Department of Computer Science and Engineering
National Institute of Technology Silchar

Lesson Plan

Cover Page: Course Overview
Semester: 2nd Year: **2014-15**

Course Title: **INTRODUCTION TO COMPUTING**	*Course Code*: **CS1101**
Total Contact Hours: 40 hrs	*Duration of class*: **1 hr**
Total Marks: **100**	*Date*: **January 2, 2015**
Lesson Plan Author: **S. Borgohain, S. Bhattacharjee, S. Chakraborty, E. Laskar, P. Das**	
Checked By:	*Date*: **January 2, 2015**

Prerequisites:

There are no identifiable or potential prerequisites to take up this course as this itself is an introductory concepts to understand programming, computing theory and associated topics presented at a surface level.

Course Overview:

This course provides students the ability to visualize a generic layout of computing and also art of programming using C. The course as such gives a balanced flavor of the basic computational aspects and programming concepts.

Course Learning Objectives-CLO

This course is the fundamental step towards understand programming paradigm and therefore basics of computer concepts are introduced. This course lays down the following objectives –

1. Discuss various components within a computer that are essential required for any program or algorithm to run on it.
2. Discuss and understand the role of the operating system and storage devices in programming.
3. Analyze and understand the representation of business logics in algorithm, pseudo-code and graphical representation (Flowcharts).
4. Write a maintainable C program for a given algorithm and implement the same.
5. Interpret the given C program. Analyze the code and bring out the expected output for the same.
6. Debug the C program and understand the various level of compilation process.

At the end of the course the students should be able to:

1. Brief out the computer concepts such as hardware, software, operating system, memory etc.
2. Design flowcharts and write the algorithms.
3. Explain Fundamentals of ‘C’ programming.
4. Demonstrate the use of basic data types.
5. Choose appropriate looping and branching constructs while building applications.
6. Design own functions based on the need.
7. Decide the usage of appropriate derived data types such as arrays, structures, pointers etc.

Course Content

INTRODUCTION TO COMPUTING
THEORY

Course Code: CS1101 **Total Marks: 100**
Hrs / Weeks: L:T:P 3:0:0 **Credits: 6**

Unit – I	
Introduction to Computer Concepts: Digital computer fundamentals – Components of a Computer System, Information Processing Cycle, Computer Hardware – Processing devices	**02 Hrs**
Problem Solving using flowcharts and pseudo-codes: Writing algorithms and drawing flowcharts for simple exercises- Swapping contents of 2 variables, Largest of given three numbers, solving a given quadratic equation, Factorial of a given integer.	**02 Hrs**
Introduction to C programming: Different programming paradigms, structure of C programs, Character set, C tokens, Keywords and Identifiers, Constants, Qualifiers, Variables, Data types, Declaration of variables.	**03 Hrs**
Unit – II	
Handling Input and Output operations: Reading a character, Writing a character, Formatted input, Formatted output, and format specifier for reading number from various number system.	**02 Hrs**
Operator and Expressions: Arithmetic operators, Relational operators, Logical Operators, Assignment operators, Increment and decrement operators, Conditional operators, Bit-wise operators, Special operators, Arithmetic expressions, evaluation of expressions, Precedence of arithmetic operators, Type conversion in expressions, Operator precedence and associativity.	**02 Hrs**
Programming Constructs: Decision Making and Branching: Decision making with if statement, Simple if statement, the if…else statement, nesting of if…else statements, The else if ladder, The switch statement, The*?:* operator, The *goto* statement. Decision making and looping: The while statement, the do statement, The for statement, Jumps in loops**.**	**05 Hrs**

Unit - III	
Arrays: One dimensional arrays, Declaration of one dimensional arrays. Initialization of one dimensional arrays, Two dimensional arrays, Initializing two dimensional arrays, Example Programs.	**04 Hrs**
Character Arrays and Strings: Declaring and Initializing String Variables, Reading Strings from Terminal, writing strings to screen, Arithmetic Operations on characters, Putting Strings together, Comparison of two strings, String handling functions.	**04 Hrs**
Unit – IV	
User-defined functions: Need for user defined functions, A multi-function program, Elements of user defined functions, Definition of functions, Return values and their types, Function calls, Function declaration, Category of functions, No arguments and no return values, Arguments but no return values, Arguments with return values, No argument but returns a value, Function that return Multiple values, Scope and Visibility.	**05 Hrs**
Structures and Unions: Introduction, Structure definition, Giving values to members, Structure initialization, Comparison of structure variables, Arrays of structure, Arrays within structures, Structure within structures, Structures and function.	**03 Hrs**
Unit – V	
Pointers: Pointer data type. Declaring and initializing a pointer. Accessing variables using pointers and one dimensional array. Pointer expressions, Pointer increments and scale factor, Pointers and arrays, Dynamic memory allocation using *malloc*, *calloc* etc.	**04 Hrs**
File Managements in C: Introduction, Definition and opening a file, Closing a file, Input/Output operations on files – getc, putc, getw, putw, fprintf, fscanf functions, Error handling during I/O operations – ferror, feof.	**04 Hrs**

Reference Books

1. E. Balaguruswamy , "Programming in ANSI C", Tata McGraw Hill, 5th edition,2010. ISBN (13): 9780070681828.
2. P. Dey, M. Ghosh, "Programming in C", oxford university press, First Edition, 2007, ISBN(13): 9780195687910.
3. Kernighan, Dennis M. Ritchie, "The C Programming Language" (2nd ed.), Prentice Hall, 2005. ISBN (13): 9780131101630.
4.

Evaluation Scheme

CIE Scheme

Assessment	Weightage in Marks
Internal	20
Mid-Sem	30
End-Sem	50
Total	**100**

Best out of 2 class tests will be considered for Internal marks.

Course Unitization for Internals and Semester End Examination

Part		Chapter	Teaching Hours	No. of Questions in			No. of Questions in End-sem
				Internal- I	Internal- II	Mid-sem	
Unit 1	1	Introducing Computer Systems	2	1	--	--	1
	2	Algorithms and Flowcharts	2	1	--	--	
	3	Introduction to C Program	3	2	--		
Unit 2	1	Managing Input and Output operations with monitor	2	1	--	--	1
	2	Operators and expressions	2	--	--	--	
	3	Decision Making and Branching	3	--	--	--	
	4	Decision making and looping	2	--	--	--	
Unit 3	1	Arrays	4	--	--	--	1
	2	Character Arrays & Strings	4	--	--	--	
Unit 4	1	User defined Functions	5	--	1		3
	2	Structures and Unions	3	--	1		
Unit 5	1	Pointers	4	--	1		2
	2	File management in C	4	--	1		

Course Coordinator:

Head of Department

Date:

Unit and Chapter wise Plan

UNIT-I

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: 1. **Introduction to Computer Concepts**	*Planned Hours: 0**2***hrs

Learning Objectives

At the end of this chapter student should be able to:

1. Fundamental understanding on various units of computing.
2. Elementary Knowledge about information processing cycle and associated aspects with focus on storage.

Lesson Schedule

Class No. Portion covered per hour

1. Exploring Computers and Their Uses, Looking Inside the Computer System – The Parts of a Computer System, The Information Processing Cycle, and Computer Hardware
2. Processing devices, Memory devices, Input and Output devices, Storage devices, Software

Model Questions

1. Explain a few of the different ways in which computers can be categorized.
2. Why are mainframe systems usually limited in the number of tasks they perform?
3. Name and differentiate between data and program
5. Identify four categories of computer hardware

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *2.* **Algorithms and Flowcharts**	*Planned Hours: **02*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Understand the representation of collective tasks of a given problem in terms of algorithms and flowcharts.
2. Design and write algorithms for a given problem

Lesson Schedule

Class No. Portion covered per hour

1. Algorithms and Flowcharts: Writing algorithms and drawing flowcharts for simple exercises- Swapping contents of 2 variables,
2. Largest of given three numbers, solving a given quadratic equation, Factorial of a given integer.

Model Questions

1. Write the various steps involved in problem solving
2. List some important reasons for using Flowchart.
3. Draw the flowchart for finding the largest among 3 numbers

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *3.* **Constants, variables and data types**	*Planned Hours: **03*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Clear of how to declare variables and data types
2. Clear focus on character set, C tokens

Lesson Schedule

Class No. Portion covered per hour

1. Introduction, Character set, C tokens, keywords & Identifiers, constants
2. Variables, data types, declaration of variables, Example Programs

Model Questions

1. Explain character set & c tokens
2. With an example differentiate between number constants and character constants
3. Explain different data types available in C
4. List the rules for declaring a variable? Explain with sample programs

UNIT-II

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *4.* ***Managing*** **Input and Output operations**	*Planned Hours:* ***02*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Understanding the roles of IO operation through standard devices in programmatic perspective.
2. Develop executable statements in C for reading & writing characters

Lesson Schedule

Class No. Portion covered per hour

1. Reading a character, Writing a character, Formatted input, Formatted output

Model Questions

1. With an example program demonstrate the usage of *getchar( )* & *putchar( )*
2. List all the options available for outputting integer constant with example

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: 5. **Operators and expressions**	*Planned Hours:* ***02*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Discuss different operators and categorize the operators in C programming
2. Discuss the precedence and associatively of operators.
3. Apply Implicit and explicit type conversions

Lesson Schedule

Class No. Portion covered per hour

1. Introduction, Arithmetic Operators, Relational operators, Logical Operators Assignment, Increment & Decrement Operators
2. Conditional, Bitwise, Special Operators, Arithmetic Expressions, Evaluation of Expressions
3. Precedence of Arithmetic Operators, Type Conversion in Expressions, Operator Precedence & Associativity, Simple Programs

Model Questions

1. Briefly demonstrate arithmetic & relational operators with example
2. Distinguish between logical and bitwise operators
3. Differentiate between = and == operators
4. Write a program to enter two numbers and find the smallest out of them. Use conditional operator
5. Evaluate the following expressions, show the precedence of operators
   a) ***(3*2) + 2 % 3+5 / 6***
   b) ***16>>2***

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *6.* **Decision Making and Branching**	*Planned Hours: **03*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Understand the statements and work flow of the decision making statements like If, If-Else.
2. Understand the role of multi flow decision statement like switch.
3. Discuss unconditional decision making statements like go-to.
4. Demonstrate all the decision making statements in C in form of an case study.

Lesson Schedule

Class No. Portion covered per hour

1. Introduction, Decision Making with if statement, Simple if statement, if..else , nesting of if..else Statements
2. The else if ladder, Switch Statement, Ternary Operator ?, Simple programs
3. The goto Statement, Simple Programming exercises

Model Questions

1. Why are control statements required? Explain nesting of if..else statements with syntax & flowchart..
2. When do we use switch Statement and what are its limitations? Demonstrate switch statement with an example program.
3. goto statements are not recommended to be used in programs, justify.

4. Using if statement write a program to check whether the voter is eligible for voting or not. If his/her age is equal or greater than 18, display message "eligible" otherwise "Not Eligible".

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *7.* **Decision making and looping**	*Planned Hours:* ***02*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Understand different looping constructs in C.
2. Demonstrate the looping constructs using a program.
3. Apply the decision making and Looping constructs.

Lesson Schedule

Class No. Portion covered per hour

1. Introduction, The While Statement, Do While Statement, simple programs
2. For Statement, Jumps & Loops, Simple programming exercises

Model Questions

1. Explain the 2 types of controlled loops with flowcharts.
2. Write the steps involved in looping process.
3. Explain the difference b/w break & continue statement with examples.
4. What is a loop? Why it is necessary in the program?
5. Write a program to display the number series 1, 3, 9, 27, 81…90 by using for loop

UNIT-III

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *8.* **Arrays**	*Planned Hours:* ***04*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Describe N-Dimensional Array with usage, declaration and initialization
2. Develop program using arrays

Lesson Schedule

Class No. Portion covered per hour

1. Introduction, One Dimensional arrays, Declarations of one dimensional arrays
2. Initialization of one Dimensional arrays, Two Dimensional arrays, initializing two Dimensional arrays.

Model Questions

1. What are 1-D arrays & 2-D arrays?
2. Explain different ways of initializing vales for a 2-D array.
3. Write a program to read 2 matrices A(MxN) and B(PxQ) and compute the product of A and B after checking compatibility for multiplication. Output all the matrices.

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *9.* **Characters Arrays & Strings**	*Planned Hours: **04*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Describe declaration and initialization of string variables
2. Arithmetic operations on characters
3. String handling functions

Lesson Schedule

Class No. Portion covered per hour

1. Declaring and Initialization of String variables
2. Arithmetic operations on strings
3. String Handling functions

Model Questions

1. Describe the functions performed on strings
2. Explain string handling functions
3. Write a program which will read a string and rewrite it in alphabetical order

UNIT-IV

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *10.* **User defined Functions**	*Planned Hours:* ***05*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Understand the need for functions and its role in making the program modular.
2. Demonstrate the use of the functions with trivial programs.

Lesson Schedule

Class No. Portion covered per hour

1. Introduction, Requirement of user-defined functions, multi-function program, Elements of User Defined Function
2. Definition of Functions, Return Values & their Types, Function Calls, Function Declaration
3. Category of Functions, No arguments and No return values, No arguments but returns a value
4. Arguments but No return values, Arguments with Return values, function that Return multiple values

Model Questions

1. What is the need for user defined functions? Explain
2. What are the 3 elements of functions? Explain definition of functions
3. Explain the different ways of passing parameter to the functions with examples
4. Differentiate between actual & formal arguments
5. Write a user-defined function for performing the following tasks
   a) Find the square of a given number
   b) Find the Area of a square
   c) Convert a decimal number to its binary equivalent
   d) Reverse the number

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *11.* **Structures and Unions**	*Planned Hours:* ***03*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1) Discuss use of structures in programming environment and focus on – declaration, and initialization.
2) Demonstrate the use of structures and union with examples.

Lesson Schedule

Class No. Portion covered per hour

1. Introduction, Introduction, Structure Definition, Giving Values to Members
2. Structure Initialization, Comparison of Structure Variables
3. Arrays of Structures, Arrays within Structures
4. Structure within Structures, Structures & Functions

Model Questions

1 What is a structure? Explain the definition of structure with proper syntax & example.

2 Explain structure within structure with suitable example.

3 Give the difference b/w arrays & structures.

4 Create a structure called student with the following members- student name, roll-no, marks in 3 tests. Write a C program to create N records & sort them using bubble sort display the records.

UNIT-V

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *12.* **Pointers**	*Planned Hours:* ***04*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1. Discuss briefly the concept of pointers – declaration, usage, initialization, different operations.
2. Decide on when a pointer has to be used.
3. Demonstrate the merits of pointers using trivial exercises

Lesson Schedule

Class No. *Portion covered per hour*

1. Introduction, Pointer Data Type, Declaring & Initializing the Pointer
2. Accessing Variables using Pointers & One Dimensional Arrays
3. Pointer Expression, Pointer Increments & Scale Factor
4. Pointers & Arrays

Model Questions

1) What is a pointer?
2) Write the benefits of pointers to the programmers.
3) Explain the concept of accessing 1- array elements using pointers.
4) Briefly explain pointer increments & scale factors.

Course Code and Title: **(CS-1101) INTRODUCTION TO COMPUTING**	
Chapter Number and Title: *13.* **File management in C**	*Planned Hours: **04*** **hrs**

Learning Objectives

At the end of this chapter student should be able to:

1 Understand I/O operation using File and focus on topics like usage, Handling file pointers.

2 Demonstrate the use of Files with various open modes and hence arrive at comparative study between them.

Lesson Schedule

Class No. *Portion covered per hour*

1. Introduction, Definition and Opening a File
2. Closing a File, I/O Operations on Files
3. Error Handling during I/O Operations

Model Questions

1. How to create & Open a file.
2. What are the functions used in files.
3. How are the input & output operations performed on a file.

National Institute of Technology, Silchar
Mid-Semester (UG) Examination, December '2015
Subject Code: CS-1101, Subject: Introduction to Computing
Semester: 2nd. Branch: All branches
Duration: One Hour. Total Marks: 30
Figure in the right hand margin indicates full marks for the question.

1. (a) What is a bit? What is a byte? **1+1**

(b) Draw the block diagram of a digital computer. **2**

(c) Draw the flowchart for finding the smallest among 03 nos. **2**

2. (a) Name and describe the four basic datatypes in C **2**

(b) The price of one kg of sugar is Rs 36.75 and one kg of rice is Rs 40. Write a complete C program without using function to get these values from the user and display the prices as follows. Do not use an array. **2**

LIST OF ITEMS

Item	**Price**
Sugar	Rs 36.75
Rice	Rs 40

(c) Identify the error in the following snippet of code: **1**

```
int a[n], i;
for(i=0;i<5;i++){
      scanf("%d", a[i]);
}
```

3. Arrange the following C operators in order of precedence **1**

(a) **i)** & **ii)** && **iii)** || **iv)** ! **v)** ==

(b) Write a C statement to check whether a given input value is even or odd using a **ternary** operator. **2**

(c) Let ***i*** is a variable of type integer with a value of −5, and ***j*** is an integer type of variable with a value of 3. Which of the following relational expressions evaluate to true? **3**

Expression	**Answer**
(i < j)	
((i > j) || !(i > j))	
(i + 8 < j)	

4. Write ***for*** statements that print the following sequences of values: **1+1**

(a) **i.** 1, 2, 3, 4, 5, 6, 7
ii. 3, 8, 13, 18, 23

(b) Rewrite the following statements (R.H.S) so that a ***do-while*** loop is used instead of a ***while*** loop. The results from executing the statements must remain the same. **2**

```
int j = 0, sum = 0;
while (j < 10) {
      sum += j;
      j ++;
}
```

5. (a) Define an array. **1**

(b) Write a C program to perform the transpose of a matrix. **2**

(c) Given the following declaration, what is stored in the 8th element of the array? **1**
int num[10]={0, 1, 2, 3, 4};

6. (a) State two advantages to the use of functions in a program. **2**

(b) Write a complete C program to find the factorial of a given number ***n*** using recursive function. **3**