

EC-106 FUNDAMENTALS OF COMPUTER PROGRAMMING

Course Title	Fundamentals Of Computer Programming	Course Code	EC-106
Year	I	Course Group	CORE
Teaching Scheme in Hrs (L: T:P)	4:1:0	Credits	3.0
Methodology	Lecture+ Tutorials	Total Contact Hours	150
CIE	60 Marks	SEE	40 Marks

Pre requisites

Basic knowledge Of Mathematics, Logical, Reasoning and IT skills

Course Outcomes

Upon completion of the course, the student shall be able to

CO1	Comprehend basics of Computers and Programming concepts
CO2	Apply fundamental C programming concepts to develop simple programs using Datatypes and Operators, Decision Making Branching and looping statements in C
CO3	Create Arrays, know the usage of strings ,Analyze and implement Modular Programming using Functions in C
CO4	Apply fundamental Python programming concepts to develop simple programs using variables, data types, operators, decision-making statements, and loop control structures by setting up an appropriate execution environment.
CO5	Use Functions, Strings and Data structures in python and handle exceptions
CO6	Develop classes, modules and packages, Process files, use regular expressions, install of modules and packages

CO-PO Mapping Table

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	1	1	1	0	1
CO2	3	3	2	2	0	0	1
CO3	3	3	3	2	0	0	1
CO4	3	2	2	3	1	0	1
CO5	3	2	2	3	1	0	1
CO6	3	2	2	3	0	1	2

Course Content and Blue Print of Marks for SEE

Unit No	Unit Name	Periods	Questions to be set for SEE			
			R		U	A
I	Introduction To Computer And Programming Concepts	20	Q4	Q1	Q9(a)	Q13(a)
II	Introduction To C Programming Language	30				
III	User defined data types and Functions	30		Q2	Q10(a)	Q14(a)
IV	Introduction To Python Programming Language	20	Q3			
V	Functions, Strings and Data Structures Error Handling	25		Q5,Q6	Q9(b),Q11(a), Q11(b)	Q13(b),Q15(a), Q15(b)
VI	Develop Classes, Packages In Python	25		Q7,Q8	Q10(b),Q12(a), Q12(b)	Q14(b),Q16(a) , Q16(b)
	Total	150	8		8	8

Course Contents

UNIT- 1. INTRODUCTION TO COMPUTER AND PROGRAMMING CONCEPTS

Duration: 20 periods (L:15– T:5)

Basic components of a computer system: Hardware: CPU, memory, input, output and storage units and binary representation of data, Software: Application Software, System Software

Use of Computers for problem solving, Introduction to algorithm - Algorithms for simple problems - Introduction to flow chart –Symbols used in flowchart - Flowcharts for simple problems - Differentiate algorithm and flowchart.

Define Computer Program and programming language - Define High level language and low level language- Define Compiler and interpreter.

Common Programming terminology: Source Code, Text editors and IDEs. Errors and Debugging, Comments and Documentation.

Popular programming languages, their strengths and weaknesses. Programming languages for Electronics.

UNIT-2. INTRODUCTION TO C PROGRAMMING LANGUAGE

Duration: 30 Periods (L:22 – T:8)

I. Introduction to C programming language -importance of C language --structure of C program-
pre processor statements - Comments in C - steps in executing the C program.-

Keywords - Constants and Variables – Simple data types- declaration of a variable - Assigning values to variables

Define operator-Classify operators - arithmetic operators – relational operators - logical operators - various assignment operators –Increment and decrement operators

Define expression – Evaluation of expressions -formatted input and output.

II. Flow Control statements of C

Decision making in C programming - simple if statement – if-else statement – nested if statement - else-if ladder –Switch statement –

Classification of various loop statements- while loop – do-while loop - for loop - break and continue statements - nesting of loops-Comparisons of different loop statements.

UNIT-3. USER DEFINED DATA TYPES AND FUNCTIONS

I. Arrays and Strings

Duration: 30 Periods(L:22 – T:8)

Arrays – definition, types, declaration and initialization of One Dimensional -Array -Accessing the elements in the Array -accessing array out of bounds problem — average of elements in an array - Declaration and initialization of two Dimensional Arrays - sample programs on matrix addition and matrix multiplication, transpose-

Strings – Definition, Declaration and initialization of String variables - Reading of strings from terminal - writing strings to screen- String handling functions with sample programs

II. User defined functions

Function – Definition - Need for user defined functions - Advantages of functions – elements of function - Return values and their types - function prototype - function call – function definition – Scope, visibility and lifetime of variables in functions

Functions with no arguments and no return values - functions with arguments with no return values -functions with arguments with return values - functions with no arguments with return values- Recursion - sample programs on recursion - passing arrays to functions.

UNIT-4. INTRODUCTION TO PYTHON PROGRAMMING LANGUAGE

Duration: 20 Periods(L:15 – T:5)

Introduction to python programming language, versions of python, Python IDLE. Variable declaration and initialization, comments, indentation, Expressions, Assignment Statement, Arithmetic Operators, Operator Precedence, Boolean Expression-Input and Output

Conditional statements in Python: if-else statement Nested-if statement and Elif statement in Python. Loops in Python: while loop including, For Loop, Nested Loops, Break and Continue Statements

UNIT-5. FUNCTIONS, STRINGS AND DATA STRUCTURES

Duration: 25 Periods(L:17 – T:8)

Data types & Python Data Structure, Mutable and Immutable data types, Tuples, Lists, Sets, Dictionaries

Strings: Length of the string and perform Concatenation. Indexing and Slicing of Strings.

Functions: Defining & Calling a function, passing arguments to function, Different types of arguments, Recursion, Scope of variables

Exception handling: Different Types of errors- Exception handling

UNIT-6. DEVELOP CLASSES, PACKAGES IN PYTHON

Duration: 25 Periods (L: 17– T:8)

Define Class- data member, methods and create an instance of class-Working with files and folders, Regular expressions - Familiarize with package installation with PIP- List Popular libraries of Python with their uses - functions of numpy, pandas, matplotlib, sklearn.

Text Books

1. Let Us C -- Yeshwanth Kanetkar BPB Publications
2. Programming in ANSIC -- E.BalaguruswamyTataMcGrawHill
3. Introduction to Computation and Programming Using Python. John V. Guttag, theMIT Press.

Reference Books

1. The C Programming Language- Brian W. Kernighan,DennisM.Ritchie- Pre Publicationntice Hall.
2. Programming in C 3rd Edition-Sams Publishing- Steven G. Kochan- Sams Publishing.
3. Expert C Programming: Deep Secrets-Peter Van Der Linden- Pearson Publications.
4. James Payne, Beginning Python using Python 2.6 and Python 3, Wrox publishing.
5. Paul Gries, Practical Programming: An Introduction to Computer Science using python.
6. The Pragmatic Bookshelf, 2nd edition (4 Oct. 2013).
7. Charles Dierach, Introduction to Computer Science using Python.

Suggested E-learning references

C Language

1. https://www.w3schools.com/c/c_intro.php?external_link=true
2. <https://www.geeksforgeeks.org/c-programming-language/>
3. <https://www.programiz.com/c-programming>
4. <https://www.javatpoint.com/c-programming-language-tutorial>
5. <https://www.freecodecamp.org/news/the-c-programming-handbook-for-beginners/>
6. <https://www.tutorialspoint.com/cprogramming/index.htm>
7. <https://archive.nptel.ac.in/courses/106/104/106104128/>

Python

1. <https://www.python.org/about/gettingstarted/>
2. <https://www.w3schools.com/python/>
3. <https://www.programiz.com/python-programming>
4. <https://www.tutorialspoint.com/python/index.htm>
5. <https://realpython.com/start-here/>
6. <https://www.codecademy.com/learn/learn-python>
7. <https://www.dataquest.io/blog/learn-python-the-right-way/>

Suggested Learning Outcomes

Upon completion of the course, the student shall be able to

CO-1 Comprehend basics of Computers and Programming concepts

- 1.1 List basic components of a computer system
- 1.2 Define CPU, memory, input, output and storage units
- 1.3 Explain binary representation of data
- 1.4 Differentiate Application Software and System Software
- 1.5 State different steps involved in problem solving
- 1.6 Define algorithm and know its characteristics
- 1.7 State the steps involved in algorithm development
- 1.8 Develop algorithms for simple problems.
- 1.9 Define flowchart and know symbols used in drawing flowcharts
- 1.10 Draw flowchart for simple problems.
- 1.11 Differentiate algorithm and flowchart
- 1.12 Define Computer Program and programming language
- 1.13 Define High level language and low level language
- 1.14 Define Compiler and interpreter
- 1.15 Comprehend Common Programming terminology: Source Code, Text editor and IDEs. Errors and Debugging, Comments and Documentation.
- 1.16 List Popular programming languages, their strengths and weaknesses.
- 1.17 List Programming languages for Electronics

CO-2: C programming concepts to develop simple programs using Datatypes and Operators, Decision Making Branching and looping statements in C

- 2.1 State the importance of C language
- 2.2 Explain the structure of C program
- 2.3 List and Explain different stages in converting the C program source code to executable code.
- 2.4 Explain the steps involved in executing the C program
- 2.5 Describe Comments in C
- 2.6 Discuss #include and #define Pre-processor directives
- 2.7 Describe Keywords, Constants and Variables
- 2.8 Classify Primary data types and explain with examples
- 2.9 Write the syntax to declare and assign values to variables.
- 2.10 Classify and explain operators with examples
- 2.11 Describe operator precedence and associativity of operators.
- 2.12 Define expression.
- 2.13 Describe evaluation of Arithmetic, Relational and logical expressions.
- 2.14 Illustrate type conversion (implicit and explicit) techniques.
- 2.15 Define Header file and discuss stdio.h, conio.h, math.h, string.h Header files.
- 2.16 Illustrate getchar(), putchar(), scanf(), printf() with programs.
- 2.17 Discuss decision making in programming
- 2.18 Explain decision making statements- if, if-else, nested if-else, else-if ladder – with syntax and programs.
- 2.19 Explain switch statement with syntax and programs.
- 2.20 Define looping and list loop statements.
- 2.21 Discuss the iterative statements – while, do- while, for statement- with syntax and programs.
- 2.22 Discuss break and continue statements.
- 2.23 Define Nested loops.
- 2.24 Develop programs using Nested loops.
- 2.25 Compare different loop statements.

CO-3: Create Arrays, know the usage of strings ,Analyze and implement Modular Programming using Functions in C

- 3.1 Define an Array.
- 3.2 Declare and initialize One Dimensional Array with an example program.
- 3.3 Develop a program to read, access and display the elements of a one dimensional array
- 3.4 Develop a c program to count the even and odd numbers for a given array of n elements.
- 3.5 Declare and initialize two Dimensional Arrays with an example program.
- 3.6 Illustrate the concept of two dimensional arrays with the following programs
 - a. Matrix addition, subtraction
 - b. Matrix multiplication
- 3.7 Define String
- 3.8 Declare and initialize a String variable.
- 3.9 Outline the different ways of reading and writing strings using scanf(), getchar(), gets(), printf(), putchar(), puts() functions.

- 3.10 Create a program to concatenate, copy, and finding out the length of a string
- 3.11 Explain about various String handling functions with sample programs.
- 3.12 Develop C programs using different string handling functions.
- 3.13 Define function,
- 3.14 Discuss the need of functions.
- 3.15 List out the advantages of functions.
- 3.16 Identify the elements of function.
- 3.17 Define and discuss function prototype, function call, and function definition with syntax and program.
- 3.18 Discuss the scope, visibility and lifetime of variables in functions.
- 3.19 Illustrate functions with no arguments and no return values with syntax and example programs.
- 3.20 Illustrate functions with no arguments and with return values with syntax and example programs.
- 3.21 Illustrate functions with arguments and with no return values with syntax and example programs.
- 3.22 Illustrate functions with arguments and with return values with syntax and example programs.
- 3.23 Define recursion.
- 3.24 Develop the following programs using recursion
 - a. Factorial of a number.
 - b. compute x^n (hint: x raised to the power n).

CO-4:Apply fundamentals of Python programming concepts to develop simple programs

- 4.1 State the importance of python programming language
- 4.2 List different versions of python
- 4.3 Lists features of Python programming.
- 4.4 Execute Python Script (Command Prompt, Script and IDE) in different ways
- 4.5 Declare and initialize variables
- 4.6 Use Comments, Indentation in programs
- 4.7 Describe Data Types Booleans, Numbers
- 4.8 Explain Operators in python
- 4.9 Write a program using Operators in python
- 4.10 Explain if-else statement, its working and execution
- 4.11 Explain Nested-if statement and execution.
- 4.12 Explain Elif statement and its execution
- 4.13 Write program using conditional statements
- 4.14 List Various Loop Statements with example to solve problems which are iterative
- 4.15 Explain While loop and its working
- 4.16 Explain For loop and its working
- 4.17 Explain Nested loop and its working
- 4.18 Write a program using loop statements
- 4.19 Explain Break and continue statements.
- 4.20 Write a program using Break and continue statements.

CO-5: Develop Functions, Strings and Data structures in python and handle exceptions

- 5.1 Parts of a Function
- 5.2 Explain Defining & Calling a function
- 5.3 Explain passing arguments to functions
- 5.4 Write a program using a function in python.
- 5.5 List different types of arguments
- 5.6 Differentiate Mutable & Immutable Data Types
- 5.7 List Mutable & Immutable Data Types
- 5.8 Demonstrate Recursion(Solve problems by using recursive method of problem solving)
- 5.9 Explain Scope of variables
- 5.10 Process strings using operators and built-in functions
- 5.11 Explain Indexing and Slicing of Strings.
- 5.12 Write a program to perform operation on Lists and Tuples
- 5.13 Write a program to perform operation on Sets, and Dictionaries
- 5.14 Difference between compile time errors, runtime errors and logical errors
- 5.15 List common compile time errors and runtime errors
- 5.16 Explain use of try/except, finally and else block to handle exceptions
- 5.17 Write a program to handle exceptions

CO-6: Develop classes, modules and packages, Process files, use regular expressions, install of modules and packages

- 6.1 Define class with its members and create instances of class
- 6.2 Create and import Modules and Packages
- 6.3 Differentiate local and global variables
- 6.4 Open, close, read, write, append data to files using programs
- 6.5 List modes of opening a file
- 6.6 Delete files and folders
- 6.7 Explain package installation with PIP-
- 6.8 List Popular libraries of Python with their uses
- 6.9 List methods of numpy, pandas, matplotlib, sklearn.
- 6.10 Write programs using standard Mathematical function sqrt, cos, sine, pow, degrees,
- 6.11 Write programs using datetime package in python application
- 6.12 Write programs using numpy package in python application
- 6.13 Write programs using the pandas package in python application.
- 6.14 Write programs using matplotlib package in python application.
- 6.15 Create patterns to use regular expressions.
- 6.16 Validate data using regular expressions.

Suggested Student Activities

1. Student activities like mini-project, quizzes, etc. should be done in groups of 3-5 students.
2. Each group should do any one of the following type of activity or any other similar activity related to the course with prior approval from the course coordinator and programme coordinator concerned.
3. Each group should conduct different activity and no repetition should occur.
4. Design a simple project that automates a task
5. Study the impact of recent technologies on health and environment; prepare a report that addresses the issues and solutions to them.
6. Study Recent Technologies like Data Mining, Data Analysis, and Data Scientist; and write a report that distinguishes these technologies

Sl.No	Description	Level	No of Questions	Marks for each question	Choice	Total Marks
01	Part-A	Remembering(R)	4	1	Nil	4 Marks
02	Part-B	Understanding(U)	4	3	2	6 Marks
03	Part-C	Application(A)	4	5	2	10 Marks
Total Marks						20 Marks

ER EXAMS

MID

**SEM-I
EXAM**

S.No	Unit No	R	U	A	Remarks
1	Unit-I	1,2	5(a)	7(a)	
			5(b)	7(b)	
2	Unit-II	3,4	6(a)	8(a)	
			6(b)	8(b)	
Total Questions		4	4	4	

MID SEM-II EXAM

S.No	Unit No	R	U	A	Remarks
1	Unit-III	1,2	5(a)	7(a)	
			5(b)	7(b)	
2	Unit-IV	3,4	6(a)	8(a)	
			6(b)	8(b)	
Total Questions		4	4	4	

The length of answer for each question framed in respect of Part-A, B &C shall not exceed $\frac{1}{4}$ of a page,1 page and 2 pages respectively

QUESTION PAPER PATTERN FOR SEMESTER END EXAM

Sl.No	Description	Level	No of Questions	Marks for each question	Choice	Total Marks
01	Part-A	Remembering(R)	8	1	Nil	8 Marks
02	Part-B	Understanding(U)	8	3	4	12 Marks
03	Part-C	Application(A)	8	5	4	20 Marks
Total Marks						40 Marks

Unit No	Questions to be set for SEE				
	R		U		A
I	Q4	Q1		Q9(a)	Q13(a)
II					
III		Q2		Q10(a)	Q14(a)
IV					
V		Q3	Q5,Q6	Q9(b), Q11(a), Q11(b)	Q13(b), Q15(a), Q15(b)
VI					
Total Questions	8		8	8	

State Board of Technical Education and Training, Telangana

Model Question paper

DECE I Year

Mid Semester-I Examination

Course Code: EC-106

Duration: 1 hour

Course Name: FUNDAMENTALS OF COMPUTER PROGRAMMING

Max. Marks: 20 Marks

- PART-A

Answer all questions, Each Question carries ONE mark **4x1 = 4 Marks**

1. List any two differences between algorithm and flowchart.
2. Define Low level language
3. List various types of constants.
4. Define an expression

PART-B

Answer TWO questions. Each question carries THREE marks **2x3 = 6 Marks**

5(a). List out any 6 symbols used in Flowchart.

(OR)

5(b). Write any three differences between high level language and low level language

6(a). Write the syntax to declare variables in C? Give examples.

(OR)

6(b). Evaluate an expression $(2+5)*(10-4)\%10$.

PART-C

Answer TWO questions. Each question carries FIVE marks **2x5 = 10 Marks**

7(a). Draw a flowchart to find the factorial of given number N.

(OR)

7(b). Explain the stages to execute a C-Program

8(a). Explain different data types in C with examples.

(OR)

8(b). Elaborate any three categories of operators in C.

State Board of Technical Education and Training, Telangana

Model Question paper

DECE I Year

Mid Semester-II Examination

Course Code: EC-106

Duration: 1 hour

Course Name: FUNDAMENTALS OF COMPUTER PROGRAMMING

Max. Marks: 20 Marks

- PART-A

Answer ALL questions, Each Question carries ONE mark

4x1 = 4 Marks

1. Define an array.
2. Give the syntax for declaring and initializing of one dimensional array
3. List any two operators in python
4. List the features of python programming language.

PART-B

Answer TWO questions. Each question carries THREE marks.

2x3 = 6 Marks

5(a). Declare and initialize a String variable.

(OR)

5(b). Explain any three String handling functions

6(a). Write about operator precedence in python

(OR)

6(b). List features of Python programming.

PART-C

Answer TWO questions. Each question carries FIVE marks

2x5 = 10 Marks

7(a). Write a C program to multiply two matrices.

(OR)

7(b). Write a C program to check whether a given string is palindrome or not.

8(a). Explain in detail about the operators in python?

(OR)

8(b). Explain the different ways of executing a python program

State Board of Technical Education and Training, Telangana

Model Question paper

DECE I Year

Semester End Examination

Course Code: EC-106

Duration: 2 hours

Course Name: FUNDAMENTALS OF COMPUTER PROGRAMMING

Max. Marks: 40 Marks

PART-A

Answer **all** questions. Each Question carries one mark

8x1 = 8 Marks

1. Give the syntax of a simple if statement in C.
2. Write the syntax to create a dictionary type variable
3. Define regular expression
4. List any two advantages of functions.
5. Write syntax of Lists in python?
6. List common compile time errors and runtime errors
7. List modes of opening a file
8. Define module

PART-B

Answer **FOUR** questions. Each question carries **three** marks.

4 x 3 = 12 Marks

9(a). Describe operator precedence and associativity of operators.

(OR)

9(b). Explain Parts of a Function in python.

10(a). Discuss the scope, visibility and lifetime of variables in functions.

(OR)

10(b). 6.8 List Popular libraries of Python with their uses.

11(a). Difference between compile time errors, runtime errors and logical errors

(OR)

11(b). List Mutable & Immutable Data Types

12(a). Differentiate local and global variables

(OR)

12(b). List any 3 methods of numpy

PART-C

Answer **FOUR** questions. Each Question carries **FIVE** marks

4 x 5 = 20 Marks

13(a) Define Header file and discuss stdio.h, conio.h, math.h, string.h Header files.

(OR)

13(b) Write a program to Demonstrate Recursion.

14(a) Illustrate functions with no arguments and with return values with syntax and example programs.

(OR)

14(b) Write a python program to Open, close, read, write, append data to files.

15(a) Write a program to perform operation on Lists and Tuples

(OR)

15(b) Write a python program to handle exceptions

16(a) Write a python programs using standard Mathematical function sqrt, cos, sine, pow, degrees,

(OR)

16(b) Write a python program to Validate data using regular expressions.