



CSE 06131223 ♦ CSE 06131224

# Structured Programming

## Lecture 1 Course Overview

Prepared by



**Md. Mijanur Rahman, Prof. Dr.**

Dept. of Computer Science and Engineering  
Jatiya Kabi Kazi Nazrul Islam University, Bangladesh

Email: [mijan@jkkniu.edu.bd](mailto:mijan@jkkniu.edu.bd)

Web: [mijanrahman.com](http://mijanrahman.com)



# Contents

## Course Overview

- **About This Course**
- **About Class and Course Evaluation**
- **The Grading System**
- **Course Prerequisites and Dependencies**
- **Course Objectives**
- **Course Contents**
- **Overall Teaching Goals**
- **Learning Outcomes**
- **Text Books and References**



# This Course

- **This is an introductory course and covers the key features of the C language and its usage.** This course is based on industrial programming experience and extensive study of the language.
- This course is aimed at advancing concepts of programming and software code organization within **the framework of structural and procedural programming paradigms.**
- The special track is organized as **a series of lectures, hands-on workshops and exercises using C programming languages** and focusing on discussing how to write a program of moderate complexity by using C language.
- This course is meant for students who have some prior programming experience and who are **familiar with procedural programming concepts.** Students without prior programming knowledge may require more time to assimilate the information provided in this course.



# Class Duration

- The duration of each semester will be 19 weeks whose breakdown is as follows:

<b>Class</b>	<b>14 weeks</b>
<b>Recess before Semester Final Examination</b>	<b>2 weeks</b>
<b>Semester Final Examination</b>	<b>3 weeks</b>
<b>Total</b>	<b>19 weeks</b>

- For theoretical course (Credit 3.0), three hours class will be conducted in a week. For lab course (Credit 1.5), three hours practical class will be conducted in a week



# Course Evaluation

- **Evaluation System for Theoretical Course** - The marking and student evaluation system will be as follows:

1. Continuous Assessments	: 40%
a. Class attendance	: 10%
b. Midterm Exam-1	: 10%
c. Midterm Exam-2	: 10%
d. Midterm Exam-3	: 10%
2. Semester Final Exam	: 60%
3. <b>Total</b>	<b>: 100%</b>



# Course Evaluation

- **Evaluation System for Lab Course** - The marking and student evaluation system will be as follows:

<b>1.</b>	<b>Continuous Assessments</b>	<b>: 40%</b>
<b>a.</b>	Class attendance	: 10%
<b>b.</b>	Lab Report	: 10%
<b>c.</b>	Continuous Evaluation	: 20%
<b>2.</b>	<b>Lab Final Exam</b>	<b>: 60%</b>
<b>a.</b>	Viva Voce	: 20%
<b>b.</b>	Lab Test	: 30%
<b>c.</b>	Ans. Script	: 10%
<b>3.</b>	<b>Total</b>	<b>: 100%</b>



# Class Attendance

- The distribution of marks for class attendance (theoretical and practical) will be as follows:

Attendance	Marks
90% and above	10
85% to 89%	09
80% to 84 %	08
75% to 79 %	07
70% to 74%	06
65% to 69%	05
60 % to 64%	04
55% to 59 %	03
50% to 54%	02
Less than 50%	00

- A student shall have to attend at least 75% of theoretical and practical classes held in a course.
- In case of shortage of attendance (not below 60%), student will be allowed to sit for examination after paying of taka 500/- as irregular fee for each course in university account.
- Below 60% will NOT be allowed to sit for examination.



# The Grading System

- Letter grades and corresponding grade points will be awarded in accordance with the provisions shown below:

Numerical Grade	Letter grade	Grade Point	Interpretation
80% and above	A+	4.00	Outstanding
75% to less than 80%	A	3.75	Excellent
70% to less than 75%	A-	3.50	Very Good
65% to less than 70%	B+	3.25	Good
60% to less than 65%	B	3.00	Satisfactory
55% to less than 60	B-	2.75	Nearly Satisfactory
50% to less than 55%	C+	2.50	Average
45% to less than 50%	C	2.25	Nearly Average
40% to less than 45%	D	2.00	Poor
Less than 40%	F	0	Fail





# Course Prerequisites and Dependencies

- This course is based on the course “**Introduction to Programming**”. So the students must have **basic knowledge of mathematical and algorithmic logics**, to understand major **control structures such as branching, loops and expressions**, to be able to use functions and to create arrays of elementary objects in their simple C programs.
- The course teaching language is **English**, so students have to have communication, reading and apprehension skills of English



# Course Objectives

- The course aims to provide exposure to **problem-solving through programming**. It aims to train the student to **the basic concepts of the C-programming language**. This course involves a lab component which is designed to give the student hands-on experience with the concepts.
- The main objective is to **provide students with good understanding of programming skills, understanding of code organization and functional hierarchical decomposition** with using complex data types.
- The course is designed to **provide complete knowledge of C language**. Students will be able to **develop logics which will help them to create programs, applications in C**. Also by learning the basic programming constructs they can easily switch over to any other language in future.



# Course Objectives

- **Major objectives are listed below:**

- To learn the fundamental programming concepts and methodologies which are essential to building good C programs.
- To practice the fundamental programming methodologies in the C programming language via laboratory experiences.
- To code, document, test, and implement a well-structured, robust computer program using the C programming language.
- To write reusable modules (collections of functions).
- To write a laboratory report.



# Course Contents

- **CSE 06131223: Structured Programming**

Credits: 3.0; Full Marks: 100; Final Exam Time: 3.0 Hours

- **Course Syllabus:**

- **Introduction to Programming Languages**, Structured Language, Problem Solving Techniques, Algorithm and Pseudo Code, Programming Style, Program Design Methodologies, Structured and Modular Program Design.
- **Introduction to C Programming**: Basic structure, C Programming environment and style, C Program Design and Execution, Simple C Programs, Variables, Data types, Types of Operators, Expressions, Statements, Input and Output.
- **Structured Programming Approach**: Control Structures: Sequential, Selection, Repetition; Types of Selection and Loop Statements; Break, Jump, Goto and Continue Statements.
- **Arrays, Pointers, and Functions**: Types of Arrays with Examples, Pointer Declaration, User-defined and Library Functions, Header files, Parameter Passing, Recursion.
- **Structures and Unions; Dynamic Memory Allocation; File Management**, Handling Files in C, Error Handling with Files.
- **Preprocessor, Header File Inclusion, Study of Standard Libraries** like `stdio.h`, `ctype.h`, `string.h`, `math.h`, `stdlib.h`, `dos.h`; Project Implementation in C.

- **CSE 06131224: Structured Programming Lab**

Credits: 2.0; Full Marks: 100; Final Exam Time: 6.0 Hours

- **Syllabus**: Different problems regarding course **CSE 06131223**.



# Overall Teaching Goals

- This is an integrated course that has the following overall goals:
  - To teach the basic computer programming skills, and their role in problem solving.
  - To teach how to program a computer using C programming language.
  - To illustrate how the output of a program is obtained.
  - To illustrate the role of computer programming in solving statistical and engineering problems.



# Learning Outcomes

- After the completion of this course, the students will be able to develop applications. It will also have the following learning outcomes:
  - Understanding programming skills.
  - Ability to describe the advantages of a high level language like C, the programming process, and the compilation process.
  - Ability to describe and use software tools in the programming process.
  - Ability to work with textual information, characters, strings, arrays.
  - Ability to demonstrate an understanding of primitive data types, values, operators and expressions in C language.
  - Ability to apply good programming principles to the design and implementation of C programs.
  - Ability to design, implement, debug and test programs using the fundamental elements of C.
  - Understanding a defensive programming concept. Ability to handle possible errors during program execution



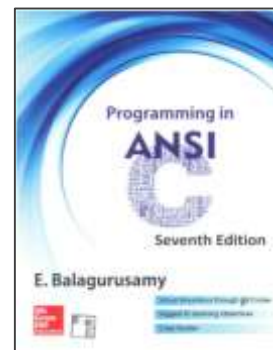
# Text Books and References

- **Online Course Materials:**
  - <https://www.tutorialspoint.com/cprogramming/index.htm>
  - <https://www.javatpoint.com/c-programming-language-tutorial>
  - <https://www.geeksforgeeks.org/c-programming-language/>
- **Online C Compiler:**
  - [https://www.onlinegdb.com/online\\_c\\_compiler](https://www.onlinegdb.com/online_c_compiler)
- **Software (Code:: Blocks):**
  - <https://www.codeblocks.org/downloads/>

## Books:



**Programming & Software Development  
With AI and Machine Learning Concepts**  
By M. M. Rahman



**Programming in ANSI C**  
By E. Balagurusamy



**Programming in C**  
By Stephen G. Kochan



**C: The Complete Reference**  
By Herbert Schildt



# Lecture 1

## Course Overview



**THE END**