

Course Code: 0613-101	Course Title: Structured Programming Languages
Course Type (Core Course/Electives/ GED): Core Course	
Year /Semester: 1 st Semester	Academic Session: 2022-2023
Course Teacher: Mir Aminul Haque	Credit Value: 3
Contact Hours: 3 hours	Total Marks: 100

Rationale: To teach students structured programming so that they can write programs and develop software accordingly.

Course Objectives: The objectives of this course are

1. To familiarize the students with programming concepts and solving technique of computational problems.
2. Help them to know the fundamental data structures and associated algorithms.
3. To facilitate necessary knowledge about error handling technique.
4. Apply the knowledge to design programs involving decision structures, loops and functions.
5. To make students able to analyze the dynamics of memory by the use of pointers and linked-list.

Course Content:

Introduction to Computer Programming, Problem Solving Techniques, Algorithm Specification and Development, Programming Style, Debugging and Testing, Documentation, Program Design Methodologies, Structured and Modular Program Designs.

Programming Languages and Paradigms: Classification, Assembler and Translators' Source and Object Programs, Structured Language, Procedural and Non-procedural Programming, Modular Programming, Object-oriented Programming.

Programming Language in C: Data types, Operators and Conversions, Statements, Control structures, Array, pointers and Strings, Functions, Preprocessor, Arrays of pointers, Structure, Union and Bit-field, External files, History of C, Structure of C Program, Identifiers and Keywords, Data types, Variable Declaration, Expression, Statement, Operators, Library functions, Data Input and Output Functions, Control Statements, Function, Recursion, Automatic, External and Static variable, Array, pointer, Structure and Unions, Data Files, Some Additional Features of C, Advanced Data types, Access Modifiers, Storage Class Specifies, Type Conversion in Assignments, Function Type Modifiers, pointer to Function.

Course Learning Outcomes (CLOs): After learning this course, students will be able to

1. Apply fundamentals of programming to solve substantial problems.
2. Identify various problems and find solutions using C programming constructs.
3. Develop solutions to various problems using C programming constructs.

Mapping of Course Learning Outcomes (CLOs) with Program Learning Outcomes (PLOs):

CLOs	PLOs	Bloom's Taxonomy (Domain/Level)	Teaching-Learning Strategy	Assessment Strategy
CLO1	PLO-A	Cognitive/Apply	Lecture, Demonstration	Test, Demonstration
CLO2	PLO-B	Cognitive/Analyze	Lecture, Demonstration, Notes	Class Test, Assignment
CLO3	PLO-C	Cognitive/Apply, Analyze	Lecture, Demonstration, Notes	Test, Presentation

[illegible]

Mapping Course Learning Outcomes (CLOs) with the Teaching-Learning and Assessment Strategy:

Week	Topic	Teaching Learning Strategy	Assessment Strategy	Corresponding CLOs
1.	Introduction to Computer Programming, Problem Solving Techniques, Algorithm Specification and Development, Programming Style, Debugging and Testing, Documentation.	<ul style="list-style-type: none"> • Lecture • Demonstration 	<ul style="list-style-type: none"> • Q/A • Test (SQ) • Demonstration 	CLO1
2.	Program Design Methodologies, Structured and Modular Program Designs, Classification, Assembler and Translators Source and Object Programs, Structured Language, Procedural and Non-procedural Programming, Modular Programming, Object Oriented Programming.	<ul style="list-style-type: none"> • Lecture • Demonstration 	<ul style="list-style-type: none"> • Q/A • Assignment 	CLO1
3.	Data Types, Operators and Conversions, Statements.	<ul style="list-style-type: none"> • Lecture • Demonstration 	<ul style="list-style-type: none"> • Q/A 	CLO2 CLO3
4.	Control structures (if-else, nested if-else, switch)	<ul style="list-style-type: none"> • Demonstration • Lecture • Text Book 	<ul style="list-style-type: none"> • Demonstration • Presentation • Quiz Test 	CLO1 CLO3
5.	Array and its concepts (1D, 2D and Multi-dimensional array)	<ul style="list-style-type: none"> • Demonstration • Lecture 	<ul style="list-style-type: none"> • Q/A 	CLO2 CLO3
6.	Pointers and Strings, Arrays of Pointers	<ul style="list-style-type: none"> • Lecture • Group work • Cooperative learning 	<ul style="list-style-type: none"> • Q/A • Test (BQ) • Assignment 	CLO2 CLO3
7.	User define functions, recursive function	<ul style="list-style-type: none"> • Lecture • Rapport building • Group work • Demonstration 	<ul style="list-style-type: none"> • Q/A • Presentation 	CLO1 CLO2
8.	Structure, Union and Bit-field	<ul style="list-style-type: none"> • Multimedia Projector • Demonstration 	<ul style="list-style-type: none"> • Q/A • Test (S B) • Assignment 	CLO1 CLO3

	External files	<ul style="list-style-type: none"> • Multimedia Projector • Demonstration 	<ul style="list-style-type: none"> • Q/A 	CLO1 CLO3
10.	Advanced data types, Access Modifiers, Storage Class Specifiers, Type Conversion in Assignments, Function Type Modifiers, Pointer to Function	<ul style="list-style-type: none"> • Rapport building • Multimedia Projector • Group work 	<ul style="list-style-type: none"> • Test (S B) 	CLO1 CLO2
11.	System Software, The Role of BIOS, Language Translators, Text Editor, The Tasks of an OS, OS Characteristics, Types of OS, Linux, UNIX, MS DOS, Windows.	<ul style="list-style-type: none"> • Lecture • Demonstration 	<ul style="list-style-type: none"> • Test (S B) • Demonstration • Assignment 	CLO1 CLO2
12.	Basic problem solving technique	<ul style="list-style-type: none"> • Lecture • Demonstration 	<ul style="list-style-type: none"> • Q/A • Demonstration 	CLO1
13.	Discussion about contest related problems	<ul style="list-style-type: none"> • Lecture • Rapport building • Demonstration 	<ul style="list-style-type: none"> • Assignment • Demonstration 	CLO1 CLO2
14.	Advance problem solving technique	<ul style="list-style-type: none"> • Rapport building • Demonstration 	<ul style="list-style-type: none"> • Q/A • Test (S B) 	CLO2 CLO3
15.	Revisions, Q and A	<ul style="list-style-type: none"> • Lecture • Group work 	<ul style="list-style-type: none"> • Q/A • Demonstration 	CLO1

References:

1. Herbert Schildt, "C: The Complete Reference", Osborne / McGraw-Hill, 4th Edition, July 1, 2017.
2. Dennis Ritchie Brian W. Kernighan, "The C Programming Language ", Prentice Hall, 1st Edition, January 1, 2015.
3. Balagurusamy, "Programming in ANSI C", McGraw-Hill Education, 2nd Edition, January 1, 2011.
4. K. N. King, "C Programming: A Modern Approach", W. W. Norton and Company, 2nd Edition, April 19, 2008.
5. Behrouz A. Forouzan, Richard F. Gilberg, "Computer Science: A Structured Programming Approach Using C", Course Technology, 3rd Edition, February 6, 2006.
6. Paul J. Deitel, "C How to Program", Prentice Hall, 5th Edition, September 4, 2006.
7. Stephen G. Kochan, "Programming in C", Sams, 3rd Edition, July 18, 2004.

Course Code: 0613-102	Course Title: Structured Programming Languages Lab
Course Type (Core Course/Electives/ GED): Core Course	
Year /Semester: 1 st Semester	Academic Session: 2022-2023
Course Teacher: Mir Aminul Haque	Credit Value: 1
Contact Hours: 2 hours	Total Marks: 50

Rationale: To teach students structured programming so that they can write programs and solves substantial problems in C.

Course Objectives: The objectives of this course are

1. Know the differences between different types of data storage and why each is needed.
2. Understand how data is transferred from RAM to permanent storage including the use of arrays.
3. Learn how the control structures of sequence, repetition, and condition can be used to process data.
4. Understand the various parts of a C program and how modular structures are introduced.
5. Create a small project that must reflect all of the knowledge gathered throughout the course.

Course Content:

The teacher will specify the Lab experiments based on 0613-102 and make a list of experiments which will be given to the students at the beginning of the class with necessary documents.

Course Learning Outcomes (CLOs): After learning this course, students will be able to

1. Apply basic programming concepts to solve substantial problems in C.
2. Evaluate basic technical documents, presentations, and group interactions using appropriate tools.
3. Prepare themselves for national and international programming contests.

Mapping of Course Learning Outcomes (CLOs) with Program Learning Outcomes (PLOs):

CLOs	PLOs	Bloom's Taxonomy (Domain/Level)	Teaching-Learning Strategy	Assessment Strategy
CLO1	PLO-A	Cognitive/Apply	Lecture, Demonstration	Q/A, Lab Report
CLO2	PLO-I	Cognitive/Analyze	Lecture, Demonstration, Group Work	Lab Report, Project
CLO3	PLO-L	Affective/Responding	Lecture, Demonstration, Group Work	Presentation, Project Work/Report

Mapping of Course – PLO – K – EP/EA																																				
Course	Program Learning Outcomes (PLOs)												Knowledge Profiles								Complex Eng. Problems (EP)					Complex Eng. Activity (EA)										
	Eng. Knowledge		Problem Analysis		Design		Investigation		Modern tool		Engineer & Society		Envir. & Sustainability		Ethics		Teamwork		Communication		Proj. Manage. & Fin.		Life-long learning													
	A	B	C	D	E	F	G	H	I	J	K	L									Natural Sciences		Mathematics		Eng. Fundamentals		Eng. Specialist		Eng. Design		Eng. Practice		Comprehension		Research Literature	
	C				C P	CA		A	A	P A	C A	A									K	K	K	K	K	K	K	K	K	K	K	K	K	K		
	K1 – K4	K5	K8	K6	K7											1	2	3	4	5	6	7	8													
	EP (P1 + two or more P2-P7)												E A																							
CSE-104 (Structured Programmig Languages Lab)	×							×			×						×	×	×							×		×	×							

Mapping Course Learning Outcomes (CLOs) with the Teaching-Learning and Assessment Strategy:

Week	Topic	Teaching Learning Strategy	Assessment Strategy	Corresponding CLOs
1.	Calculate the area of a circle, Add two/three/n integers	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO1
2.	Find out the average/maximum/minimum of two/ three/n integers	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO2
3.	Factorial of an integer, Case conversion of a character Calculate the GCD, LCM of two/n integers	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO1 CLO2
4.	Test an integer whether it is odd or even, prime or non-prime	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO1 CLO2

	Add 1+2+.... +n Add n integers using array Add/multiply two matrices	<ul style="list-style-type: none"> • Demonstration • Discussion • Group work 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO1 CLO3
6.	Reverse a string Count the length of a string	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO2 CLO3
7.	Sort n integers in ascending/descending order	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO2 CLO3
8.	Sort n students name, roll, marks using structure	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO1 CLO2
9.	A program to create a file, write char/int into it, read char/int from that file	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO1 CLO2
10.	Rewrite programs using user-defined function	<ul style="list-style-type: none"> • Demonstration • Discussion • Group work 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO2 CLO3
11.	Write recursion programs.	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO3
12.	Read and Write files	<ul style="list-style-type: none"> • Demonstration • Discussion 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO3
13.	Short project using C	<ul style="list-style-type: none"> • Demonstration • Discussion • Group work 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO3
14.	Project Work	<ul style="list-style-type: none"> • Demonstration • Discussion • Group work 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO3
15.	Project Work	<ul style="list-style-type: none"> • Demonstration • Discussion • Group work 	<ul style="list-style-type: none"> • Demonstration • Lab reports • Viva 	CLO3

References:

1. Herbert Schildt, "C: The Complete Reference", Osborne / McGraw-Hill, 4th Edition, July 1, 2017.
2. Dennis Ritchie Brian W. Kernighan, "The C Programming Language ", Prentice Hall, 1st Edition, January 1, 2015.

3. Balagurusamy, "Programming in ANSI C", McGraw-Hill Education, 2nd Edition, January 1, 2011.
4. K. N. King, "C Programming: A Modern Approach", W. W. Norton and Company, 2nd Edition, April 19, 2008