, Developing a Program, Algorithm, Flowchart, and Pseudo code.

**Programming Language:** Introduction, Evolution of Programming Languages, Classification of Programming Languages, Generations of Programming Languages, Features of a Good Programming Language, Selection of a Programming Language.

**Computer Software:** Introduction, Categories of Computer Software - System Software (Operating System, System Utility, Language Translators etc.) and Application Software (Word processor, Spreadsheet, Presentation, Database software etc.)

### Unit IV: Network and Internet

Introduction to Network, Types of Networks, Introduction to Internet, ISP, URL, IP Address, Web Page, Web Site, Web Server, Web Browser, Internet Services (WWW and Electronic Mail - Creating E-Mail account, sending and receiving E-Mails).

#### Recommended Books:

- Anita Goel; **Computer Fundamentals**; Pearson
- ITL Education Solution Limited, R&D Wing; **Introduction to Computer Science**; Pearson Education

#### Further readings:

- CI Stens School of Computing, **Internet and Introduction**, TMH
- Rajaraman V., **Fundamental of Computers**, Prentice Hall of India Pvt. Ltd., New Delhi
- Peter Nortorns, **Introduction to Computer**, TMH



**MN-1A (Pr): Introduction to Computer Science Lab**

1 Credit | 30 Minimum Class Hours | Semester I

**Objectives:**

The main objectives of the course are as follows–

- To use standard word, spreadsheets, and presentation packages.
- To use Email services.
- To understanding computer hardware.
- To learning basic application software tools.

**Learning Outcomes:**

After completion of this course, a student will be able to–

- Handle a computer system for day-to-day including web browsing.
- Prepare documents, spreadsheets, and presentations.
- Use Email services.

**Outline of the Course**

Minimum Class Hours		Exam Time (Hours)		Credits		Marks								
						Semester Internal		End Semester		Full Mark		Pass Marks		Total Marks
Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th + Pr
45	30	3	3	3	1	15	N/A	60	25	75	25	30	10	75+25=100

**Marks Distribution of End Semester Practical Examination**

- Experiments – 15 Marks
- Viva-Voce – 05 Marks
- Practical File – 05 Marks

**Experiment List****Unit I: Word Processor**

1. Prepare your Curriculum Vitae using various formatting tools.
2. Using mail merge prepare an interview Call Letter.
3. Create an index page for the Practical File using table having columns such as Sl. No, Experiment Name, Date of Experiment, Page No., Remarks etc. Use formatting tools, Table Layout option, Table Design options etc.

**Unit II: Spreadsheet**

4. Create a database of students, which contains marks obtained by students of a class in different subjects and then calculate maximum, minimum, average and sum of marks in each subject. Also calculate the percentage of total marks of each student using functions and formulas. Sort the data on the basis of percentage of total marks column in descending order.
5. Draw effective charts (Pie Chart, Bar Graph etc.) to present students' data as given in Experiment No. 4.

**Unit III: Presentation**

6. Make a presentation on your college. Use various effects, animation and transitions etc. (Minimum 5 slides).

**Unit IV: Internet**

7. Create an e-mail id in any e-mail service provider website such as gmail, yahoo mail, hotmail etc.
- (i) Write a mail to a software company for the post of Software Engineer. Send your resume as an attachment. Also send a copy of mail to your teacher as a blind carbon copy (BCC).
  - (ii) Read the unread mails present in your mailbox.
  - (iii) Delete the unnecessary mails from the inbox.
  - (iv) Sign out from your email account.
8. Search the official website of Kolhan University in Google search engine using any web browser. Open the official website of Kolhan University from Google search results and download the course **MN-1A (Introduction to Computer Science)** of FYUGP Syllabus.

*Note: Additional lab assignments may be included based on topics covered in the theory paper*



## MJ-1 (Th): C Programming Language

3 Credits | 45 Minimum Class Hours | Semester I

### Objectives:

This course helps the students in understanding a powerful, portable and flexible structured programming language which is suitable for both systems and applications programming. It is a robust language which contains a rich set of built-in functions and operators to write any complex program.

### Learning Outcomes:

After completion of this course, a student will be able to–

- Understand and use the process of abstraction using a programming language such as ‘C’.
- Analyze step by step and develop a program to solve real world problems.

### Outline of the Course

Minimum Class Hours		Exam Time (Hours)		Credits		Marks								
						Semester Internal		End Semester		Full Mark		Pass Marks		Total Marks
Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th + Pr
45	30	3	3	3	1	15	N/A	60	25	75	25	30	10	75+25=100

Unit	Topic	Minimum Class Hours
I	C Fundamentals	08
II	Control Structures and C Preprocessor	10
III	Arrays, Strings, Pointers, and Functions	17
IV	Structures and File Handling	10
<b>Total</b>		<b>45</b>

### Detailed Syllabus

#### Unit I: C Fundamentals

History, Structures of ‘C’ Programming, Function as Building Blocks, Character Set, Tokens, Keywords, Identifiers, Variables, Constant, Data Types, Comments.

**Operators:** Types of Operators, Precedence and Associativity, Expression, Statement and Types of Statements.

**Built-in Functions:** Console I/O Functions, Mathematical Functions, and Character Functions.

#### Unit II: Control Structures and C Preprocessor

**Control Structures:** Sequence Structure, Selection Structure (if Statement, if-else Statement, if-else-if Statement, Nested if-else Statement, switch-case Statement), Loop Structure (while, do-while, for Loop, Nested Loop), Other Statements (break, continue, goto).

**C Preprocessor:** Types of C Preprocessor Directives.

#### Unit III: Arrays, Strings, Pointers, and Functions

**Arrays:** One–Dimensional Arrays (Definition, Declaration, Initialization, Accessing and Displaying Array Elements, Passing Array to a Function), Two–Dimensional Arrays (Definition, Declaration, Initialization, Accessing and Displaying Array Elements).

**Strings:** Definition, Declaration, Initialization, Standard Library Functions.

**Pointers:** Definition, Declaration, Initialization, Indirection Operator, Address of Operator, Operations on Pointers, Array of Pointers, Dynamic Memory Allocation.

**Functions:** Declaration and Definition, Function Call, Types of Function, Parameter Passing (Call by Value, Call by Reference), Scope of Variables, Storage Classes (Automatic, Register, Extern, Static Variables), Recursive Function.

#### Unit IV: Structures and File Handling

**Structures:** Definition, Declaration, Initialization, Accessing Structure Elements, Array of Structures, Pointers and Structures, Passing Structures to a Function.

**File Handling:** Introduction, Defining and Opening a File, Closing a File, Input/Output Operations on Text and Binary Files, Error Handling During I/O Operation, Random Access to Files.

#### Recommended Books:

- Rajaraman V., **Computer Programming in C** (Second Edition), New Delhi: Tata McGraw-Hill Publication, 1992
- Kanetkar Y., **Let Us C** (Third Edition), New Delhi: BPB Publications, 1999
- Gottfried, B. S., **Theory and Problems of Programming with C**, New Delhi: Tata McGraw-Hill Publication, 1997
- Balaguruswamy E. **Programming in ANSI C** (Second Edition), New Delhi: Tata McGraw-Hill Publication, 1992

#### Further readings:

- Dennis Ritchie, **The C Programming Language**, New Delhi: Pearson Education
- Forouzah, Ceilberg Thomson, **Structured Programming Approach Using C**, Learning Publication
- Deitel & Deitel, **C How To Program**, New Delhi: Prentice Hall India, 1996
- R. B. Patel, **Fundamental of Computers and Programming in C**, Khanna Book Publishing Company PVT. LTD. Delhi, India, 1st edition, 2008



**MJ-1 (Pr): C Programming Language Lab**

1 Credit | 30 Minimum Class Hours | Semester I

**Objectives:**

This course helps the students in understanding a powerful, portable and flexible structured programming language which is suitable for both systems and applications programming. It is a robust language which contains a rich set of built-in functions and operators to write any complex program.

**Learning Outcomes:**

After completion of this course, a student will be able to–

- Develop modular, efficient and readable C programs by hands-on experience.
- Interpret good profound knowledge in C programming language and enable them to build programs using Control Structures, Arrays, Strings, Pointers, Functions, Structures, and File Handling to solve the real world problems.
- Illustrate memory allocation to variables dynamically and perform operations on text files.

**Outline of the Course**

Minimum Class Hours		Exam Time (Hours)		Credits		Marks								
						Semester Internal		End Semester		Full Mark		Pass Marks		Total Marks
Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th	Pr	Th + Pr
45	30	3	3	3	1	15	N/A	60	25	75	25	30	10	75+25=100

**Marks Distribution of End Semester Practical Examination**

- Experiments – 15 Marks
- Viva-Voce – 05 Marks
- Practical File – 05 Marks

**Experiment List****Unit I: C Fundamentals**

1. Write a program to evaluate the arithmetic expression  $((A + B / C * D - E) * (F - G))$ . Read the values of A, B, C, D, E, F, and G from the standard input device.
2. Write a program to check whether a number is even or odd using ternary (or, conditional) operator.
3. Write a program to find the size of int, float, double and char data type.

**Unit II: Control Structures and C Preprocessor**

4. Write a program to find the largest number among three numbers.
5. Write a program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +, -, \*, /, % and use “switch-case” statement).
6. Write a program to generate Fibonacci series up to  $N^{\text{th}}$  term.
7. Write a program to check whether a number is palindrome or not.
8. Write a program to check whether a number is prime or not.
9. Define a macro that receives an array and the number of elements in the array as arguments. Write a program for using this macro to print the elements of the array.

### Unit III: Arrays, Strings, Pointers, and Functions

10.	Write a program to find the sum of all elements, average of all elements, and the second largest element in a “One–Dimensional” integer array.
11.	Write a program that lets the user perform arithmetic operations on two “Two–Dimensional” integer arrays. Your program must be menu driven, allowing the user to select the operations (e.g., Press 1 for Addition and Press 2 for Multiplication).
12.	Write a program to reverse a given string and then count the number of vowels, consonants and spaces in that reversed string.
13.	Write a program that lets the user perform string operations on standard library functions. Your program must be menu driven, allowing the user to select the operations (e.g., Press 1 to demonstrate the usage of function “ <b>strlen()</b> ”, Press 2 to demonstrate the usage of function “ <b>strlwr()</b> ”, Press 3 to demonstrate the usage of function “ <b>strupr()</b> ”, Press 4 to demonstrate the usage of function “ <b>strcat()</b> ”, and so on).
14.	Write a program to concatenate two given strings and find the length of the resultant string using pointer.
15.	Write a program to read and print an integer array. The program should input total number of elements (limit) and elements in array from user. Use dynamic memory allocation to allocate (i.e., “ <b>malloc()</b> ” or “ <b>calloc()</b> ” functions) and deallocate (i.e., “ <b>free()</b> ” function) array memory.
16.	Write program that use function to return the greatest common divisor of two given integers.
17.	Write a program to print the transpose of a given matrix using function.
18.	Write a program to generate Fibonacci series up to N <sup>th</sup> term using recursive function.

### Unit IV: Structures and File Handling

19.	<p>Write a program that lets the user perform arithmetic operations on two complex numbers. Define a structure that will hold the data for a complex number. Your program must be menu driven, allowing the user to select the operations (+, -, and *) and input the complex numbers. Furthermore, your program must consist of following functions:</p> <ul style="list-style-type: none"> <li>(i) Function “<b>showChoice()</b>”: This function shows the options to the user and explains how to enter data.</li> <li>(ii) Function “<b>add()</b>”: This function accepts two complex number structures as arguments and returns a complex number structure with the sum of the two complex numbers.</li> <li>(iii) Function “<b>subtract()</b>”: This function accepts two complex number structures as arguments and returns a complex number structure with the difference of the two complex numbers.</li> <li>(iv) Function “<b>multiply()</b>”: This function accepts two complex number structures as arguments and returns a complex number structure with the product of the two complex numbers.</li> </ul>
20.	Write a program to create a text file named “ <b>MyInfo.txt</b> ”, open it, type–in some information about yourself. Read and count the number of characters in the file. And finally, copy the contents of the file to another file named “ <b>MyInfo_Copy.txt</b> ” and display the contents of this file.

*Note: Additional lab assignments may be included based on topics covered in the theory paper.*

