BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI
PILANI CAMPUS
Instruction Division

Second Semester 2017-2018

Course Handout (Part II)

January08, 2017

In addition to part I (General Handout for all courses appended to the time table) this portion gives further specific details regarding the course.

Course No.	: CS F111
Course Title	: COMPUTER PROGRAMMING
Instructor-in-charge	: Dr. Manoj Kannan
Lecture Instructors	: Dr. Sundaresan Raman(sundaresan.raman@pilani.bits-pilani.ac.in)
	Dr. Manoj Kannan(manojkannan@pilani.bits-pilani.ac.in)
Lab Instructors	: Dr. Sundaresan Raman (sundaresan.raman@pilani.bits-pilani.ac.in)
	Dr. Lavika Goel(lavika.goel@pilani.bits-pilani.ac.in)
	Dr. Mayuri Digalwar(mayuri@pilani.bits-pilani.ac.in)
	Ms. Aryav Yadav (p20170026@pilani.bits-pilani.ac.in)
	Ms. Prerna Kaushik (p2013192@pilani.bits-pilani.ac.in)
	Mr. Harry Johnson (h20170120@pilani.bits-pilani.ac.in)
	Mr. G. Sasidhar (h20170122@pilani.bits-pilani.ac.in)

1. Course Description:

The primary goals of the course are to introduce:

- Basic representation of data and how to process data using the representation inside a computer.
- Techniques for specifying data, operations on data, and problem solving using C programming language.
- Systematic techniques and approaches for constructing programs.

2. Scope of the Course:

The course covers the following topics: Basic Model of a Computer; Problem Solving – Basic Computing Steps and Flow Charting (Assignment, Sequencing, Conditionals, Iteration). Programming Constructs – Expressions, Statements, Conditionals, Iterators/Loops, Functions/Procedures; Data Types – Primitive Types, Tuples, Choices (Unions or Enumerations), Lists/Arrays, Pointers and Dynamically Allocated Data. Input output and Files.

3. Learning Outcomes:

- Given the requirements for any computational problem, or given an algorithm, the students will be able to write equivalent program in high-level language such as C.
- Given a problem that's iterative in nature, students would be able to use the loop constructs appropriately, using nested loops if required.
- Given the memory requirements, students will be able to design the appropriate data structures (static, dynamic memory allocation) for the given problem.
- Given a complex problem, students will be able to logically break down into simpler functions and implement them in C.
- Given any type of data, students will be able to predict their internal representation and thus reason particular outputs for a given input.
- Given the requirements, students will be able to write programs to create files, update files and read data from files.

4. Textbook:

T1: Hanly, J.R. and E.B. Koffman. *Problem Solving and Program Design in C(7/e)*. Pearson Education, 2013.

5. Reference Books:

R1: Patt, Yale. Introduction to Computing Systems: From bits & gates to C &beyond (2/e). McGraw Hill Education, 2017.

The authors take a bottom-up approach to introduce computers and computing.

R2: Forouzan, B.A. and Richard F. Gilberg . Computer science A structured programming approach using C (3/e). Cengage Learning, 2007.

The book gives a fairly comprehensive overview of C, with several example programs.

R3: Gottfried, B.S. and Jitender Chhabra. Programming with C (Schaum's Outlines Series, 3/e). McGraw Hill Education, 2017.

Another beginner's book on C programming, with lots ofdrill exercisesand programs.

R4:Kernighan, B.W and Dennis Ritchie. The C Programming Language (2/e). Pearson Education India, 2015.

Considered the ultimate treatise on C, it conveys the philosophy and practice of C very tersely, but is pitched at an advanced beginnger level.

R5:Das, S. Unix: Concepts and Applications (4/e). McGraw Hill Education, 2017.

Provides a great introduction to using Unix commands.

6. Lecture Plan:

Lec. #	Learning Objectives	Topics to be covered	Learning Outcomes	Reference to Text
01	Introduction to the course	Introduction to Programming; need for programming; overview of computers and computing; learningoutcomes	Students can write simple C programs, compile and execute them in a Unix environment	T1: 1.1-1.3
02-03	Introduction to working with Unix (Linux)	Compilation and execution of a program, other useful Unix commands		R5
04-06	Representation of numbers inside the computer	Internal representation of data; IEEE floating point representation of numbers		R1
07	Solving problems using flowcharts	How to express a problem using flowcharts, using prime number problem as an example		T1: 2.4; Class notes
08-09	Overview of the C programming language	A programming example – prime numbers		Notes
10-11	Simple data storage and manipulation	Data Types; variable names; sizes, constants and declarations	Students can evaluate a complex arithmetic expression; also specify the exact internal data representation.	T1: 2.1-2.2
Lec. #	**Learning Objectives**	**Topics to be covered**	**Learning Outcomes**	**Reference to Text**

12-14	Flow of control	Statements – if… else, if… else… if, switch Loops – while; do… while; for; break and continue	Given a problem that's iterative or conditional in nature, students would be able to use the loop constructs / if-else construct appropriately.	T1: 4.1-4.3, 4.7-4.8, 5.1-5.2, 5.4-5.8
15-19	Modularity and program structure	Functions and program Structure; return types; scope rules; header files Recursion The C Preprocessor	Given a complex problem statement, students will be able to logically break down into simpler modules involving pointers and arrays, and write a modular program using functions.	T1: 3.1, 3.4-3.5, 6.1-6.4, 10.1-10.4
20-24	Pointers and Arrays	Pointers and function arguments; call by value; call by reference; pointer arithmetic; arrays of pointers; string manipulation		T1: 6.1, 8.1-8.5, 9.1-9.5
25-28	Multi-dimensional arrays	Multidimensional arrays; pointers vs. multidimensional Arrays; Command Line Arguments		T1: 8.7-8.8, 13.7
29-33	User-defined data types	Structures and unions; enumerated data types; type definitions	Students will be able to create user-defined data types pertaining to a given problem, create and manipulate data structures using dynamic memory management, and handle text files.	T1: 14.3-14.4
34	Input and Output	Standard input and output functions; formatted input and output; File handling		T1: 2.3
Lec. #	**Learning Objectives**	**Topics to be covered**	**Learning Outcomes**	**Reference to Text**

35-40	Dynamic memory and simple data structures in C	Dynamic memory management; linked lists		T1: 12.1-12.2 Notes

7. Evaluation Scheme:

#	Evaluation Component	Marks	Date and Time	Remarks
1	Mid-semester Test	35 (17.5%)	9/3 4:00- 5:30 PM	Closed-book
2	Comprehensive Examination	75 (37.5%)	11/5 AN	One section will be open-book
3	Online Programming Test	25 (12.5%)	22-Apr-2018	Open-book; 1½ hours
4	Quizzes/Assignments	35 (17.5%)	To be announced	Some may be given online
5	Practical session evaluation	30 (15%)	During the weekly practical session	Each practical session will be evaluated out of 3 marks

8. Practical Sessions:

Supervised practical sessions will be held every week wherein the students will practice writing, executing and debugging C programs in a Unix environment. Students are expected to work in their respective Linux accounts created on the *Prithvi* server (172.24.16.31). Every student is expected to maintain a laboratory record notebook in which the programs are documented and brought to the lab every week. Each lab session will be evaluated for 3 marks – 1 mark for attendance and punctuality, 1 mark for active participation, and 1 mark for lab record completion.

9. Attendance Policy:

Attendance shall be recorded in lectures and practical classes. Poor attendance is likely to:

- Have your user privileges on Prithvi server suspended;
- Have you lose an opportunity to take part in assignments and quizzes;
- Make you miss out on great classroom experience.

10. Makeup Policy:

For a foreseen absence, make-up request should be made personally by meeting the Instructor-in-Charge. Reasons for unanticipated absence that qualify one for make-up include medical or similar personal emergencies only. In such cases, the Instructor-in-Charge must be informed either by email or by telephone. Usually, make-ups for regular laboratory sessions and assessments held therein are not awarded. The decision by the Instructor-in-Charge regarding granting makeups shall be final.

11. Grading Policy:

Award of grades would be guided in general by the histogram of marks. Decision for borderline cases would be based on the student's attendance in lectures and laboratory sessions, and instructors' overall assessment of the individual's sincerity andendeavour. If a student does not give sufficient opportunity for being assessed, either by missing a component entirely or by not applying oneself to the task seriously, he/she may be awarded 'NC' report.

12. Chamber Consultation Hour:

Dr. Manoj Kannan – Fridays 4:30 to 5:30 PM in Rm. 3270; Phone: 01596–515855

Dr. Sundaresan Raman – (To be announced) inRm. 6121-O; Phone: 01596–515684

To contact the practical class instructors for consultation, you may send them an email.

13. Announcements and Notices:

All announcements will be done in the classroom, and may be followed up with an email. For sharing course material, either *Nalanda* (nalanda.bits-pilani.ac.in) or Google Drive may be used. Important notices, such as seating arrangement for tests and exams, will also be displayed on notice boards.

Instructor-in-Charge
CS F111