UNIVERSITY OF CALICUT

THENHIPALAM, CALICUT UNIVERSITY P.O

DEGREE OF
BACHELOR OF SCIENCE (B.Sc.)
IN
COMPUTERSCIENCE

(CHOICE BASED CREDIT AND SEMESTER SYSTEM)

UNDER THE

FACULTY OF SCIENCE

SYLLABUS

(FOR THE STUDENTS ADMITTED FROM THE ACADEMIC YEAR 2017 – 18 ONWARDS)

BOARD OF STUDIES IN COMPUTER SCIENCE (UG)

THENHIPALAM, CALICUT UNIVERSITY P.O
KERALA, 673 635, INDIA
JUNE, 2017

REGULATIONS

FOR THE DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE

(CHOICE BASED CREDIT AND SEMESTER SYSTEM)

EFFECTIVE FROM THE ACADEMIC YEAR 2017-18

B.SC COMPUTER SCIENCE PROGRAMME OBJECTIVE

The basic objective of the Programme is to open a channel of admission for computing courses for students, who have done the 10+2 and are interested in taking computing/IT as a career. After acquiring the Bachelor's Degree (B.Sc. Computer Science) at University of Calicut, there is further educational opportunity to go for an MCA or other Master's Programme like MSc (Computer Science), MSc (IT), MBA, etc., at this university or at any other University/Institute. Also after completing the B.Sc. Computer Science Programme, a student should be able to get entry level job in the field of Information Technology or ITES or they can take up self-employment in Indian & global software market. The specific objectives of the Programme include

1. To attract young minds to the potentially rich & employable field of computer applications
2. To be a foundation graduate Programme this will act as a feeder course for higher studies in the area of Computer Science/Applications
3. To develop skills in software development so as to enable the B.Sc. Computer Science graduates to take up self-employment in Indian & global software market.
4. To train & equip the students to meet the requirements of the Software industry in the country and outside.

PROGRAMME STRUCTURE

Duration: The duration of the B.Sc. Computer Science Programme shall be 6 semesters distributed over a period of 3 academic years. The odd semesters (1, 3, 5) shall be from June to October and the even Semesters (2, 4, 6) shall be from November to March. Each semester shall have 90 working days inclusive of all examinations.

Courses: The B.Sc. Computer Science Programme includes four types of courses, viz., Common Courses (Code A), Core courses (Code B), Complementary courses (Code C) and Open course

(Code D). The minimum number of courses required for completion of the B.Sc. Computer Science Programme is 36.

Credits: Each course shall have certain credits. For passing the B.Sc. Computer Science Programme the student shall be required to achieve a minimum of 120 credits of which 36 (14 for common English courses + 8 for common languages other than English + 16 credits for General courses) credit shall be from common courses, a minimum of 2 credits for project and 2 credits for the open course. Minimum credits required for core, complementary and open courses put together are 82.

Attendance: A student shall be permitted to appear for the semester examination, only if he/she secures not less than 75% attendance in each semester. Attendance shall be maintained by the concerned Department. Condonation of shortage of attendance to a maximum of 9 days in a semester subject to a maximum of two times during the whole period of the B.Sc. Computer Science Programme may be granted by the University. Benefits of attendance may be granted to students who attend the approved activities of college/university with prior concurrence of the Head of the institution. Participation In such activities may be treated as presence in lieu of their absence on production of participation/attendance certificate in curricular/ extracurricular activities. It should be limited 9 days in a semester. The condonation of shortage of attendance shall be granted according to the existing prescribed norms.

If a student registered in first semester of the B.Sc. Computer Science Programme is continuously absent from the classes for more than 14 working days at the beginning of the semester without informing the authorities the matter shall immediately be brought to the notice of the Registrar of the university. The names of such students shall be removed from the rolls.

Admission to repeat courses should be within the sanctioned strength. However if more candidates are there, the candidates who have suffered serious health problems, on production of a medical certificate issued by a physician not below the rank of a Civil Surgeon in Government service, may be permitted to repeat the course, with a written order issued by the Registrar, Calicut University (by considering his/her SGPA/CGPA and percentage of attendance). The number of such candidates should not exceed two.

Grace Marks: Grace Marks may be awarded to a student for meritorious achievements in co-curricular activities (in Sports/Arts/ NSS/NCC/ StudentEntrepreneurship) carried out besides the regular class hours. Such a benefit is applicable and limited to a maximum of 8 courses in an academic year spreading over two semesters. No credit shall be assigned for such activities.

Project: Every student of the B.Sc. Computer Science Programme shall have to work on a project of not less than 2 credits under the supervision of a faculty member as per the curriculum.

Extension Activities: Compulsory social service (CSS) for a period of 15 days is essential for the successful completion of the B.Sc. Computer Science Programme.

ADMISSION

The admission to all programmes will be as per the rules and regulations of the University. The eligibility criteria for admission shall be as announced by the University from time to time.

Separate rank lists shall be drawn up for reserved seats as per the existing rules.

The admitted candidates shall subsequently undergo the prescribed courses of study in a college affiliated to the university for six semesters within a period of not less than three years; clear all the examinations prescribed and fulfill all such conditions as prescribed by the university from time to time.

The College shall make available to all students admitted a Prospectus listing all the courses offered in various Departments during a particular semester. The information so provided shall contain title of the courses, the semester in which it is offered and credits for the courses. Detailed syllabi shall be made available in the University/college websites.

There shall be a uniform calendar prepared by the University for the Registration, conduct /schedule of the courses, examinations and publication of results. The University shall ensure that the calendar is strictly followed.

There shall be provision for inter collegiate and inter university transfer in third and fifth semester within a period of two weeks from the date of commencement of the semester. For the interuniversity or intra-university transfer of a student, he/she has a minimum of 20 credits in the credit bank a) in the same discipline and b) within Kerala.

Complementary changes at the time of college transfer are permitted in the third semester if all conditions are fulfilled. Complementary changes will not be permitted in the fifth semester.

REGISTRATION

Each student shall register for the courses he/she proposes to take through 'on line', in consultation with the Faculty Adviser within two weeks from the commencement of each semester. The college shall send a list of students registered for each Programme in each Semester giving the details of courses registered, including repeat courses, to the university in the prescribed form within 45 days from the commencement of the semester.

A student shall be permitted to register for the examination also. If registration for examination is not possible owing to shortage of attendance beyond condonation limit, the student shall be permitted to move to the next semester. In such cases, a request from the student may be forwarded through the principal of the college to the University within two weeks of the commencement of that semester. An undertaking from the Principal may also be obtained stating that the students will be permitted to make up the shortage of attendance in that semester after completing 6 semesters.(Students shall make up the shortage of attendance in 'Repeat Semester' after completion of the Programme).

The 'Repeat Semester' shall be possible only once for the entire Programme and shall be done in the same college.

A student who registered for the course shall successfully complete the Programme within 6 years from the year of first registration. If not, such candidate has to cancel the existing registration and join afresh as a new candidate.

The students who have attendance within the limit prescribed, but could not register for the examination have to apply for the token registration, within two weeks of the commencement of the next semester.

COURSE EVALUATION

Total marks for each core, elective course, including lab coursesshall be 100 marks and Project Evaluation cum Programme viva voce shall be 150 marks and open course shall be 50 marks.

The evaluation scheme for each course shall contain two parts (1) Internal evaluation (2) external evaluation 20% weight shall be given to the internal evaluation. The remaining 80% weight shall be for the external evaluation.

INTERNAL EVALUATION

The internal assessment shall be based on a predetermined transparent system involving written test, assignments, seminars and attendance in respect of theory courses and on test/record/viva/attendance in respect of lab courses.

20% of the total marks in each course (i.e., 20 marks), including lab are for internal examinations and for and project evaluation cum programme viva voce, it shall be 30 marks.

Components with percentage of marks of Internal Evaluation of Theory Courses are

Test paper (50%)	–	10 Marks
Attendance (25%)	–	5 Marks
Assignment/Seminar/Viva (25%)	–	5 Marks

Components with percentage of marks of Internal Evaluation of Lab Courses are

Test paper (50%)	–	10 Marks
Attendance (25%)	–	5 Marks
Assignment/Lab involvement (25%)	–	5 Marks

Attendance of each course will be evaluated as below

Above 90% attendance	–	5 Marks
85 to 89%	–	4 Marks
80 to 84%	–	3 Marks
76 to 79 %	–	2 Marks
75%	–	1 Marks

Internal evaluation for the project shall be generally based on content, method of presentation, final conclusion, and orientation to research aptitude. The split up shall be

Punctuality	–	6 Marks
Use of Data	–	6 Marks
Scheme/Organization of Report	–	8 Marks
Viva-voce	–	10 Marks

(If a fraction appears in internal marks, nearest whole number is to be taken)

To ensure transparency of the evaluation process, the internal assessment marks awarded to the students in each course in a semester shall be notified on the notice board at least one week before the commencement of external examination. There shall not be any chance for improvement for internal marks. The course teacher(s) shall maintain the academic record of each student registered for the course, which shall be forwarded to the University by the college Principal after obtaining the signature of both course teacher and HOD.

The marks secured for internal examination only need be sent to university, by the colleges concerned.

EXTERNAL EVALUATION

There shall be University examinations for each course at the end of each semester.

Practical examinations shall be conducted by the University at the end of second, fourth and sixth semesters.

External project evaluation cum Programme viva-voce shall be conducted along with the project evaluation at the end of the sixth semester.

External evaluation carries 80% of marks, i.e., 80 Marks, for each course.

External evaluation of even (2, 4 and 6) semesters will be conducted in centralized valuation camps immediately after the examination. Answer scripts of odd semester (1, 3 and 5) examination will be evaluated by home valuation. Allquestion papers shall be set by the university.

The model of the question paper for external examination (theory courses) of 3 Hrs duration shall be

1. **Section A:** 10 compulsory short answer type questions of 1 Marks each (**Total 10 Marks**)
2. **Section B:** 5 compulsory paragraph type questions of 3 Marks each (either a single question or can have subdivisions) (**Total 15 Marks**)
3. **Section C:** 5 short essay type questions of 5 Marks each, to be attempted from a set of 8 questions – at least one question from each unit (either a single question or can have subdivisions) (**Total 25 Marks**)
4. **Section D:**3 long essay type questions of 10 Marks each, to be attempted from a set of 5 questions – at least one question from each unit (either a single question or can have subdivisions) (**Total 30 Marks**)

The external examination in theory courses is to be conducted with question papers set by external experts. The evaluation of the answer scripts shall be done by examiners based on a well-defined scheme of valuation and answer keys shall be provided by the University.

The external examination in practical courses shall be conducted by two examiners, one internal and an external, appointed by the University.

The project evaluation with Programme viva voce will be conducted by two examiners, one internal and an external (appointed by the University), at the end of the sixth semester.

No practical examination will be conducted in odd semester. Practical examinations for B.Sc. Computer Science Programme shall be conducted in the even semester 2, 4 and 6.

The model of the question paper for external examination (lab courses) of 3 Hrs. duration shall be

1. **Section A**: One marked question of 30 Marks from Programming Lab Part A is to be attempted (Design Algorithm/Flowchart/Interface 10 Marks, Code 10 Marks and Result 10 Marks. **Total 30 Marks**)

2. **Section B:**One marked question of 30 Marks from Programming Lab Part B is to be attempted (Design Algorithm/Flowchart/Interface 10 Marks, Code 10 Marks and Result 10 Marks. **Total 30 Marks**)

3. **Section C:**Lab viva voce (Total 10 Marks)

4. **Section D:**Lab Record (Total 10 Marks)

Project guidelines – Students must be encouraged to do projects in the latest tools or tools appropriate for their topic. Department should conduct monthly evaluation of the project and give necessary instructions to the students as and when required. Number of students in a project group must be limited to 4.

The scheme of evaluation for project cum Programme viva voce shall be

1. Relevance of the Topic, Statement of Objectives, Methodology (Reference/Bibliography) (**Total 30 Marks**)

2. Presentation, Quality of Analysis/Use of Statistical tools, Findings and recommendations (**Total 40 Marks**)

3. Project cum Programme Viva Voce (**Total 50 Marks)**

REVALUATION

In the new system of grading, revaluation is permissible. The prevailing rules for revaluation are applicable.

Students can apply for photocopies of answer scripts of external examinations. Applications for photocopies/scrutiny/revaluation should be submitted within 10 days of publication of results. The fee for this shall be as decided by the university.

COURSE IMPROVEMENT

A maximum of two courses (Common, Core, Complementary or Open) can be improved in each semester. Improvement of a particular semester can be done only once. The student shall avail the improvement chance in the succeeding year after the successful completion of the semester concerned. The internal marks already obtained will be carried forward to determine the grades/marks in the improvement examination. If the candidate fails to appear for the improvement examination after registration, or if there is no change in the results of the improvement examination appeared, the marks/grades obtained in the first appearance will be retained.

Improvement and supplementary examinations cannot be done simultaneously.

EVALUATION AND GRADING

Mark system is followed instead of direct grading for each question (for both internal and external examinations). For each course in the semester letter grade, grade point and % of marks are introduced in 7 point indirect grading system. The grading on the basis of a total internal and external mark will be indicated for each course and for each semester and for the entire Programme.

Indirect Grading System in 7 point scale is as below.

% of Marks	Grade	Interpretation	Grade Point (G)	Range of Grade Points	Class
90 and above	A+	Outstanding	6	5.5 to 6	First class with distinction
80 to below 90	A	Excellent	5	4.5 to 5.49	

70 to below 80	B	Very Good	4	3.5 to 4.49	First class
60 to below 70	C	Good	3	2.5 to 3.49	
50 to below 60	D	Satisfactory	2	1.5 to 2.49	Second class
40 to below 50	E	Pass/Adequate	1	0.5 to 1.49	Pass
Below 40	F	Failure	0	0 to 0.49	Fail

An aggregate of E grade with 40% marks (after external and internal put together) is required in each course for a pass and also for awarding a degree.

Appearance for Internal Assessment (IA) and End Semester Evaluation (ESEexternal)) are compulsory and no grade shall be awarded to a candidate if she/he is absent for IA/ESE or both. For a pass in each course 40% marks or E grade is necessary.

A student who fails to secure a minimum grade for a pass in a course is permitted to write the examination along with the next batch.

After the successful completion of a semester, Semester Grade Point Average (SGPA) of a student in that semester is calculated using the formula given below. For the successful completion of a semester, a student should pass all courses. However, a student is permitted to move to the next semester irrespective of SGPA obtained.

The Semester Grade Point Average can be calculated as

$$\text{SGPA} = \frac{\text{Sum of the Credit Points of all courses in a semester}}{\text{Total Credits in that Semester}}$$

i.e.

$$\text{SGPA} = \frac{\text{C1 x G1} + \text{C2 x G2} + \text{C3 x G3} \ldots.}{\text{n}}$$

where *G1*, *G2*, … are grade points of different courses; C1, C2, ... are credits of different courses of the same semester and *n* is the total credits in that semester.

The Cumulative Grade Point Average (CGPA) of the student is calculated at the end of a Programme. The CGPA of a student determines the overall academic level of the student in a

Programme and is the criterion for ranking the students. CGPA can be calculated by the following formula

The Cumulative Grade Point Average (CGPA) can be calculated as

$$\text{CGPA} = \frac{\text{TOTAL CREDIT PONTS OBTAINED IN SIX SEMESTERS}}{\text{TOTAL CREDITS}}$$

GRADE CARD

The University shall issue to the students grade/marks card (by online) on completion of each semester, which shall contain the following information.

a) Name of University
b) Name of college
c) Title of Under Graduate Programme
d) Semester concerned
e) Name and Register Number of student
f) Code number, Title and Credits of each course opted in the semester
g) Internal marks, External marks, total marks, Grade point (G) and letter grade for each course in the semester
h) The total credits, total credit points and SGPA in the semester (corrected to two decimal places)
i) Percentage of total marks

The final Grade/mark Card issued at the end of the final semester shall contain the details of all courses taken during the entire Programme including those taken over and above the prescribed minimum credits for obtaining the degree. However, for the compilation of CGPA only the best performed courses, if any, with maximum grade points alone shall be taken subject to the minimum credits requirements (120) for passing a specific degree. The final grade card shall show the percentage of marks, CGPA (corrected to two decimal places) and the overall letter grade of a student for the entire Programme. The final grade/mark card shall also include the grade points and letter grade of common courses, core courses, complementary courses and open courses, separately. This is to be done in a seven point indirect scale.

AWARD OF DEGREE

The successful completion of all the courses (common, core, complementary and open courses) prescribed for the B.Sc. Computer Science Programme with E grade (40 %) shall be the minimum requirement for the award of B.Sc. Computer Science Programme degree.

GRIEVANCE REDRESSAL COMMITTEE

COLLEGELEVEL

The College shall form a Grievance Redressal Committee in each department comprising of course teacher and one senior teacher as members and the Head of the department as Chairman. This committee shall address all grievances relating to the internal assessment grades of the students. There shall be a CollegeLevel Grievance Redressal Committee comprising of Student Advisor, two senior teachers and two staff council members (one shall be elected member) as members and principal as Chairman.

UNIVERSITYLEVEL

The University shall form a Grievance Redressal Committee as per the existing norms.

BACHELOR OF COMPUTER SCIENCEPROGRAMME STRUCTURE

LEGEND	
Item	**Description**
C	Credits
E	External Component
I	Internal Component
L	Lecture Hours
P	Practical Hours
T	Total

Semester I									
No	**Course Code**	**Course Name**	**C**	**Marks**			**Hrs/wk**		
				I	**E**	**T**	**L**	**P**	**T**
01	XXXXA01	Communication Skills in English	4	20	80	100	5	0	5
02	XXXXA02	Critical Reasoning, Writing and Presentation	3	20	80	100	4	0	4
03	XXXXA07	Communication Skills in Languages other than English	4	20	80	100	5	0	5
04	BCS1B01	Computer Fundamentals & HTML	3	20	80	100	1	2	3
05	XXXXC01	Complementary Mathematics I	3	20	80	100	4	0	4
06	XXXXC01	Optional Complementary I	3	20	80	100	4	0	4
Total			**20**			**600**	**23**	**2**	**25**

Semester II

No	Course Code	Course Name	C	Marks			Hrs/wk		
				I	E	T	L	P	T
07	XXXXA03	Reading Literature in English	4	20	80	100	5	0	5
08	XXXXA04	Reading on Indian Constitution Secularism and Sustainable Environment	3	20	80	100	4	0	4
09	XXXXA08	Literature in Languages other than English	4	20	80	100	5	0	5
10	BCS2B02	Problem Solving using C	3	20	80	100	1	2	3
11	BCS2B03	Programming Laboratory I: Lab Exam of 1st& 2nd Sem. HTML & Programming in C	2	20	80	100	0	0	0
12	XXXXC02	Complementary Mathematics	3	20	80	100	4	0	4
13	XXXXC02	Optional Complementary II	3	20	80	100	4	0	4
Total			**22**			**700**	**23**	**2**	**25**

Semester III

No	Course Code	Course Name	C	Marks			Hrs/wk		
				I	E	T	L	P	T
14	XXXXA11	Numerical Skill	4	20	80	100	4	0	4
15	XXXXA12	General Informatics	4	20	80	100	4	0	4
16	BCS3B04	Data Structures using C	4	20	80	100	3	4	7
17	XXXXC02	Complementary Mathematics III	3	20	80	100	5	0	5

18	XXXXC02	Optional Complementary III	3	20	80	100	5	0	5
Total			**18**			**500**	**21**	**4**	**25**

Semester IV

No	Course Code	Course Name	Credit	Marks			Hrs/wk		
				I	E	T	L	P	T
19	XXXXA13	Entrepreneurship	4	20	80	100	4	0	4
20	XXXXA14	Basics of Audio & Video Media	4	20	80	100	4	0	4
21	BCS4B05	Database Management System and RDBMS	4	20	80	100	3	4	7
22	BCS4B06	Programming Laboratory II: Lab Exam of 3rd& 4th Sem. Data Structures & RDBMS	2	20	80	100	0	0	0
23	XXXXC03	Complementary Mathematics IV	3	20	80	100	5	0	5
24	XXXXC02	Optional Complementary IV	3	20	80	100	5	0	5
Total			**20**			**600**	**21**	**4**	**25**

Semester V

No	Course Code	Course Name	Credit	Marks			Hrs/wk		
				I	E	T	L	P	T
25	BCS5B07	Computer Organization and Architecture	4	20	80	100	5	0	5
26	BCS5B08	Java Programming	4	20	80	100	3	3	6
27	BCS5B09	Web Programming Using PHP	4	20	80	100	3	3	6
28	BCS5B10	Principles of Software Engineering	4	20	80	100	4	0	4
29	XXX5DXX	Open Course	2	10	40	50	2	0	2
		Project Work	0				0	2	2
Total			**18**			**450**	**17**	**8**	**25**

List of Open Courses (XXX5DXX)

No	Course Code	Course Name
29	BCS5D01	Introduction to Computers & Office Automation
29	BCS5D02	Web Designing
29	BCS5D03	Introduction to Problem Solving and C Programming
29	BCS5D04	Introduction to Data Analysis using spread sheet

Semester VI

No	Course Code	Course Name	Credit	Marks			Hrs/wk		
				I	E	T	L	P	T
30	BCS6B11	Android Programming	4	20	80	100	4	1	5
31	BCS5B12	Operating Systems	4	20	80	100	4	1	5
32	BCS5B13	Computer Networks	4	20	80	100	5	0	5
33	BCS5B14	Programming Laboratory III: Lab Exam of 5th Sem. Java & PHP Programming	2	20	80	100	0	0	0
34	BCS5B15	Programming Laboratory IV: Android & Linux shell Programming	2	20	80	100	0	4	4
35	BCS5B16X	Elective Course	3	20	80	100	4	0	4
36	BCS5B17	Project Work	3	30	120	150	0	2	2
Total			**22**			**750**	**17**	**8**	**25**

List of Electives (BCS5B17X)

No	Course Code	Course Name
35	BCS6B16A	System Software
35	BCS6B16B	Microprocessor and Applications
35	BCS6B16C	Discrete Structures

35	BCS6B16D	Computer Graphics
35	BCS6B16E	Technical Writing
35	BCS6B16F	Fundamentals of Life Skill Education

SEMESTER I

BCS1B01 – COMPUTER FUNDAMENTALS

Course Number:4
Contact Hours per Week:3 (1T + 2L)
Number of Credits:3
Number of Contact Hours:45 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objectives

- To equip the students with fundamentals of Computer
- To learn the basics of Computer organization
- To equip the students to write algorithm and draw flow chart for solving simple problems
- To learn the basics of Internet and webpage design

Prerequisites

- Background of the basic science at +2 level

Course Outline

Unit I[3 T]

Concepts of Hardware and Software: Computer Languages, Language Translators, Features of good language, Basics Computer Organization: Von Neumann Model, Input Unit, Output Unit, Storage Unit, Control Unit, Memory Hierarchy, Primary Storage, Cache Memory, Registers, Secondary Storage Devices, Basics of Hardware Components – SMPS, Motherboard, Add-on Cards, Ports, Memory, Adapters, Network cables, Basic Computer Configuration

Unit II [5 T]

Number Systems and Boolean Algebra – Decimal, Binary, Octal and Hexadecimal Numbers, Arithmetic involving Number Systems, Inter Conversions of Number Systems, 1's and 2's Complements, Complement Subtractions, Digital Codes – Binary Coded Decimal (BCD), ASCII Code ,Unicode, Gray Code, Excess-3 Code. Boolean Algebra: Boolean Operations, Logic Expressions, Postulates, Rules and Laws of Boolean Algebra, DeMorgan's Theorem, Minterms, Maxterms, SOP and POS form of Boolean Expressions for Gate Network, Simplification of Boolean Expressions using Boolean Algebra and Karnaugh Map Techniques (up to 4 variables)

Unit III [3 T]

Fundamentals of Problem Solving – The Problem Solving Aspect, Top-down Design, Definition – Algorithm, Flowchart, Program - Properties of Flowcharts – Flowchart Symbols for Designing Application Programs, Sample Algorithms – Sum, Average, Finding Smallest Number, Checking Odd/Even Number, Prime Number, Quadratic Equation

Unit IV [2T + 15L]

Basics of Web Design – www, W3C, Web Browser, Web Server, Web Hosting, Web Pages, DNS, URL, Introduction to HTML, XHTML, DHTML, HTTP.

Overview of HTML 5 – Basic Formatting Tags: heading, paragraph, break, underline, bold, italic, superscript, subscript, font and image, attributes: align, color, bgcolor, font face, border, size, navigation links using anchor tag: internal, external, mail and image, lists: ordered, unordered and definition, HTML media tags: audio and video

Unit V [2T+15L]

Creating Simple Tables: row, col, heading, cell, border, spanning – Form Controls: Input types – text, password, text area, button, checkbox, radio button, select box, hidden controls, frames and frame sets

CSS: Introduction - Concept of CSS, Creating Style Sheet: inline and internal, CSS Properties, CSS Styling: Background, Text Format, Controlling Fonts - Working with Block Elements and Objects, CSS ID and Class

Text Books

1. Sinha. P.K, Computer Fundamentals, BPB Publications
2. Ram. B, Computer fundamentals, New Age International Pvt. Ltd Publishers
3. Rajaraman V and Radhakrishnan, An introduction to Digital computer Design, PHI,
4. HTML 5 Blackbook,Dream Tech Press,2016 Edition

Reference Books

1. Thomas L Floyd, Digital Fundamentals, Universal Book Stall
2. Bartee T.C, Digital Computer Fundamentals, THM

SEMESTER II

BCS2B02 – Problem Solving Using C

Course Number:10
Contact Hours per Week: 3(1T+2L)
Number of Credits:3
Number of Contact Hours:45Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To equip the students with fundamental principles of Problem Solving aspects.
- To learn the concept of programming
- To study C language
- To equip the students to write programs for solving simple computing problems

Prerequisites

- Knowledge of basic computer operations

Course Outline

Unit I [2T+ 3L]

Introduction to C Programming - Overview and Importance of C, C Program Structure, Sample programs.

Familiarization of Integrated Development Environment - Invoking IDE, Opening a new window in IDE, Writing, Saving and Compiling a C program, making an Executable File.

Elements of C Language and Program Constructs: Character Set, C Tokens, Keywords and Identifier, Constants, Variables, Data types, Variable Declaration and Assignment of Values, Symbolic Constant Definition.

Unit II [2T+ 5L]

C Operators - Arithmetic operators, relational operators, and logical operators, assignment operators, increment and decrement operators, conditional operators, special operators, arithmetic expressions, evaluation of expressions, precedence of arithmetic operators, Type conversion in expressions, operator precedence and associativity, Mathematical Functions, I/O operations - Library functions.

Unit III [3T + 6L]

Data input output functions - Simple C programs – Flow of Control - Decision making with IF statement, Simple IF statement, If-else statement, Nesting of If-else and else-if Ladder, Switch statement, Conditional operator, goto statement. Looping - While loop, Do-While, and For Loops, Nesting of loops, jumps in loop, skipping of loops.

Unit IV [4T+ 8L]

Arrays and Strings - One dimensional array, two dimensional and multi-dimensional arrays, strings and string manipulation functions.

The Concept of modularization and User defined functions - Definition - Multifunction Program, proto-types, Passing arguments, calling functions, various categories of functions, Nesting of functions and recursion, functions and arrays, scope and lifetime of variables in functions, multi-file programs.

Structures & Union structure definition - giving values to members, structure initialization, comparison of structure variables, arrays of structures, arrays within structures, structures within arrays, structures and functions, Unions, bit fields.

Unit V [4T+8L]

Pointers - Understanding pointers, accessing the address of a variable, declaring and initializing pointers, accessing a variable through its pointer, pointer expressions, pointer and arrays, pointer and character string, pointers and functions, pointers and structures, pointer to pointer dynamic memory allocation.

Files: Creating, Processing, Opening and Closing a datafile, command line operations

Textbooks

1. Balaguruswami. E, Programming in ANSI C, Tata McGraw-Hill Education, 2008

References

1. Brian W. Kernighan & Dennis M. Ritchie, The C Programming Language, Prentice Hall, 2nd Edition 1998
2. Yashavant P. Kenetkar, Let us C
3. ByranGotfried, Schaums Outline series Programming with C
4. Ashok N. Kamthane, Programming in C, Pearson, 2nd Edition

BCS2B03 - Programming Laboratory I: Lab Exam of 1st & 2nd Semester HTML & Programming in C

Course Number: 11
Number of Credits: 2
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To make the students learn web designing
- To make the students learn programming environments.
- To practice procedural programming concepts.
- To make the students equipped to solve mathematical or scientific problems using C

Prerequisites

- Knowledge in operating computer.
- Knowledge in Computer fundamentals.

Course Outline:

Part A: HTML lab work
List of Exercises:

1) Design a web page to display the information of Computer Science department of your college by usingbasic page tags. Display the information in the form of paragraphs/sentences. Alsouse effects to highlight the information like bold, italic or underline.

2) (a) Create a webpage with HTML describing your department. Use paragraph and list tags.
   (b) Apply various colors to suitably distinguish key words. Also apply font styling like italics, underline and two other fonts to words you find appropriate. Also use header tags.

3) Use the above webpage to
   a) Create links on the words e.g. “Wi-Fi” and “LAN” to link them to Wikipedia pages.
   b) Insert an image and create a link such that clicking on image takes user to other page.
   c) Change the background color of the page. At the bottom create a link to take user to the top of the page.

4) Design a page to display the information in table format. Display the list of colleges offering B.Sc. Computer Science along with the details College Name, Address, Contact no. Address column will consist of sub columns as House Name, Post, City and pin code.

(Purpose: - Introduction of table tags along with the sub columns and other supportive tags like caption, cell spacing, cell padding etc.)

5) Create a webpage to create a photo Album. When the user clicks on the Image and Video Link it should open the corresponding album.

6) Design web pages which display the product images and its information with it. The products are computer, printers and laptop. The information displayed of product should be configuration/ technical details, price etc.

   (Purpose: - Study image tag. Display image in tabular form along with the other text information.)

7) Design a page to display the B.Sc. Computer Science syllabus by using List tag.

   (Purpose: - Introduce list tag to display data in ordered or unordered format as main, sub main, sub-sub main, etc by using nos. or special types of bullets.)

8) Create HTML pages using Hyper Links

   i. File Link

   ii. Single Page Link

9) Create a hyperlink to show the information and syllabus of B.Sc. Computer Science. When click on the links each page should display the objective of respective course, Lesson plan, Course durationetc.

   (Purpose: - Introduce anchor tag to create links between pages. One can able to transfer the control to next page, previous page or to a specific page like Home page.)

10) Design an HTML page describing your University infrastructure. Use tables to provide layout to your HTML page.

11) Use <span> and <div> tags to provide a layout to the above HTML page instead of a table layout.

12) HTML pages with Tables

    i. Use Tables to layout HTML pages

    ii. Use <span> and <div> tags to provide a layout to the above page instead of table layout

    iii. Use Frames to divide the page contents into different parts

    iv. Embed Audio and Video into your HTML webpage

13) Design a page to display the information in table format. Display the list of colleges offering Computer Science along with the details College Name, Address, Contact no.Address column will consist of sub columns as Post Office, City, District and Pin code.

    (Purpose: - Introduction of table tags along with the sub columns and other supportive tags like caption, cell spacing, cell padding etc.)

14) Create an HTML page to show the use of Navigation Frame

15) Create an HTML page to show the use of Floating Frame

16) Create an HTML page to show the use of Inline Frame

17) Design web pages to display the information about your college and UG Programmes offered in your college. Divide the page into three frames. The top frame should display the title of the college, left frame shout display the UGProgrammes and the right frame display the details of selected programme like fees, syllabus etc.

(Purpose: - Study frame tag which allow to divide the screen into no of sections.)

18) Design web pages to accept the student information. Student should enter the details like first name, last name, middle name, city up to 25 characters, and address up to 50 characters. Show the combo box to select the qualification, option button for gender selection. Display the information accepted in a formatted form.

(Purpose: - Study form tag which allow to design the formatted screen to accept the information from the user.)

19) Design a website to show the use of the following input controls

   i. Checkbox,

   ii. Radio button,

20) Design a website to show the use of the following input controls

   i. Select box,

   ii. Hidden controls

21) Design CSS style sheet to define settings for heading, body, table and links.

(Purpose: - Study CSS style sheet facility. This allows setting the default settings for all the pages.)

22) Design a webpage to show the use of External Style Sheets

Part B: C Programming

Write programs to do the following:

1. Find the sum of digits and reverse of a number.
2. Find the distance between two points.
3. Find the factorial of a number.
4. Find the Nth Fibonacci number using recursion.

5. Print the reverse of a string using recursion.
6. Solve the problem of Towers of Hanoi using recursion.
7. Find Sin(x) and Cos(x) in the range 0^o to 180^o (interval 30^o) using functions.
8. Create a pyramid using ‘*’.
9. Display the multiplication tables up to the number N.
10. Find the number of words in a sentence.
11. Perform matrix addition, subtraction, multiplication, inverse, and transpose using pointers and functions.
12. Replace a part of the string with another string.
13. Find the power of a number using structure and union.
14. Find the average of prime numbers in a group of N numbers using function.
15. Find the sum of the series $S = 1 + (½)^2 + (1/3)^3 + ...$ to 0.0001% accuracy.
16. Display the rightmost digit in a floating point number.
17. Create a pattern with the number N.

    e.g. N = 39174 Pattern:
    ```
    3 9 1 7 4
    9 1 7 4
    1 7 4
    7 4
    4
    ```
18. Display the short form of a string. E.g. Computer Science : CS
19. Currency conversion (any four currencies)
20. Find the currency denomination of a given amount.
21. Prepare sales bill using array of structures.
22. Addition and subtraction of complex numbers using structure.
23. Find the amstrong numbers within a given range.
24. Check for palindrome string/number.
25. Check for leap year.
26. Find the number of special characters in a given string.
27. Store and read data from a text file.
28. Write odd and even numbers into separate files.
29. Swapping of two numbers using call-by-reference method.
30. Copy the contents of one file into another one using command line parameters.
31. Base conversion of numbers.
32. Calculate the percentage of marks obtained for N students appeared for examination in M subjects using array of structures.
33. Display a table of the values of function y = exp(-x) for x varying from 0.0 to N in steps of 0.1
34. Design a Scientific Calculator and include as many functions as possible.
35. Merge two numeric arrays in sorted order.
36. Fill upper triangle with 1, lower triangle with -1 and diagonal elements with 0.

37. Count the occurrence of different words in a sentence.
38. Convert an input amount into words.
39. Convert a time in 24 hour clock to a time in 12 hour clock using structure.
40. Change the date/time format using structure.

Include any 15 HTML programs and 20 C programs in the record book.

Note: All lab works should be neatly recorded in a Laboratory Record Book in written form. However Program results can be pasted in the left hand side of the fare record. All students should have a rough record (observation note book) too, in which they write all the works to be carried out in the lab prior to his/her entering the lab. He/She may also note down the i/p and o/p that he gives for program verification in the observation note book (rough record).

UNIVERSITY OF CALICUT

നിർമ്മായ കർമ്മണാ ശ്രീ:

SEMESTER III

BCS3B04 – Data Structures Using C

Course Number: 16
Contact Hours per Week: 7 (3T + 4L)
Number of Credits: 4
Number of Contact Hours: 105 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To introduce the concept of data structures
- To make the students aware of various data structures
- To equip the students implement fundamental data structures

Prerequisites

- Knowledge in C Programming Language

Course Outline

UNIT – I[6 T + 4L]

Introduction: Elementary data organization, Data Structure definition, Data type vs. data structure, Categories of data structures, Data structure operations, Applications of data structures, Algorithms complexity and time-space trade off, Big-O notation.

Strings: Introduction, strings, String operations, Pattern matching algorithms

UNIT – II [10 T + 14 L]

Arrays: Introduction, Linear arrays, Representation of linear array in memory, Traversal, Insertions, Deletion in an array, Multidimensional arrays, Parallel arrays, sparse matrix.

Linked List: Introduction, Array vs. linked list, Representation of linked lists in memory, Traversal, Insertion, Deletion, Searching in a linked list, Header linked list, Circular linked list, Two-way linked list, Applications of linked lists, Algorithm of insertion/deletion in Singly Linked List (SLL).

UNIT – III [9 T + 14 L]

Stack: primitive operation on stack, algorithms for push and pop. Representation of Stack as Linked List and array, Stacks applications: polish notation, recursion.

Introduction to queues: Primitive Operations on the Queues, Circular queue, Priority queue, Representation of Queues as Linked List and array, Applications of queue: Algorithm on insertion and deletion in simple queue and circular queue.

UNIT – IV [10 T + 14 L]

Trees - Basic Terminology, representation, Binary Trees, Tree Representations using Array & Linked List, Basic operation on Binary tree: insertion, deletion and processing, Traversal of binary trees: In order, Pre-order & post-order, Algorithm of tree traversal with and without recursion, Binary Search Tree, Operation on Binary Search Tree, expression trees, implementation using pointers, applications.

UNIT – V [10 T + 14 L]

Introduction to graphs, Definition, Terminology, Directed, Undirected & Weighted graph, Representation of graphs, graph traversal- depth-first and breadth-first traversal of graphs, applications.

Searching: sequential searching, binary searching, Hashing – linear hashing, hash functions, hash table searching; Sorting: Quick Sort, Exchange sort, Selection sort and Insertion sort.

TEXT BOOKS

1. Seymour Lipschutz, “Data Structures”, Tata McGraw- Hill Publishing Company Limited, Schaum’s Outlines, New Delhi.
2. YedidyanLangsam, Moshe J. Augenstein, and Aaron M. Tenenbaum, “Data Structures Using C”, Pearson Education., New Delhi.
3. Horowitz and Sahani, “Fundamentals of data Structures”, Galgotia Publication Pvt. Ltd., NewDelhi.

REFERENCE BOOKS

1. Trembley, J.P. And Sorenson P.G., “An Introduction to Data Structures With Applications”, Mcgraw- Hill International Student Edition, New York.
2. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, Addison- Wesley, (An Imprint of Pearson Education), Mexico City.
3. A.K.Sharma, Data Structures Using C, Pearson, Second edition, 2011
4. Nair A.S., Makhalekshmi, Data Structures in C, PHI, Third edition 2011.

5. R. Kruse etal, “Data Structures and Program Design in C”, Pearson Education Asia, Delhi-2002
6. K Loudon, “Mastering Algorithms with C”, Shroff Publisher & Distributors Pvt. Ltd.

SEMESTER IV

BCS4B05 – Database Management System and RDBMS

Course Number: 21
Contact Hours per Week: 7 (3T + 4L)
Number of Credits: 4
Number of Contact Hours:105 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To learn the basic principles of database and database design
- To learn the basics of RDBMS
- To learn the concepts of database manipulation SQL
- To study PL/SQL language

Prerequisites

- Basic knowledge of computers, data structures and programming

Course Outline

Unit I [6 T+ 2L]

Database System concepts and applications Introduction to databases, File Systems vs. DBMS, Advantages and Disadvantages of using DBMS Approach, Database administrators and user, Data Models, Schemas, and Instances, Types of Data Models, Three Schema Architecture and Data Independence, Database Languages and Interfaces.

Unit II [9 T+ 4L]

Entity-Relationship Model - Conceptual Data Models for Database Design Entity Relationship Models, Concept of Entity, Entity Sets, Relationship Sets, Attributes, Domains, Constraints, Keys, Strong and Weak Entities, Concepts of EER.

Relational Data Model Relations, Domains and Attributes, Tuples, Keys. Integrity Rules, Relational Algebra and Operations, Relational Calculus and Domain Calculus, Relational Database Design using ER to Relational Mapping.

Unit III [10 T+10L]

Relational Database Design - Relational database design Anomalies in a Database, Normalization Theory, Functional Dependencies, First, Second and Third Normal Forms, Relations with more than one Candidate Key, Good and Bad Decompositions, Boyce Codd Normal Form, Multivalued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

Unit IV [10 T +20L]

SQL Concepts: Basics of SQL, DDL, DML, DCL, Tables – Create, Modify and Delete table structures, Rename and Drop tables, Defining constraints – Primary key, foreign key, unique, not null, check, IN operator

Select Command, Logical Operators, Functions - aggregate functions, Built-in functions – numeric, date, string functions, set operations, sub-queries, correlated sub-queries, Use of group by, having, order by, join and its types, Exist, Any, All. View - Creation, Renaming the column of a view, destroys view.

Unit V [10 T+24L]

Transaction Management and Concurrency Control - Transaction Properties (ACID), states, Commit, Rollback; Concurrency Control Lost update problems, Locks, two phase locking.

Programming with SQL: Data types: Base and Composite, Attributes. Variables – Constants - Using set and select commands, Control Structures: IF, IF THEN ELSE, IF THEN ELSEIF, CASE. Loops: LOOP, EXIT, CONTINUE, WHILE, FOR, and FOREACH - Looping Through Arrays - Looping Through Query Results. Security: Locks: Table-level Lock, Row-level Lock, Deadlock, Advisory Lock. Cursors: Boud and Unbound Cursors, Declaration, Opening, Working with cursors: FETCH, MOVE, UPDATE/DELETE, CLOSE, Looping through a Cursor. Concept of Stored Procedures – Advantages and Disadvantages – Creation – Parameters Setting for Function- Alter – Drop – Grant and Revoke - Passing and Returning data to/from Stored Procedures - Using stored procedures within queries – Triggers: Creation, Modification, Deletion, Error Handling: Control Structures, Cursors, Functions, Triggers.

Textbooks:

1. Abraham Silberschatz, Henry F Korth, S.Sudharshan, Database System Concepts, 6thEdition
2. W. Gilmore, Beginning PHP and PostgreSQL 8: From Novice to Professional, Goels Computer Hut (2007), ISBN: 9788181286000
3. PosgreSQL Official Documentation Online

References:

1. Alex Krigel and Boris M.Trukhnov, SQL Bible, Wiley pubs

2. Paul Nielsen, Microsoft SQL Server 2000 Bible, Wiley Dreamtech India Pubs.
3. CJ Date, Introduction to Database Systems, Addison Wesley.
4. Ramkrishnan, Database Management Systems, McGraw Hill

BCS4B06- Programming Laboratory II: Lab Exam of 3rd & 4th Semester Data Structures & RDBMS

Course Number: 22
Number of Credits: 2
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To make the students equipped to solve mathematical or scientific problems using C
- To learn how to implement various data structures.
- To provide opportunity to students to use data structures to solve real life problems.

Prerequisites

- Knowledge in operating computer.
- Theoretical knowledge in Data structures.
- Knowledge in Database

Course Outline

Part A: Data structure – Lab Questions

1. Sort a given list of strings
2. Reverse a string using pointers.
3. Implement Pattern matching algorithm.
4. Search an element in the 2-dimensional array
5. Append 2 arrays
6. Merge two sorted array into one sorted array.
7. Search an element in the array using iterative binary search.
8. Search an element in the array using recursive binary search.
9. Implement sparse matrix
10. Implement polynomial using arrays
11. Implement singly linked list of integers.
12. Delete a given element from a singly linked list
13. Sort a singly linked list.
14. Delete an element from a singly linked list
15. Implement a doubly linked list of integers
16. Implement a circular linked list.
17. Implement polynomial using linked list

18. Addition of 2 polynomials
19. Implement Stack using array
20. Implement Stack using linked list
21. Infix expression into its postfix expression
22. Implement Queue using array
23. Implement Queue using linked list
24. Implement a binary search tree of characters.
25. Traverse a binary search tree non recursively in preorder
26. Traverse a binary search tree non recursively in inorder
27. Traverse a binary search tree non recursively in postorder
28. Traverse a binary search tree recursively in preorder
29. Traverse a binary search tree recursively inorder
30. Traverse a binary search tree recursively postorder.
31. Delete an element from a binary search tree.
32. Search an element in a binary search tree
33. Implement linear sort
34. Implement bubble sort
35. Implement exchange sort
36. Implement selection sort.
37. Implement insertion sort.
38. Implement quick sort.
39. Implement merge sort.
40. Implement heap sort

Part B: DBMS– Lab Questions

1. Create a table employee with fields (EmpID,EName, Salary, Department, Age). Insert some records. Write SQL queries using aggregate functions for
   A. Display the total number of employees.
   B. Display the age of the oldest employee of each department.
   C. Display departments and the average salaries
   D. Display the lowest salary in employee table
   E. Display the highest salary in sales department;
2. A trading company wants to keep the data of their Order Processing Application using the following relations.

Customer_Master

Customer_Number	-	Primary Key
Customer_Name	-	Not NULL
Address	-	
Pincode	-	

Order_Master

Order_Number	-	Primary Key
Order_date	-	Not NULL
Customer_Number	-	Refers Customer_master table
Order_amount	-	Not NULL

Order_Detail

Line_Number	-	Primary Key
Order_Number	-	Primary Key
Item_No	-	Not NULL, Refers ITEM table
Quantity	-	Not NULL

ITEM

Item_No	-	Primary Key
UnitPrice	-	Not NULL

SHIPMENT

Order_Number	-	Primary Key
Warehouse_No	-	Primary Key, Refers Warehouse table
Ship_Date	-	Not NULL with Integrity Check

WAREHOUSE

Warehouse_No	-	Primary Key
City	-	Not NULL

A. Create the above tables by properly specifying the primary keys and foreign keys.
B. Enter at least five tuples for each relation.
C. Produce a listing: Cust_Name, No_of_orders, Avg_order_amount, where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer.
D. List the Order_Number for orders that were shipped from ***all*** the warehouses that the company has in a specific city.
E. Demonstrate the deletion of an item from the ITEM table and demonstrate a method of handling the rows in the ORDER_ITEM table that contains this particular item.

3. In this session you need to create database for an Employee management system of an ABC organization. The details about different tables are given below. According to that you can proceed further and create tables using PostgreSQL/ MySQL

Create the following tables with the specified constraints:

Department:

Department Number	-	Primary Key
Department Name	-	Not NULL unique
Manager ID	-	Refers to EmployeeID of employee table.
Manager		
Date of Joining	-	Not NULL.

Employee:

First Name	-	Not NULL
Middle Initials		
Last Name	-	Not NULL
Employee ID	-	Primary Key
Date of Birth	-	Not NULL
Address		
Gender	-	M or F
Salary	-	Range of 5000 to 25000
Date of Joining		
Department Number	-	Refers to Department Number of Department table.

Department location:

Department Number	-	Refers to Department number of department table.
Department Location	-	Not NULL.

Department number & Department location are combined Primary Key

Project:

Project Name	-	Not NULL.
Project Number	-	Primary Key.
Project Location	-	Not NULL.
Department number	-	Refers to department number of Department table.

Works-on:

Employee ID	-	Not NULL refers to Employee ID of Employee table.
Project Number	-	Not NULL refers to Project number of Project table.
Hours	-	Not NULL.

Employee ID & Project Number are combined primary key.

Dependent:

Employee ID	-	Refer to employee table Employee ID field
Dependent Name	-	Gender - M or F
Date of Birth	-	Not NULL
Relationship	-	Not NULL

Now enter a few sets of meaningful data and answer the following queries.

A. List the department wise details of all the employees.
B. Find out all those departments that are located in more than one location.
C. Find the list of projects.
D. Find out the list of employees working on a project.
E. List the dependents of the employee whose employee id is 001

4. These sessions is similar to the previous one, but in this session, assume that you are developing a prototype database of the College library management system, for that you need to create the following tables:

- Book Records
- Book details
- Member details and
- Book issue details

Book Records:

Accession Number
ISBN Number

Books:

ISBN Number
Author
Publisher
Price

Members:

Member ID
Member Name
Maximum Number of books that can be issued
Maximum Number of days for which book can be issued

Book Issue:

Member ID
Accession Number
Issue Date
Return Date

You must create constraints, including referential integrity constraints, as appropriate. Please note accession number is unique for each book. A book, which has no return date, can be considered as issued book. Enter suitable data into the tables. Now answer the following:

A. Insert data in all the three tables (use insert).
B. Insert appropriate description associated with each table and the column (use comment).
C. Display the structure of the tables.
D. Display the comments that you have inserted.
E. Using SELECT statement, write the queries for performing the following function:
   a. Get the list of all books (No need to find number of copies).
   b. Get the list of all members.
   c. Get the Accession number of the books which are available in the library.
   d. On return of a book by a member calculate the fine on that book.
   e. List of books issued on 01-Jan-2005.
   f. Get the list of all books having price greater than Rs. 500/-
   g. Get the list of members who did not have any book issued at any time.
   h. Get the list of members who have not returned the book.
   i. Display member ID and the list of books that have been issued to him/her from time to time.
   j. Find the number of copies of each book (A book accession number would be different but ISBN number would be the same).
   k. Find the number of copies available of a book of given ISBN number.
   l. Get the member ID and name of the members to whom no more books can be issued, because they have already got as many books issued as the number for which they are entitled.

5. This session is based on Lab 2 where you have created a library management system. In this session you have different query specification.

You must create appropriate forms, reports, graphs, views and data filtering, use of multilevel report, etc. to answer these queries.

A. Get the list of ISBN-Number, Book name, available copies of the books of which available copies are greater than zero.
B. Get the list of ISBN-Number, Book name, Total copies, available copies of the book of which available copies are greater than zero. List should be displayed in alphabetical order of book name.
C. Get the list of ISBN number, Book name, Author, total copies, cost (cost is price total copies). List should be displayed in descending order of cost.
D. Get the list of books issued to each member.
E. Write query to know the maximum and average price of the books.

F. Get the list of all existing members and the number of days for which a member is allowed to keep the book. Also find out the members who have got the maximum number of books issued.

G. Get the list of member codes of those members who have more than two books issued.

H. Find the details of the books presently issued to a member.

I. Create the history of issue of a book having a typical accession number.

J. To set the width of the book name to 35.

6. Create the following table and perform the necessary tasks defined below one by one. You must use the query tools/ SQL/ Reports/ Forms/ Graphs/Views/ using client/server wherever needed.

1. Create the following table named customer

Column name	**type**	**size**
Customer ID	Character	10
Name	Character	25
Area	Character	3
Phone	Numeric	7

Insert the appropriate data into table and do the following.

- Update Phone numbers of all customers to have a prefix as your city STD Code
- Print the entire customer table
- List the names of those customers who have e as second letter in their names.
- Find out the Customer belonging to area 'abc'
- Delete record where area is NULL.
- Display all records in increasing order of name.
- Create a table temp from customer having customer-id, name, and area fields only
- Display area and number of records within each area (use GROUP by clause)
- Display all those records from customer table where name starts with a or area is 'abc'.
- Display all records of those where name starts with 'a' and phone exchange is 55.

7. Answer the following queries using Library system as created earlier. You must create a view to know member name and name of the book issued tothem, use any inbuilt function and operators like IN, ANY, ALL,EXISTS.

a. List the records of members who have not been issued any book using EXISTS operator.

b. List the members who have got issued at least one book (use IN / ANY operator).

c. List the books which have maximum Price using ALL operator.

d. Display Book Name, Member Name, and Issue date of Book. Create a view of this query of the currently issued books.

8. Create a table of Employee (emp_number, name, dept_number, salary) and Department (dept_ number, dept_name). Insert some records in the tables through appropriate forms having integrity checks. Add some records in employee table where department value is not present in department table. Now answer the following query:

a. Display all records from employee table where department is not found in department table.

b. Display records from employee table in a report format with proper headings. This report must also contain those records where department number does not match with any value of department table.

c. Display those employee records who have salary less than the salary of person whose emp_number= A100.

d. Create another table: SalesData (RegionCode, City, SalespersonCode, SalesQty).

e. Display records where salesperson has achieved sales more than average sales of all sales persons of all the regions.

9. Create the following tables:

Order party: (Order number, Order date, customer code)
Order: Order number, Item code, Quantity
The key to the second table is order-number + item-code
Create a form for data entry to both the tables.

10. Create a table shop with fields Item_ID, Item_Name, Price, and Quantity. Write a procedure 'sales' to update the quantity by accepting Item_ID and Quantity as argument. Write PostgreSQL block to invoke the procedure

11. Implement student information system

12. SQL scripts to display various reports like Result of an Examination, Salary Report, Sales Report, Sales reports grouped on Sales person or item, etc

13. Write simple PostgreSQL anonymous blocks for displaying whole numbers from 1 to 100, odd numbers from 1 to 100, even numbers from 1 to 100, positive whole numbers up to a given number, odd numbers from 1 to a given number, even numbers from 2 to a given number, Fibonacci numbers up to 100, Strange numbers up to 1000, factorials of the numbers from 1 to 10, etc.

14. Create a table product with the fields(Product_code primary key, Product_Name, Category, Quantity, Price). Insert some records Write the queries to perform the following.

a. Display the records in the descending order of Product_Name

b. Display Product_Code, Product_Name with price between 20 and 50

c. Display the Product_Name and price of categories bath soap, paste, washing powder
d. Display the product details whose Quantity less than 100 and greater than 500
e. Display product names starts with 's'
f. Display the products which not belongs to the category 'paste'
g. Display the product names whose second letter is 'a' and belongs to the Category 'washing powder'

15. Create a STUDENT table with following fields:

Field Name	**Type**	**Width**
RegNo	character	10
Name	character	20
Paper1	numeric	3
Paper2	numeric	3
Paper3	numeric	3
Paper4	numeric	3
Paper5	numeric	3
Total	numeric	3
Result	character	6
Grade	character	1

Enter the RegNo, Name and Marks in 5 Papers of at least 10 students. Write a SQL program to process the records to update the table with values for the fields Total (Paper1+Paper2+Paper3+Paper4+Paper5), Result (‘Passed’ if total is greater than or equal to 50% of the total; ‘Failed’ otherwise), and Grade (‘A’ if mark obtained is greater than or equal to 90% of the total mark, ‘B’ if mark obtained is greater than or equal to 75% of the total mark, ‘C’ if mark obtained is greater than or equal to 60% of the total mark, ‘D’ if mark obtained is greater than or equal to 50% of the total mark, and ‘F’ if mark obtained is less than 50% of the total mark). Display a report in descending order of the total mark, showing the data entered into the table along with the total marks, result and grade.

16. An examination has been conducted to a class of 10 students and 4 scores of each student have been provided in the data along with their reg_no, name, total and avg_score.

Assign null values to the fields total and average. Write Postgresql block to do the following

Find the total and average of each student. Update the table with the calculated values Assign a letter grade to each student based on the average Score as

avg_score between 90 and 100	-	A
avg_score 75 -89	-	B
avg_score 60- 74	-	C
avg_score 50 -59	-	D
avg_score below 50	-	Failed

17. Prepare a salary report of the employees showing the details such as:

EmpNo, Name, Basic Pay, DA, Gross Salary, PF, Net Salary, Annual Salary and Tax For this purpose, create a table named SALARIES having the following structure.

Field Name	Type	Width
EmpNo	Character	10
Name	Character	20
Basic	Numeric	6

Enter the records of at least 10 employees. Use the following information for calculating the details for the report:

DA is fixed as the 40% of the basic pay.
PF is fixed as 10% of the basic pay.
Gross Salary is (Basic Pay + DA).
Net Salary is (Gross Salary – PF)
Annual Salary is (12 * Net Salary)

Tax is calculated using the following rules:

If annual salary is less than 100000, No Tax
If annual salary is greater than 100000 but less than or equal to 150000, then the tax is 10% of the excess over 100000.
If annual salary is greater than 150000 but less than or equal to 250000, then the tax is 20% of the excess over 150000.
If annual salary is greater than 250000, then the tax is 30% of the excess over 250000.

18. Generate a Hospital information system that can generate the following reports:

- Patients who belongs to in-patient category
- Patients who belongs to out-patient category

For this purpose, create a table named HOSPITAL having the following structure.

Field Name	Type	Width
PatientID	character	10
Name	character	20
Age	numeric	3
Doctor	character	20
PatientType	character	15
ConsultCharge	numeric	6
BloodTestCharge	numeric	6
XrayCharge	numeric	6
OtherCharges	numeric	6
TotalAmount	numeric	6

Enter the records of at least 10 patients. Write a SQL program to display the report in the ascending order of patient name.

19. Using the Hospital table created in Lab 16, generate a Hospital information system that can generate the following reports:

- Patients undergone blood test.
- Patients who have taken X-Rays

20. Design a Hotel Bill calculating system that generates hotel bills for the customers.

21. Design an Electricity Bill Report generating system that generates electricity bills details of customers for a month.

22. Generate a Library Information System that generates report of the books available in the library.

23. Programs involving multiple tables.

24. Create a table named Elec_Bill (Cust_No, Cust_Name, Units_Consumed, Bill_Amt). Set bill_amt as null. Write a PostgreSQL function to calculate the Bill_Amt by accepting Cust_No and Units_Consumed. Write a PostgreSQL block to update the calculated amount by invoking the function.

25. Create two tables Book (BookID, BookName, Author, Publisher) and Book_Del (Date_of_Del, BookID, BookName)

Create and application to generate a trigger before deleting a record from book table. The trigger procedure should insert the deleted BookID and BookName along with current date to the table Book_Del.

Include any 20 Data structure Lab questions and 15 DBMSLab questions in the record book.

Note: All lab works should be neatly recorded in a Laboratory Record Book in written form. However Program results can be pasted in the left hand side of the fare record. All students should have a rough record (observation note book) too, in which they write all the works to be carried out in the lab prior to his/her entering the lab. He/She may also note down the i/p and o/p that he gives for program verification in the observation note book (rough record).

SEMESTER V

BCS5B07|Computer Organization & Architecture

Course Number:25
Contact Hours per Week:5T
Number of Credits:3
Number of Contact Hours:75 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objectives of the Course

- To learn logic gates, combinational circuits and sequential circuits
- To learn basics of computer organization and architecture

Prerequisites

- Boolean algebra

Course Outline

Unit I [14 T]

Digital Logic - Positive and negative logic, logic gates ,NOT gate, OR gate, AND gate, XOR and X-NOR gates, Universal gates- NAND gate, NOR gate,. Combinational circuits- Half adder, half subtractor, full adder, full subtractor, ripple carry adders, look-ahead carry adders, decoders, BCD to 7-segment decoder, encoders, multiplexers and demultiplexers.

Unit II [16 T]

Sequential Logic Circuits: Edge triggering, Pulse triggering ,SR latch, SR flip flop, JK flip flop, Master Slave JK flip flop, D flip flop, T flip flop. Shift register: serial in - serial out, serial in - parallel out, parallel in - serial out, parallel in-parallel out configurations. counters (asynchronous & synchronous), up/down counter, decade counter, mod N counter, Ring counter, Johnson's counter

Unit III [15 T]

Basic Computer Organization and Design: Instruction Codes , Computer Registers, Computer Instructions, Instruction types, Timing and Control, Instruction Cycle, Memory reference Instructions, Register reference instructions, Input, Output and Interrupt Design of Basic Computer, Design of Accumulator logic

Unit IV [15 T]

Micro programmed Control: Control Memory, Address sequencing, Micro program Example, Design of control unit. Processor Organization: general register organization, stack organization, instruction formats, addressing modes, data transfer and manipulation, program control

Unit V [15 T]

Memory Organization: Memory mapping, Associative memory, Cache memory, Virtual Memory, Memory Management Hardware, hit/miss ratio, Input-Output Organization: Peripheral devices, I/O interface, Modes of Transfer-asynchronous and synchronous, Priority Interrupt, Strobe Control, Handshaking. Direct Memory Access, Input-Output Processor, Serial Communication. I/O Controllers

Textbooks:

1. Thomas L Floyd, Digital Fundamentals, Universal Book Stall (Unit I and II)
2. M. Morris Mano, *Computer System Architecture*PHI (Unit III – V)

References:

1. Rajaraman V. &Radhakrishnan, An Introduction to Digital Computer Design, PHI.
2. William Stallings, *Computer Organization and Architecture*, PHI.
3. Malvino& Leach, Digital Principles & Applications, TMH
4. Jain R.P. , Modern Digital Electronics, TMH
5. Malvino, Digital Computer Electronics, TMH
6. Bartee T.C., Digital Computer Fundamentals, THM
7. William H. Gothmann, Digital Electronics: An Introduction to Theory and Practice, PHI

BCS5B08|Java Programming

Course Number:26
Contact Hours per Week:6 (3T + 3L)
Number of Credits:4
Number of Contact Hours:90 Hrs.
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Objectives

- To review on concept of OOP.
- To learn Java Programming Environments.
- To practice programming in Java.
- To learn GUI Application development in JAVA.

Prerequisites

- Knowledge in OOP & Programming

Course Outline

Unit I [7 T + 4L]

Introduction to OOPS, Characteristics of OOPS, Object oriented languages, comparison between procedural and object oriented programming, basic principles of Object Orientation-class, object, abstraction, encapsulation, inheritance, polymorphism, modularity, and message passing. Features of object orientation - attributes, state, identity, operation, behaviour.

Unit II [8 T + 8L]

Introduction to Java: History, Versioning, The Java Virtual Machine, Byte code, Writing simple java program, Language Components: Primitive Data Types, Comments, Keywords, literals, The if Statement, The switch Statement, The for Statement, The while and do while Statements, The break Statement, The continue Statement, Operators – Casts and Conversions, Arrays. Introduction to classes and methods, constructors, Passing Objects to Methods, Method Overloading, Static and final, The this Reference, finalize, inner and nested classes. Inheriting class, extends, member access and inheritance, super keyword, Object class. Dynamic method dispatch, method overriding, abstract class, interface, packages, import statement.

Unit III [10 T + 10 L]

Exceptions, I/O and Threads Input and Output in Java: The File Class, Standard Streams, Keyboard Input, File I/O Using Byte Streams, Character Streams, File I/O Using Character Streams - Buffered Streams, File I/O Using a Buffered Stream, Keyboard Input Using a Buffered Stream, Writing Text Files. Threads: Threads vs. Processes, Creating Threads by Extending Thread, Creating Threads by Implementing Runnable, Advantages of Using Threads, Daemon Threads, Thread States, Thread Problems, Synchronization. Exceptions: Exception Handling, The Exception Hierarchy, throws statement, throw statement, Developing user defined Exception Classes- The finally Block.

Unit IV [10 T + 12 L]

Database Connectivity & Applets: Introduction to JDBC : The JDBC Connectivity Model, Database Programming, Connecting to the Database, Creating a SQL Query, Executing SQL Queries, Getting the Results, Updating Database Data, Executing SQL Update/Delete, The Statement Interface, The ResultSet Interface, ResultSetMetaData. Introduction to GUI Applications - Applets - Types of Applet, Applet Skeleton, Update method, repaint Methods, Html Applet tag and passing parameter to applet.

Unit V [10 T + 11 L]

Events and GUI Applications: Event Handling: The Delegation Event Model, Event Classes, Event Listener Interfaces, Adapter Classes. Java Desktop Applications, Introduction to the AWT, Overview of the AWT, Structure of the AWT, The AWT hierarchy, Containers, Components, Canvas, Frame Working with: Color, Font, FontMetrics, Simple Graphics- Point, line, Rectangle, Polygon, Controls - Button, , Checkbox, Choice, , Label, List, Scroll bar, TextArea, TextField, Layout Manager, MenuBar, Menu, MenuItem , Checkbox MenuItem.

Text Books

1. Herbert Scheldt, Java The Complete Reference, 8th Edition, Tata McGraw-Hill Edition, ISBN: 9781259002465

References

1. E Balaguruswamy, Programming in Java: A Primer, 4th Edition, Tata Mcgraw Hill Education Private Limited, ISBN: 007014169X.
2. Kathy Sierra, Head First Java, 2nd Edition, Shroff Publishers and Distributors Pvt Ltd, ISBN: 8173666024.
3. David Flanagan, Jim Farley, William Crawford and Kris Magnusson, Java Enterprise in a Nutshell: A Desktop Quick Reference, 3rd Edition, O'Reilly Media, ISBN: 0596101422.

BCS5B09|Web Programming using PHP

Course Number:27
Contact Hours per Week:6 (3T + 3L)
Number of Credits:4
Number of Contact Hours:90 Hrs.
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Remarks:- HTML portion covered in Computer Fundamentals and HTML paper is trimmed. Emphasis is given for HTML 5.

Objectives

- To review on concept of OOP.
- To learn Java Programming Environments.
- To practice programming in Java.
- To learn GUI Application development in JAVA.

Prerequisites

- Knowledge in OOP & Programming

Course Outline

Unit I [8 T + 8L]

Introduction web-documents: Static, Dynamic, Active - Web programming: client side and server side scripting. HTML 5: Document Structure, Elements, Attributes, Types of Elements and Attributes, Basic HTML Data types. Using HTML5 form elements: datalist, keygen, output, progress, meter. File uploading using forms - Frameset and frames. CSS: External CSS, CSS3 Syntax, Selector: Universal, Class, ID. Working with Lists and Tables, CSS ID and Class – Navigation Bar - Image Gallery – Image Opacity.

Unit II [8 T + 8L]

Javascript: Introduction, Client side programming, script tag, comments, variables. Including JavaScript in HTML: head, body, external. Data types. Operators: Arithmetic, Assignment, Relational, Logical. Conditional Statements, Loops, break and continue. Output functions: write, writeln, popup boxes: prompt, alert, confirm. Functions: Built-in Global Functions: alert(), prompt(), confirm(), isNan(), Number(), parseInt(). User Defined Functions, Calling Functions with Timer, Events Familiarization: onLoad, onClick, onBlur, onSubmit, onChange, Document Object Model (Concept). Objects: String, Array, Date.

Unit III [10 T + 10 L]

PHP: Introduction, Server side programming, Role of Web Server software, Including PHP Script in HTML: head, body, external. Comments, Data types, variables and scope, echo and print. Operators: Arithmetic, Assignment, Relational, Logical. Conditional Statements, Loops, break and continue. User Defined Functions.

Unit IV [9 T + 9 L]

Working with PHP: Passing information between pages, HTTP GET and POST method, Cookie, Session. String functions: strlen, strops, strstr, strcmp, substr, str_replace, string case, Array constructs: array(),list() and foreach(). Header().

Unit V [10 T + 10 L]

PHP & PostgreSQL: Features of PostgreSQL, data types, PostgreSQL commands – CREATE DATABASE, CREATE TABLE, DESCRIBE TABLE (\d table_name or using usinginformation_schema), SELECT, SELECT INTO, CREATE AS, DELETE, UPDATE, INSERT. PHP - PostgreSQL Integration: Establishing Database Connection (pg_connect(), pg_connection_status(), pg_dbname()), Getting Error String (pg_last_error()), Closing database Connection (pg_close()), Executing SQL statements (pg_query(), pg_execute()), Retrieving Data (pg_fetch_row(), pg_fetch_array(), pg_fetch_all(), pg_fetch_assoc(), pg_fetch_object(), pg_num_rows(), pg_num_fields() pg_affected_rows(), pg_num_rows(), pg_free_result()), Insertion and Deletion of data using PHP, Displaying data from PostrgreSQL database in webpage. Introduction to AJAX - Implementation of AJAX in PHP - Simple example for partial page update.

Textbook

1. HTML 5 Blackbook, Dreamtech Press, ISBN 9879351199076, 2016 Edition.
2. W. Gilmore, *Beginning PHP and PostgreSQL 8: From Novice to Professional* , Goels Computer Hut (2007), ISBN: 9788181286000

Reference

1. HTML 5 Blackbook, Dreamtech Press, ISBN 987-93-5119-907-6, 2016 Edition.
2. Jon Duckett, *Beginning Web Programming with HTML,XHTML, CSS*, Wrox.
3. Jim Converse & Joyce Park, *PHP & MySQL Bible*, Wiley.
4. PosgreSQL Official Documentation Online

BCS5B10|Principles of Software Engineering

Course Number:28
Contact Hours per Week:4T
Number of Credits:4
Number of Contact Hours:60 Hrs.
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Objectives

- To learn engineering practices in Software development.
- To learn various software development methodologies and practices.
- To learn and study various Evaluation methods in Software Development.

Prerequisites

- Knowledge in Programming

Course Outline

UNIT I [12T]

Software and Software Engineering: Overview of Software Engineering, Practice & Myths; Software Process; Generic process model- Framework Activity, Task Set, Process Patterns, Process Improvement; SDLC, Prescriptive process model- Waterfall Model, Spiral Model, Incremental Process Model, Evolutionary Process Model; Specialized Process Models: Component Based Development, the Formal Methods Models;

Agile development-Agile Process; Extreme Programming; Other Agile Process Models – ASD, Scrum, DSDM, FDD, LSD, Agile Modeling, Agile Unified Process..

UNIT II [12T]

Requirements Engineering- Establishing the Groundwork- Eliciting Requirements - Developing use cases - Building the requirements model - Negotiating, validating Requirements - Requirements Analysis-Requirements Modeling Strategies.

UNIT III [14T]

MODELING WITH UML: Concepts and Diagrams - Use Case Diagrams - Class Diagrams - Interaction Diagrams - State chart Diagrams – Activity Diagrams - Package Diagrams - Component Diagrams - Deployment Diagrams -Diagram Organization- Diagram Extensions. Design Process- Design concepts: Abstraction, Architecture, patterns, Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Aspects, Refactoring, Object Oriented Design Concepts, Design Classes- Design Model: Data, Architectural, Interface, Component, Deployment Level Design Elements.

UNIT IV [10T]

Structured coding Techniques-Coding Styles - Standards and Guidelines-Documentation Guidelines-Modern Programming Language Features: Type checking-User defined data types-Data Abstraction Exception Handling - Concurrency Mechanism.

UNIT V [12T]

TESTING: Software Quality- Software Quality Dilemma- Achieving Software Quality- Testing: Strategic Approach to software Testing- Strategic Issues - Testing: Strategies for Conventional Software, Object oriented software, Web Apps-Validating Testing- System Testing- Art of Debugging.

MAINTENANCE: Software Maintenance-Software Supportability- Reengineering - Business Process Reengineering- Software Reengineering- Reverse Engineering - Restructuring- Forward Engineering- Economics of Reengineering

TEXT BOOKS

1. Roger S, *“Software Engineering – A Practitioner’s Approach”*, seventh edition, Pressman, 2010.
2. Pearson Education, *“Software Engineering by Ian Sommerville”*, 9th edition, 2010.
3. Roff: UML: A Beginner’s Guide TMH

REFERENCES

1. Hans Van Vliet, “Software Engineering: Principles and Practices”, 2008.
2. Richard Fairley, “Software Engineering Concepts”, 2008.
3. RohitKhurana, Software Engineering: Principles and Practices, 2nd Edition, Vikas Publishing House Pvt Ltd.
4. PankajJalote, An Integrated Approach to Software Engineering, 3rd Edition, Narosa Publishing House.
5. Alhir, learning UML, SPD/O’Reily

SEMESTER VI

BCS6B11| Android Programming

Course Number: 30
Contact Hours per Week: 5 (4T + 1L)
Number of Credits: 4
Number of Contact Hours: 75 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To have a review on concept of Android programming.
- To learn Android Programming Environments.
- To practice programming in Android.
- To learn GUI Application development in Android platform with XML

Prerequisites

- Knowledge in OO & Java Programming.

Course Outline

Unit I [12T+3L]

Introducing the android computing platform, History of android, an- droid softwarestack, Developing end user application using Android SDK, Android java packages,Setting up the development environment, Installing android development tools(ADT), Fundamental components, Android virtual devices, Running on realdevice, Structure of android application, Application life cycle.

Unit II [12T+3L]

Understanding android resources - String resources, Layout resources, Resourcereference syntax, Defining own resource IDs - Enumerating key android resources,string arrays, plurals, Colour resources, dimension resources, image resources,Understanding content providers - Android built in providers, exploring databaseson emulator, architecture of content providers, structure of android content URIs,reading data using URIs, using android cursor, working with where clause,inserting updates and deletes, implementing content, Understanding intents basicsof intents, available intents, exploring intent composition, Rules for ResolvingIntents to Their Components, ACTION PICK, GET CONTENT, pending intents

Unit III [12T+3L]

User interfaces development in android - building UI completely in code, UI usingXML, UI in XML with code, Android's common controls - Text controls, buttoncontrols, checkbox control, radio button controls, image view, date and timecontrols, map view control, understanding adapters, adapter views, list view, gridview, spinner control, gallery control, styles and themes, Understanding layoutmanagers - linear layout manager, table layout manager, relative layout manager,frame layout manager, grid layout manager.

Unit IV [12T+3L]

Android menus - creating menus, working with menu groups, responding tomenu items, icon menu, sub menu, context menu, dynamic menus, loading menuthrough XML, popup menus, Fragments in Android structure of fragment,fragment life cycle, fragment transaction and back stack, fragment manager, saving fragment state, persistence of fragments, communications with fragments,startActivity() and setTargetFragment(), using dialogs in android, dialogfragments, workingwith toast, Implementing action bar - tabbed navigation action bar activity, implementing base activity classes, tabbed action bar and tabbedlistener, debug text view layout, action bar and menu interaction, list navigationaction bar activity, spinner adapter, list listener, list action bar, standard navigationaction bar activity, action bar and search view, action bar and fragments.

Unit V [12T+3L]

Persisting data - Files, saving state and preferences - saving application data,creating, saving and retrieving shared preferences, preference framework andpreference activity, preference layout in XML, native preference controls,preference fragments, preference activity, persisting the application state, includingstatic files as resources, Working with file system, SQLLite - SQLLite types,database manipulation using SQLLite, SQL and database centric data model for Android, Android database classes.

References:

1. SatyaKomatineni& Dave MacLean,*Pro Android 4*, Apress.
2. Retomeier, *Professional Android 4 Application Development*, Wrox.
3. ZigurdMednieks, Laird Dornin, G. Blake Meike, and Masumi Nakamura, *Programming Android*, O'Reilly

BCS6B12| Operating Systems

Course Number: 31
Contact Hours per Week: 5 (4T + 1L)
Number of Credits: 4
Number of Contact Hours: 80 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objectives

- To learn objectives & functions of Operating Systems.
- To understand processes and its life cycle.
- To learn and understand various Memory and Scheduling Algorithms.
- To have an overall idea about the latest developments in Operating Systems

Prerequisites

- Knowledge in Data structures.

Course Outline

Unit I [12T + 3P]

Operating System - Objectives and functions – The Evolution of Operating Systems: Serial Processing, Simple batch Systems, Multi Programmed batch Systems, Time Sharing Systems, Parallel Systems, Distributed Systems, Real time systems. Definition of Process, Process States, Process Control Block, Operations on Process, Process Communication, Communication in Client server System, Basic concepts of threads, Concurrency, Principles of Concurrency, Mutual exclusion, Semaphores, Messages, Dead lock: Prevention, Detection, Avoidance.

Unit II [12 T + 3P]

Linux Shell Programming: Introduction – Shells available in Unix: Bourne shell (sh), C shell (csh), TC shell (tcsh), Korn shell (ksh), Bourne Again SHell (bash). Bash: special characters – getting help – man pages – Linux Directory Layout – Command for Navigating the Linux Filesystems: pwd, cd, ls, file, cat, cp, mv, mkdir, rmdir, whereis – Piping and Redirection - Informational Commands: ps, w, id, free – clear, echo, more. File permissions – Setting Permissions – Making a file executable. Creating shell programs: comments, variables, operators (arithmetic, relational, logical) – single and double quotes - read – echo – test - conditional commands, iterative commands – break – continue - evaluating expressions using expr, bc – strings – grep – arrays.

Unit III [12T + 3P]

CPU Scheduling: Scheduling Criteria, Scheduling algorithms: FCFS, SJF, Priority, RR, Multilevel, Feedback Queue - Process synchronization, The Critical Section Problem, Synchronization Hardware, Classical Problems of Synchronization: Reader Writer, Dining Philosopher. File and Database System, File System, Functions of organization, Allocation and Free Space Management.

Unit IV [12T + 3P]

Memory Management, Address Binding, Logical Vs Physical Address Space, Dynamic Loading, Dynamic Linking and Shared Libraries, Overlays, Swapping, Contiguous Memory allocation, Paging, Segmentation, Virtual memory, Demand Paging, Page Replacement, Thrashin.

Unit V [12 T + 3 P]

Protection and security: policy and mechanism, authentication, authorization. Mobile OS: Concepts, history, features, architecture, future scope. Case studies: Android, UNIX kernel and Microsoft Windows NT (concepts only).

Text Books

1. Silberschatz, Galvin and Gagne, Operating System Concepts, John Willey & Sons
2. William Stallings, Operating Systems, Internals and Design Principles, PHI
3. Mendel Cooper, Advanced Bash-Scripting Guide, Available at http://www.tldp.org/LDP/abs/abs-guide.pdf

References:

4. Nutt G.J, Operating Systems: A Modern Perspective, Addison Wesley

BCS6B13| Computer Networks

Course Number: 32
Contact Hours per Week: 5T
Number of Credits: 4
Number of Contact Hours: 75 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objectives

- To learn about transmissions in Computer Networks.
- To learn various Protocols used in Communication.
- To have a general idea on Network Administration.

Prerequisites

- Knowledge in data structure.
- Knowledge in Operating System.

Course Outline:

Unit I [15 T]

Introduction to Computer networks, Topology, categories of networks, Internetwork, Internet, Network Models, Layered model, OSI and TCP/IP models, Physical layer, Switching - Circuit switching, Packet Switching and Message Switching, DTE - DCE Interface, EIA - 232 interface, X.21 modems.

Unit II [15T]

Data link layer, Error detection and correction, Types of errors, Single CSC error and Burst error, Vertical redundancy check (VRC), longitudinal redundancy Check (LRC), Cyclic Redundancy Check(CRC), Error correction - Single CSC error correction, Hamming code Data compression - Huffman code, data link control, Line discipline, Flow control, Error control, Multiple Access, Random Access, ALOHA, pure ALOHA and slotted ALOHA, CSMA/CD and SCMA/CA, Polling, Wired LANs, Ethernet - IEEE standards, Wireless LANs - IEEE - 802.11,basics of Bluetooth,wifi,wimax and mobile networks (2G,3G,4G)

Unit III [15T]

Network layer, Networking and Internetworking devices - Repeaters, Bridges, Routers, Gateways, Logical addressing - IPv4 & IPv6 addresses, Network Address Translation(NAT), Internet protocols, internetworking, Datagram, Transition from IPv4 to IPv6, Address Mapping-Error reporting and multicasting - Delivery, Forwarding and Routing algorithms, Distance Vector Routing, Link State Routing,

Unit IV [15 T]

Transport layer, Process-to-process Delivery: UDP, TCP and SCTP, Congestion control and Quality of Service, Application Layer, Domain Name Systems-Remote Login-Email FTP, WWW, HTTP, Introductory concepts on Network management: SNMP.

Unit V [15T]

Cryptography and Network Security: Introduction – Goals of Security – Attacks - Services and Techniques. Basics of Cryptography: Plain Text - Cipher Text – Encryption – Decryption. Confidentiality: Basics of Symmetric Key Ciphers - Traditional Symmetric Key Ciphers: Substitution, Transposition, Stream & Lock, Modern – Components of Modern Block Cipher – DES - Modern Stream Cipher. Basics of Asymmetric Key Ciphers – RSA Cryptosystem. Integrity: Message – Message Digest – Hash Function. Authentication: MAC. Digital Signature : Analogy with Manual Signature – Process – Signing the Digest – Services – RSA Digital Signature Scheme.

Textbook:

1. Behurouz A Forozan, *Introduction to Data Communications & Networking*, TMH

References:

1. Andrew S. Tanenbaum, Computer Networks, PHI
2. William Stallings, Data and Computer Communications, VIIth Edition, Pearson Education
3. William Stallings, Cryptography and Network Security, Principles and Practices, Prentice Hall of India.
4. Steven Graham and Steve Shah, Linux Administration: A Beginners Guide, Third Edition, Dreamtech,2003.

BCS6B14| Programming Laboratory III: Lab Exam of Vth Semester Java & PHP Programming

Course Number: 33
Contact Hours per Week: 0
Number of Credits: 2
Number of Contact Hours: 0 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To practice Java programming.
- To practice client side and server side scripting.
- To practice PHP Programming.
- To practice developing dynamic websites.
- To practice how to interact with databases through PHP.

Prerequisites

- Theoretical knowledge in Java programming.
- Theoretical knowledge of PHP Programming.

Course Outline

Part A: Java Programming

1. Write a program to find the distance between two points.
2. Write a program to find the sum, difference, product, quotient and remainder of two numbers passed as command line argument.
3. Write java program to display Fibonacci series up to a limit.
4. Write java program to display armstrong numbers within a range.
5. Given the sides of a triangle, write a program to check whether the triangle is equilateral, isosceles or scalene and find its area.
6. Read an array of 10 or more numbers and write a program to find the
   a) Smallest element in the array
   b) Largest element in the array
   c) Second largest element in the array
7. Write a program to perform base conversion
   a) Integer to binary
   b) Integer to Octal
   c) Integer to Hexadecimal
8. Write a program to verify De Morgan’s Law
9. Write a program to merge two arrays.

10. Write a program to find the trace and transpose of a matrix.
11. Write java program to find the sum of the digits and reverse of a given number using class and objects.
12. Write a program to sort a set of n numbers using a class.
13. Create a class ‘Account’ to represent a bank account. Write a program to deposit and withdraw amounts from the account.
14. Using class and objects, Write a java program to find the sum of two complex numbers (Hint: Use object as parameter to function).
15. Create a class Time with hh, mm, ss as data members. Write a java program to find the sum of two time intervals (Hint: Use object as parameter to function).
16. Write a program to count and display total number of objects created to a class (Hint: static members).
17. Write a java program to find the volume of cube, rectangular box, cylinder using function overloading.
18. Create a class student with methods to read and display the student details. Create a derived class result with methods to read marks of 5 subjects. Write a java program to display the total and grade of students, creating objects of class result.
19. Create a class Employee with ID, Name Designation and Dept. Create a child class salary with Basic, HRA, DA and Allowance. Write a program to compute the net salary assuming that HRA is 1250, DA, Allowance are 110% and 35% of the Basic salary.
20. Write a program to demonstrate inheritance hierarchy by using class a base class shape and 'TwoDim' and 'ThreeDim' as sub classes. Create classes ‘square’ and 'triangle' derived from TwoDim and 'sphere and 'cube' derived from ThreeDim. A reference variable of shape is used to determine area of various shapes.
21. Write a program to demonstrate the order in which constructors are invoked in multilevel inheritance.
22. Create an abstract class shape with two data members and an abstract method area. Create two child classes rectangle and triangle. Write a program to display the area of the shapes.
23. Create an interface calculator having methods to perform basic arithmetic operation. Write a program to implement the interface to perform operation on integer and float values.
24. Create a class factorial with a method that accept a number and return its factorial in a package P1. Using the factorial class, write a program to find the factorial of a number.
25. Write a multi thread java program for displaying odd numbers and even numbers up to a limit (Hint : Implement thread using Runnable interface).
26. Write a multi thread java program for displaying numbers ascending and descending order (Hint: create thread by inheriting Thread class).
27. Write a program to handle arithmetic exception.
28. Create a user defined exception ‘MinBalExp’ to be invoked when the read number is

less than a pre-set value.

29. Create a user defined exception ‘OddValExp’ to be invoked when the read number is an odd number.
30. Write a program to copy a file to another. Pass the file names as command line arguments.
31. Write a program to track keyboard events on an applet.
32. Write an applet to display a rectangle with specified coordinate and colour passed as parameter from the HTML file.
33. Create an AWT application to add, remove items in a list box.
34. Create an AWT application to select gender using radio buttons.
35. Design a window to accept the qualifications of a user using checkboxes.
36. Create an applet for a displaying smiling face.
37. Write a program to display ip address of the system.
38. Write a program to implement echo server (A server that echo the messages the client sends).
39. Create a database table employee (id, name, design, dept). Write a program to list the employees using JDBC.
40. Write a program to insert a new employee record to the above table.

Part B: PHP Programming

HTML & CSS

1. Design a website of an educational institution using framesets and links. A sample design is as shown below.

Top Frame	
Menu frame	Details frame

2. Design a webpage that illustrates the use of the following form controls: (i) input controls: single-line text, password, multi-line text. (ii) buttons: submit and reset.
3. Design a webpage that illustrates the use of the following form controls: (i) input controls: check box, radio button, select box (ii) buttons: submit and reset.
4. Design a webpage that illustrates the use of the following form controls: (i) input controls: datalist, multi-select box, grouped select box (ii) buttons: submit and reset.
5. Design a webpage that illustrates the use of field sets and legends.
6. Design a web page to demonstrate Border colors using internal CSS.
7. Design a web page to demonstrate Text alignment using CSS.
8. Design a web page to demonstrate inline CSS.
9. Design a webpage to invert the behavior of the <h1> to <h6> tags using external CSS.

10. Design a webpage for a simple image gallery.

JavaScript

11. Write a javascript program to perform find the area and circumference of a circle.
12. Write a javascript program to check whether a given number is perfect, abundant or deficient. Use alert box to display the output.
13. Write a javascript program to check whether the given sides can form a triangle. If yes, find the type (isosceles, equilateral and scalene) and area of the triangle. Use prompt dialogue box to accept the sides.
14. Write a javascript program to display the nth prime number. Value of n should be accepted from the user. Validate the value entered by the user: Only positive numbers except 0 are to be accepted.
15. Write a JavaScript program to find all years in which 1st January is a Sunday between a given range (eg:- between 2010 and 2017). Use
16. Design a webpage to illustrate image rollover.
17. Design a JavaScript program to illustrate the following events: onLoad, onClick, onBlur,onSubmit, onChange.
18. Design a JavaScript program to display the multiplication table of a no accepted from the user.
19. Design a form that accepts two integers. Provide 4 buttons for Add, Subtract, Multiply, Divide. Add JavaScript program to add, subtract, multiply and divide the given numbers when these buttons are clicked. Use output element to display the results.
20. Write a JavaScript program to create a table after accepting row and column numbers from the user. Contents of each cell should be corresponding row-column number (e.g. Row-0 Column-0).
21. Write a JavaScript program to store different colors in an array and change the background color of the page using this array elements
22. Write a JavaScript program to create clock with a timing event.
23. Write a JavaScript program for form validation for question numbers 2, 3 and 4.
24. Design a webpage to demonstrate the use of **progress** HTML element.

PHP

25. Write a PHP program to check whether the given number is Armstrong or not.
26. Write a PHP program to check whether a given number is perfect, abundant or deficient.
27. Display the Fibonacci series up to a given number.
28. Create a php program to display the bio data of a person by reading the personal details using an HTML page.
29. Create a login page using database.

30. Create a mysql table student with fields roll no, name, mark, grade. Insert records in the table. Write a PHP program to display the mark list of a student by accepting the register no of the student.
31. Write a php application to generate the pay slip of an employee by accepting name, basic salary and designation. The net salary will be calculated based on the following conditions.

Designation	**conveyance allowance**	**extra allowance**
Manager	1000	500
Supervisor	750	200
Clerk	500	100
Peon	250	

HRA – 25 %

Income tax

Gross <=200	0
2500 < gross <=4000	3%
4000< gross <=5000	5%
Gross >5000	8%

Gross= basic + HRA + conveyance + extra

Net = gross – income tax

32. Create a table “product” with fields itemcode, itemname, unitprice. Write php program to insert 5 records into the table and display it in a table format.
33. Write a php program for delete and update operation on account table. The account table contain fields such as accountno, name and amount.
34. Write an HTML page to display a list of fruits in a list box. Write php program to display the names of the fruits which are selected by the user.
35. Write php program to store current date/time in a cookie and display the ‘last visited on‘ date time on the web page upon reopening of the same page.
36. Design a PHP page to implement a login screen using sessions. Login details are to be verified from the server side with values stored in a database.
37. Write a php program to create an array and store 10 names in the array. Do the following operations.
    a. Display the contents using for each statement.
    b. Display the array in a sorted order.

c. Display the array without the duplicate elements
d. Remove the last element and display
e. Display the array in reverse order
f. Search an element in the given array.

38. Design a PHP page to illustrate the use of **keygen** HTML element.
39. Design a PHP page to illustrate the use of **meter** HTML element – accept five cities and the temperature of those cities and display the result graphically.
40. Design a PHP page to illustrate the use of file upload – uploading files of a type with a specified size to the webserver.

Include any 20 Java Programming Lab questions and 20 PHP Programming Lab questions in the record book.

<u>Android versions eclipse, studio etc. may be used for doing the lab works.</u>

Note: All lab works should be neatly recorded in a Laboratory Record Book in written form. However Program results can be pasted in the left hand side of the fare record. All students should have a rough record (observation note book) too, in which they write all the works to be carried out in the lab prior to his/her entering the lab. He/She may also note down the i/p and o/p that he gives for program verification in the observation note book (rough record).

BCS6B15| Programming Laboratory IV: Lab Exam of Android & Linux shell Programming

Course Number: 34
Contact Hours per Week: 4L
Number of Credits: 2
Number of Contact Hours: 60 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To practice Android programming.
- To practice user interface applications.
- To develop mobile application.
- To practice shell programming

Prerequisites:

- Theoretical knowledge in Android programming.
- Theoretical knowledge of Shell Programming.

Course Outline:

Part A: Android Programming

1. Programs to understand basic arithmetic operations
2. Programs to understand basic logic operations
3. Programs to understand loops and control statements
4. Programs to understand GUI in android
5. Android application for adding two numbers
6. Develop simple user interface to display message
7. Create two menu items-opening a file-saving a file
8. Inserting values into Spinner control using Text view and Button.
9. Implementation of background image
10. Starting another activity from your own activity using intent
11. Create a new activity that services ACTION-PICK for contact data which display each of the contact in the contact database and lets the user to select one before closing and returning the selected contacts URL to the calling activities
12. Create Android application to linkify a text view to display web and E-mail address as hyperlinks. When clicked they will open the browser and E-mail address respectively
13. Implementation of array adapter

14. Create an alert dialogs used to display a message and offer two button options to continue. Clicking either button will close the dialog after executing the attached click listener
15. Get data from Text view control and insert into database using SQLite. Another activity shows inserted data in a List View control
16. Load menu item by parsing XML data.
17. Program to implement simple calculator
18. Program to Get IP Address
19. Program to Home and Lock Screen Widget (Temperature Widget)
20. Create a new contact using intent
21. A Button control shows Date picker and Text view control displays selected date.
22. Insert data into Spinner and delete selected item using SQLite.
23. Program to create simple login screen.
24. Create an Android application to display the map of your locality. Use ACTION_VIEW intent by passing latitude and longitude as parameters.
25. Create an Android application to convert a voice into text (using Google Speech to Text service)
26. Create an Android application to populate a list view by getting names & numbers from a SQLite database table.
27. Display the saved contacts available in the android phone in a listview using content providers
28. Create an image grid. Images should be placed under the resources section.
29. Create an android app with three tabs. First tab should contain two Edit text and that should accept age and name. In the second tab you need another 3 edit texts that accept education address and phone number. After these information is proved, when the third tab is open it should display all the provided information neatly.
30. Create a custom toast with an Image and a TextView.
31. Apply a Custom List style to a ListView. ListView should have at least 10 Items.
32. Determine the acceleration of your android device along all three axes using accelerometer? (i.e. x,y,z).
33. Capture an Image from the primary camera of an android device and save that picture into the internal storage.
34. Create an app to list files under a given folder name in an EditText
35. Fetch data from an arbitrary URL given in an EditText and display it in a TextView
36. Create an SQLite database named student. Accept student details from the MainActivity and save it in the table called student. Display the calculated result in the second activity when a button on the MainActivity is clicked.
37. Create an android app to switch the wifi on and off also illustrate the use of permission in android?
38. Create a spinner that takes data from the String.xml file.

39. Create a simple android application that opens the default messaging application available in the android device?
40. Create an app to display message in the notification bar?

Part B: Shell Programming

1. Write a script to find area of a circle
2. Write a shell script to find given number is even or odd
3. Write a shell script to make a menu driven calculator using case
4. Write a shell script to find the greatest of three numbers
5. Write a shell script to compute mean and standard deviation of three numbers
6. Write a shell script to find sum of all digits from a given number
7. Write a shell script to find reverse of a number
8. Write a shell script to find prime numbers upto a given number
9. Write a shell script to find n fibonacci numbers
10. Write a shell script to check whether a given number is Amstrong or not
11. Write a shell script to reverse a string and check whether a given string is palindrom or not
12. Write a shell script to count no of line,words and characters of a input file
13. Code for Write a shell program to convert all the contents into the uppercase in a particular file in Unix
14. Write a script to find the value of one number raised to the power of another. Two numbers are entered through the keyboard.
15. Write a shell script find the factorial of a given number
16. An employee Basic salary is input through keyboard where da is 40% of basic salary and hra is 20% of basic salary. Write a program to calculate gross salary
17. Write a shell script to find the average of the number entered as command line arguments
18. Code for Shell script which whenever gets executed displays the message â€œGood Morning/Good afternoon /Good Evening â€œdepending on the time it get executed"
19. Write a shell script to Display Banner, calander of given year
20. Code for a program to display current date and time, number of users , terminal name, login date and time
21. Write a shell script which uses all the file test operators
22. Write a shell script to copy the contents of file to another. Input file names through command line. The copy should not be allowed if second file exists.
23. Write a shell script to find number of vowels, consonants, numbers in a given string.
24. Code for Shell script to perform operations like display, list, make directory and copy, rename, delete
25. Write a shell script to compare two files and remove one of them if they are same

Include any 15 Android Programming Lab questions and 15 Shell Programming Lab questions in the record book.

Note: All lab works should be neatly recorded in a Laboratory Record Book in written form. However Program results can be pasted in the left hand side of the fare record. All students should have a rough record (observation note book) too, in which they write all the works to be carried out in the lab prior to his/her entering the lab. He/She may also note down the i/p and o/p that he gives for program verification in the observation note book (rough record).

BCS6B17| Project Work

Course Number: 36
Contact Hours per Week: 4 (0T + 2L in V Sem + 2L in VI Sem)
Number of Credits: 3
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal – 10 Marks + External – 40 Marks

Objective

To provide practical knowledge on software development process

Prerequisites

Basic programming and system development knowledge

Course Outline

The objective of the B.Sc. Computer Science final project work is to develop a quality software solution by following the software engineering principles and practices. During the development of the project the students should involve in all the stages of the software development life cycle (SDLC). The main objective of this project course is to provide learners a platform to demonstrate their practical and theoretical skills gained during five semesters of study in B.Sc. Computer Science Programme.

During project development students are expected to define a project problem, do requirements analysis, systems design, software development, apply testing strategies and do documentation with an overall emphasis on the development of a robust, efficient and reliable software systems. The project development process has to be consistent and should follow standard. For example database tables designed in the system should match with the E-R Diagram. SRS documents to be created as per IEEE standards.

Students are encouraged to work on a project preferably on a live software project sponsored by industry or any research organization. Topics selected should be complex and large enough to justify as a B.Sc. Computer Science final semester project. The courses studied by the students during the B.Sc. Computer Science

Programme provide them the comprehensive background knowledge on diverse subject areas in computer science such as computer programming, data structure, DBMS, Computer Organization, Software Engineering, Computer Networks, etc., which will be helping students in doing project work. Students can also undertake group project to learn how to work in groups. However, the maximum number of students in a group must be limited to 4.

For internal evaluation, the progress of the student shall be systematically assessed through *two or three stages of evaluation at periodic intervals*.

A bonafied project report shall be submitted in hard bound complete in all aspects.

Open Courses (XXX5DXX)

BCS5D01| Introduction to Computers & Office Automation

Course Number: 29
Contact Hours per Week: 2T
Number of Credits: 2
Number of Contact Hours: 30 Hrs.
Course Evaluation: Internal – 10 Marks + External – 40 Marks

Objective

- To learn Office Automation.

Prerequisites

- Basic knowledge in Computer & Internet.

Course Outline

Unit I [7T]

Introduction to Computers: Types of Computers - DeskTop, Laptop, Notebook and Netbook. Hardware: CPU, Input / Output Devices, Storage Devices – System - Software - Operating Systems, Programming Languages, Application Software - Networks - LAN, WAN - Client - Server.

Unit II [7T]

Documentation Using a Word Processor (OpenOffice Writer / M.S. Word) - Introduction to Office Automation, Creating & Editing Document, Formatting Document, Auto-text, Autocorrect, Spelling and Grammar Tool, Document Dictionary, Page Formatting, Bookmark, Advance Features - Mail Merge, Macros, Tables, File Management, Printing, Styles, linking and embedding object, Template.

Unit III [8T]

Electronic Spread Sheet (Open Office Calc/MS-Excel) - Introduction to Spread Sheet, Creating & Editing Worksheet, Formatting and Essential Operations, Formulas and Functions, Charts, Advanced features – Pivot table & Pivot Chart, Linking and Consolidation.

Unit IV [8T]

Presentation using (OpenOffice Impress/MS-Power Point): Presentations, Creating, Manipulating & Enhancing Slides, Organizational Charts, Charts, Word Art, Layering art

Objects, Animations and Sounds, Inserting Animated Pictures or Accessing through Object, Inserting Recorded Sound Effect or In-Built Sound Effect.

References:

1. Michael Miller, *Absolute Beginner's Guide to Computer Basics*, Prentice Hall.
2. Russell A. Stultz, *Learn Microsoft Office*, BPB Publication.
3. H.M.Deitel, P. J. Deitel, et al., *Internet & World Wide Web - How to program*, Prentice Hall.

BCS5D02| Web Designing

Course Number: 29
Contact Hours per Week: 2T
Number of Credits: 2
Number of Contact Hours: 30 Hrs.
Course Evaluation: Internal – 10 Marks + External – 40 Marks

Objective

- To learn Web designing.

Prerequisites

- Basic knowledge in Computer & Internet.

Course Outline

Unit I [7T]

HTML: Introduction - history of html, sgml - structure of html document, web page layout, html tags and types - font type, paragraph formatting, meta data, blockquote, hyperlinks, linking, comments, white space, horizontal ruler, images, ordered and unordered lists, frames, tables, forms

Unit II [7T]

DHTML: Introduction, DHTML technologies, elements of DHTML, document object model, events - window events, form events, keyboard events, mouse events, style sheets, properties used in style sheets - background properties, positioning properties.

Unit III [8T]

Javascript: Introduction and advantages of javascript, java script syntax, writing javascript in html, javascript operators, arrays and expressions, programming constructs - for .. in loop, while loop - dialog boxes and prompts - alert, prompt, confirm methods - functions - built-in functions and userdefined functions, scope of variables, handling events, using event handlers and event methods, form object, properties, methods, form element’s properties and methods.

Unit IV [8T]

HTML Editor: Introduction, advantages, creating, opening, saving a web page, building forms, formatting and aligning text and paragraph, adding lists, styles and themes, linking pages, working with images, frames.

Reference:

1. H. M. Dietel, *Internet and World Wide Web*, Pearson.

BCS5D03| Introduction to Problem Solving and C Programming

Course Number: 29

Contact Hours per Week: 2T

Number of Credits: 2

Number of Contact Hours: 30 Hrs.

Course Evaluation: Internal – 10 Marks + External – 40 Marks

Objective

- To introduce fundamental principles of Problem Solving aspects.
- To learn the concept of programming.
- To learn C language.

Prerequisites

- Background of the basic science at +2 level

Course Outline

Unit I [7T]

Introduction: The problem solving aspect, Top-down design, Implementation of algorithms, Program verification, efficiency of algorithms. Introduction to C Programming, overview and importance of C, C Program Structure and Simple programs, Creation and Compilation of C Programs under Linux and Windows Platforms.

Unit II [7T]

Elements of C Language and Program constructs, Character Set, C Tokens, Keywords and Identifier, Constants, Variables, Data types, Variable declaration and assignment of values, Symbolic constant definition. C-Operators, Arithmetic operators, relational operators, and logical operators, assignment operators, increment and decrement operators, conditional operators, special operators, arithmetic expressions, evaluation of expressions, precedence of arithmetic operators, Type conversion in expressions, operator precedence and associativity, Mathematical Functions, I/O operations.

Unit III [8T]

Decision making, Branching and Looping, Decision making with IF statement, Simple IF statement, If.. .else statement, Nesting of If..else and else...if Ladder, Switch statement, Conditional operator, Go-to statement. Looping: While loop, Do-While, and For Loops, Nesting of loops, jumps in loop, skipping of loops.

Unit IV [8T]

Array & Strings - One dimensional array, two dimensional array and multidimensional array, strings and string manipulation functions. Structures & Union structure definition , giving values to members, structure initialization, comparison of structure variables, arrays of structures, arrays within structures, structures within arrays, structures and functions, Unions, CSC-fields.

References:

1. Balaguruswami, *Programming in ANSI C*
2. Brian W. Kernighan & Dennis M. Ritchie, *The C Programming Language*
3. Yashvant P. Kanetkar, *Let Us C*
4. ByranGotfried, *Programming with C,* Schaums Outline Series

BCS5D04| Introduction to Data Analysis using Spread sheet

Course Number: 29
Contact Hours per Week: 2T
Number of Credits: 2
Number of Contact Hours: 30 Hrs.
Course Evaluation: Internal – 10 Marks + External – 40 Marks

Objective

- To introduce the importance of software tools.
- To learn the Analysisusing Spread sheets.

Prerequisites

- Background of the Basic Science andstatistics at +2 level

Course Outline

Unit I [8T]

Introduction to MS Excel and Understanding Basic Working with it: Quick review on MS Excel Options, Ribbon, Sheets , Difference between Excel 2003, 2007, 2010 and 2013- Saving Excel File as PDF, CSV and Older versions - Using Excel Shortcuts - Copy, Cut, Paste, Hide, Unhide, and Link the Data in Rows, Columns and Sheet Using Paste Special Options - Formatting Cells, Rows, Columns and Sheets - Protecting & Unprotecting Cells, Rows, Columns and Sheets with or without Password - Page Layout and Printer Properties - Inserting Pictures and other objects in Worksheets

Unit II [7T]

Introduction to Pivot table: Use multiple pivot tables and pivot charts to create dashboard, Connect multiple slicers to the pivot tables.

Unit III [8T]

Pivot table applications in analytics: filter the data shown in the pivot in different ways to achieve subsets of the data, Use calculated fields on top of the pivot table to calculate profitability and find anomalies.

Unit IV [7T]

Formulae and Function: Use formulas to aggregate the data as an alternative to pivot tables for more flexible reporting layouts. Usage of multiple tables in a single pivot, introduction to data table.

Text Books:

1. Winston, *Microsoft Excel 2013: Data Analysis and Business Modeling*, Prentice Hall India Learning Private Limited (2013), ISBN: 9788120349605

References:

1. John Walkenbach, Microsoft Excel 2013 Bible, Wiley (23 April 2013), ISBN: 9788126541720.
2. Paul McFedries, Excel 2013 Formulas and Function 1st Edition, Pearson India (2014), ISBN: ISBN: 9789332524026.

Electives

BCS6B16a | System Software

Course Number: 35
Contact Hours per Week: 4T
Number of Credits: 3
Number of Contact Hours: 60 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To build fundamental knowledge in system software.
- To learn functions of various system software.
- To learn specifically learn compilation process of a program.

Prerequisites

- Basic knowledge in Programming.

Course Outline:

Unit I [12T]

System software: General concept, Assemblers, loaders, linkers, macros, compilers, interpreters, operating system, Design of assemblers.

Unit II [12T]

Macros and macro processors, Macro definitions and instructions, Macro calls, Features of Macros, Design of Macro processors.

Unit III [12T]

Loading, linking and relocating Loader schemes- Binders, linking loaders, overlays, dynamic binders- Dynamic loading and dynamic linking – Relocatability of programs.

Unit IV [12T]

Compilers - Phases of a compiler - Lexical, Syntax, Intermediate code generation, Optimization, Code generation, Symbol table and error correcting routines – Passes of a compiler.

Unit V [12T]

Case studies of lexical and syntax analyzers: LEX and YAAC.

References:

1. D.M. Dhamdhere, *Systems Programming and Operating Systems*
2. John J Donovan, Systems programming
3. Jim Welsh and R M Mckeag, Structured System Programming, Prentice Hall.
4. Principal of Compiler Design, Alfred Aho V and Jeffrey D Ullman,Addison- Wesley Publi.
5. L Lbech, System Software

BCS6B16b| Microprocessor and Applications

Course Number: 35
Contact Hours per Week: 4T
Number of Credits: 3
Number of Contact Hours: 60 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To understand internals of Microprocessor.
- To learn architecture of 8086 Microprocessor
- To learn instruction set of 8086 Microprocessor
- To learn how to program a Microprocessor

Prerequisites

- Basic knowledge of Computer.

Course Outline

Unit I [12 T]

16-Bit Microprocessor, 8086, Architecture, Pin Configuration, 8086 Minimum and Maximum mode configurations,

Unit II [12 T]

Addressing modes, 8086 Instruction set (Data transfer, Arithmetic, Branch, Processor control & String instruction), 8086 interrupts.

Unit III [12 T]

Assembler Directives Data Definition And Storage Allocation, Program Organization, Alignment, Program End value, Returning Attribute, Procedure Definition, Macro Definition, Data Control, Branch Displacement, Header File Inclusion, Target Machine Code Generation, Control Directives.

UnitIV [12 T]

Peripherals and Interfacing, Interfacing output displays (8212), interfacing input keyboards, key Debounce, Programmable communication interface (8251A), programmable peripheral interface

(8255), Programmable DMA Controller (8257), Programmable interrupt controller (8259), Programmable interval timer (8253).

UnitV [12 T]

Advanced Microprocessors Introduction to 80186, 80286, 80386, 80486 and Pentium processors, General introduction to BIOS and DOS interrupts.

Text Books

1. K.R. Venugopal, Raj Kumar,Microprocessor, X86 programming, BPB publications, New Delhi
2. Yu. Cheng Liu, Glenn A Gibson, Microcomputer Systems The 8086/8088 Family. Architecture, Programming & Designing, Prentice Hall of India Pvt. Ltd., New Delhi

References

3. Barry.B.Brey, The Intel Microprocessor 8086/8088. 80186, 80286, 80386 and 80486 Architecture Programming and Interfacing, Prentice Hall of India Pvt.Ltd.1995.
4. Abel P, IBM PC Assembly Language & Programming, Pearson Education Asia
5. B Ram, Fundamentals of Microprocessors and Microcomputers,DhanpatRai Publications Pvt. Ltd., New Delhi
6. MohamadRafiquzzaman, Microprocessors and Microcomputer Based System Designing, Universal Bookstall, New Delhi
7. Ray A.K.Bhurchandi.K.M, Advanced Microprocessor and Peripherals, Tata McGrawHill.

BCS6B16c | Discrete Structures

Course Number: 35
Contact Hours per Week: 4T
Number of Credits: 3
Number of Contact Hours: 60 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Course Outline

Unit I [12 T]

Prepositional Calculus - Propositions, Logical Connectives, Disjunction, Conjunction, Negation, Conditional Connectives, Precedence Rule, Logical Equivalence, Logical Quantifiers. Methods of Proof - What is a Proof? Different Methods of Proof, Direct Proof, Indirect Proofs, Counter Examples, Principle of Induction. Boolean Algebra and Circuits – Boolean Algebras, Logic Circuits, Boolean Functions.

Unit II [12T]

Sets, Relations and Functions: Introducing Sets, Operations on Sets, Basic Operations, Properties Common to Logic and Sets Relations, Cartesian Product, Relations and their types, Properties of Relations, Functions, Functions, Operations on Functions.

Unit III [12T]

Combinatorics - An Introduction, Multiplication and Addition Principles, Permutations, Permutations of Objects not Necessarily Distinct, Circular Permutations, Combinations, Binomial Coefficients, Combinatorial Probability

Unit IV [12T]

Pigeonhole Principle, Inclusion-Exclusion Principle, Applications of Inclusion - Exclusion, Application to Surjective Functions, Application to Probability, Application to Derangements.

Unit V [12 T]

Partitions and Distributions Integer Partitions, Distributions, Distinguishable Objects into Distinguishable Containers, Distinguishable Objects into Indistinguishable Containers, Indistinguishable Objects into Distinguishable Containers, Indistinguishable Objects into Indistinguishable Containers.

References

1. Gerald J. Alred ,Charles T. Brusaw, Walter E. Oliu, *Handbook of Technical Writing*, Tenth Edition.
2. Gary Blake and Robert W. Bly, The Elements of Technical Writing, New York: Macmillan Publishers.
3. Hackos, JoAnn T., *Managing Your Documentation Projects*. Wiley, 1994.
4. R.L. Adams, *SEO Black Book - A Guide to the Search Engine Optimization Industry's Secrets* (The SEO Series 1).
5. Danny Dover, Erik Dafforn, *Search Engine Optimization (SEO) Secrets*.

BCS6B16d | Computer Graphics

Course Number: 35
Contact Hours per Week: 4T
Number of Credits: 3
Number of Contact Hours: 60 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objective

- To learn basics of Computer Graphics.

Course Outline

Unit I [12 T]

Introduction to Computer Graphics Definition, Application, Pixel, Frame Buffer, Raster and Random Scan display, Display devices CRT, Color CRT Monitors, basics of LCD & LED Monitors.

Unit II[12 T]

Scan Conversion of line DDA algorithm of line drawing, Scan conversion of circles Bresenham's circle generating algorithm, Polygon, FillingScan line polygon filling algorithm.

Unit III [12 T]

Two Dimensional transformation, Translation, Rotation, Scaling, Homogeneous Coordinates, Reflection, Shear.

Unit IV [12 T]

Window to view port transformation, clipping, line clipping, Cohen Sutherland line clipping, Polygon clipping, Sutherland and Gary Hodgman polygon clipping algorithm.

Module V [12 T]

Color Models &Color Applications Light and Color, Different color models, RGB, CMY, YIQ. Introduction to GIMP Image Manipulation using GIMP.

Textbook:

1. Donald Hearn and M. Pauline Baker, Computer Graphics, PHI, New Delhi.

References:

1. Zhigang Xiang and Roy Plasock , Computer Graphics, Schaum's Outlines.
2. Deborah Morley, Understanding Computers Today and Tomorrow, Introductory Edition.

UNIVERSITY OF CALICUT

നിർമ്മായ കർമ്മണാ ശ്രീ:

BCS6B16e| Technical Writing

Course Number: 35
Contact Hours per Week: 4T
Number of Credits: 3
Number of Contact Hours: 60 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objectives

- Acquire the skills and knowledge for professional technical communication, web content writing, soft skill development and search engine optimization.

Prerequisites

- Basic communication skills in English.

Course Outline

Unit I [12 T]

Basics of Technical Communication: Technical Communication - features; Distinction between General and Technical communication; Language as a tool of communication; Levels of communication: Interpersonal, Organizational, Mass communication; the flow of Communication: Downward, Upward, Lateral or Horizontal (Peer group); Barriers to Communication.

Unit II [12 T]

Constituents of Technical Written Communication: Word formation, Prefix and Suffix; Synonyms and Antonyms; Homophones; One Word Substitution; Technical Terms; Paragraph Development: Techniques and Methods -Inductive, Deductive, Spatial, Linear, Chronological etc; The Art of Condensation- various steps.

Unit III [12 T]

Forms of Technical Communication - Business Letters: Sales and Credit letters; Letter of Enquiry; Letter of Quotation, Order, Claim and Adjustment Letters; Memos, Notices, Circulars; Job application and Resumes.

Unit IV [12 T]

Reports: Types; Significance; Structure, Style & Writing of Reports. Technical Proposal; Parts;

Types; Writing of Proposal; Significance, Technical Paper, Project, Dissertation and Thesis Writing. E-Media: E-mail – E-Newsletter – Blogging – E-Magzines – Social Networks.

Unit V [12 T]

Soft Skills: Presentation Strategies - Preparing a Presentation – Body Language – Voice Dynamics – Handling Questions.

Text Books

1. Professional Communication: For GautamBuddh Technical University &Mahamaya Technical University, Pearson Education India.
2. Phillip A. Laplante, Technical Writing: A Practical Guide for Engineers and Scientists (What Every Engineer Should Know), CRC Press.

References

1. Gerald J. Alred ,Charles T. Brusaw, Walter E. Oliu, Handbook of Technical Writing, Tenth Edition.
2. Gary Blake and Robert W. Bly, The Elements of Technical Writing, New York: Macmillan Publishers.
3. Hackos, JoAnn T., Managing Your Documentation Projects. Wiley, 1994.

BCS6B16f| Fundamentals of Life Skill Education

Course Number: 35
Contact Hours per Week: 4T
Number of Credits: 3
Number of Contact Hours: 60 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objectives

- Develop intra-personal, inter-personal, critical thinking, decision making and communication skills.
- Establish self management and help to maintain work life balance.
- Get an insight to career planning and development

Course Outline

Unit I[12 T]

Introduction to life skill education, definition, components, pillars of learning, need for life skill training, approaches - critical thinking skills/decision making skills, interpersonal/communication skills, criteria for using life skills.

Unit II [12 T]

Communication skills, communication, definition , components- sender, message, channel , receiver, feedback, types of communication, effective interpersonal communication, barriers, communication noise, listening, ways to improve interpersonal communication, effective public speaking interview, group discussion

Unit III [12 T]

Career planning, career planning steps, choosing a career, career development, career guidance and career guidance centre, need and importance of career guidance, career guidance centre and sources, making a career decision, preparing a resume and tips

Unit IV [12 T]

Self management, self esteem, definitions, practice self acceptance, practice self acceptance characteristics of people with high self-esteem, low self esteem, characteristics and causes, self-esteem building, self awareness importance, develop self awareness, self control, developing self control, emotional intelligence or emotional quotient, emotional quotient , two aspects of emotional intelligence, five domains of emotional eq or ei, social intelligence, coping with emotions, emotional intelligence,

Unit V [12 T]

Stress and strain: concept of stress, meaning and definition of stress, types of stress, major symptoms of stress, manage everyday stress. strain-mental strain, causes of strain, conflict, conflict resolution, understanding conflict in relationships, emotional awareness, managing and resolving conflict, stages of healthy conflict resolution, styles of conflict resolution, styles of dealing with conflict, developing positive thinking, positive and negative self-talk, better self-talk, impacts , assertiveness, behaviour , importance of assertive behaviour.

Text Books

1. ShaliniVerma, Development of Life Skills and Professional Practice , Vikas Publishing House; First edition (2014)

References:

1. Dr. K. RavikanthRao and Dr. P. Dinakar, Life Skills Education, Neelkamal; First edition (2016)
2. http://www.universityofcalicut.info/SDE/opencourses/Life_skill_education.pdf

UNIVERSITY OF CALICUT

THENHIPALAM, CALICUT UNIVERSITY P.O

SYLLABUS

FOR

COMPUTER SCIENCE (COMPLEMENTARY)

(CHOICE BASED CREDIT AND SEMESTER SYSTEM)

UNDER THE

FACULTY OF SCIENCE

FOR THE STUDENTS ADMITTED FROM THE ACADEMIC YEAR 2014 – 15 ONWARDS

BOARD OF STUDIES IN COMPUTER SCIENCE (UG)

THENHIPALAM, CALICUT UNIVERSITY P.O

KERALA, 673 635, INDIA
JUNE, 2017

REGULATIONS

FOR

COMPUTER SCIENCE - COMPLEMENTARY

(CHOICE BASED CREDIT AND SEMESTER SYSTEM)

EFFECTIVE FROM THE ACADEMIC YEAR 2017-18

COURSE EVALUATION

Total marks for each complementary course, including lab course, shall be 80 marks.

The evaluation scheme for each course shall contain two parts (1) Internal evaluation (2) external evaluation

16 marks shall be given to the internal evaluation. The remaining 64 marks shall be for the external evaluation.

INTERNAL EVALUATION

16 marks in each course, including lab, are for internal examinations.

The internal assessment shall be based on a predetermined transparent system involving written test, assignments, seminars and attendance in respect of theory courses and on test/record/viva/attendance in respect of lab courses.

Components with percentage of marks of Internal Evaluation of Theory Courses are:

Test paper	–	8 Marks
Attendance	–	4 Marks
Assignment/Seminar/Viva	–	4 Marks

Components with percentage of marks of Internal Evaluation of Lab Courses are:

Test paper	–	8 Marks
Attendance	–	4 Marks
Assignment/Lab involvement	–	4 Marks

Attendance of each course will be evaluated as below:

Above 90% attendance	–	4 Marks
85 to 89%	–	3 Marks

80 to 84%	–	2 Marks
76 to 79 %	–	1.5 Marks
75%	–	1 Marks

(If a fraction appears in the final internal marks awarded, nearest whole number is to be taken)

To ensure transparency of the evaluation process, the internal assessment marks awarded to the students in each course in a semester shall be notified on the notice board at least one week before the commencement of external examination. There shall not be any chance for improvement for internal marks. The course teacher(s) shall maintain the academic record of each student registered for the course, which shall be forwarded to the University by the college Principal after obtaining the signature of both course teacher and HOD.

The marks secured for internal examination only need be sent to university, by the colleges concerned.

EXTERNAL EVALUATION:

There shall be University examinations for each course at the end of each semester.

Practical examinations shall be conducted by the University at the end of fourth semester.

External evaluation carries 64 Marks, for each course.

External evaluation of even (2 and 4) semesters will be conducted in centralized valuation camps immediately after the examination. Answer scripts of odd semester (1 and 3) examination will be evaluated by home valuation. All question papers shall be set by the university.

The model of the question paper for external examination (theory courses) of 3 Hrs. duration shall be:

1. **Section A**: 9 compulsory objective type questions (MCQ/fill in the blank/ matching/one word/etc) of 1 mark each (**Total 9Marks**)
2. **Section B:** 5 compulsory short answer type questions of 2 Marks each (either a single question or can have subdivisions) **(Total 10 Marks)**
3. **Section C:** 5 short essay type questions of 5 Marks each, to be attempted from a set of 8 questions – at least one question from each unit (either a single question or can have subdivisions**) (Total 25 Marks)**
4. **Section D:** 2 long essay type questions of 10 Marks each, to be attempted from a set of 3 questions (either a single question or can have subdivisions) **(Total 20 Marks)**

The external examination in theory courses is to be conducted with question papers set by external experts. The evaluation of the answer scripts shall be done by examiners based on a well-defined scheme of valuation and answer keys shall be provided by the University. The external examination in practical courses shall be conducted by two examiners, one internal and an external, appointed by the University.

The model of the question paper for external examination (lab courses) of 3 Hrs. duration shall be:

1. Section A: One marked question of 25 Marks from Programming Lab Part A (C Programming) is to be attempted (Design - Algorithm/Flowchart/Interface: 5 Marks, Code: 10 Marks and Result: 10 Marks. **Total 25 Marks**)
2. Section B: One marked question of 25 Marks from Programming Lab Part B (VB.NET Programming) is to be attempted (Design - Algorithm/Flowchart/Interface: 5 Marks, Code: 10 Marks and Result: 10 Marks. **Total 25 Marks**)
3. Section C: Lab viva voce (**Total 10 Marks**)
4. Section D: Lab Record (**Total 4 Marks**)

REVALUATION:

In the new system of grading, revaluation is permissible. The prevailing rules for revaluation are applicable. Students can apply for photocopies of answer scripts of external examinations. Applications for photocopies/scrutiny/revaluation should be submitted within 10 days of publication of results. The fee for this shall be as decided by the university.

IMPROVEMENT COURSE

A maximum of two courses (Common, Core, Complementary or Open) can be improved in each semester. Improvement of a particular semester can be done only once. The student shall avail the improvement chance in the succeeding year after the successful completion of the semester concerned. The internal marks already obtained will be carried forward to determine the grades/marks in the improvement examination. If the candidate fails to appear for the improvement examination after registration, or if there is no change in the results of the improvement examination appeared, the marks/grades obtained in the first appearance will be retained.

Improvement and supplementary examinations cannot be done simultaneously.

EVALUATION AND GRADING

Mark system is followed instead of direct grading for each question (for both internal and external examinations). For each course in the semester letter grade, grade point and % of marks are introduced in 7- point indirect grading system. The grading on the basis of a total internal and external mark will be indicated for each course and for each semester and for the entire programme.

Indirect Grading System in 7 point scale is as below:

% of Marks	**Grade**	**Interpretation**	**Grade Point (G)**	**Range of Grade Points**	**Class**
90 and above	A+	Outstanding	6	5.5 to 6	First class with distinction
80 to below 90	A	Excellent	5	4.5 to 5.49	
70 to below 80	B	Very Good	4	3.5 to 4.49	First class
60 to below 70	C	Good	3	2.5 to 3.49	
50 to below 60	D	Satisfactory	2	1.5 to 2.49	Second class
40 to below 50	E	Pass/Adequate	1	0.5 to 1.49	Pass
Below 40	F	Failure	0	0 to 0.49	Fail

An aggregate of E grade with 40% marks (after external and internal put together) is required in each course for a pass and also for awarding a degree.

Appearance for Internal Assessment (IA) and End Semester Evaluation (ESE-external)) are compulsory and no grade shall be awarded to a candidate if she/he is absent for IA/ESE or both. For a pass in each course 40% marks or E grade is necessary

A student who fails to secure a minimum grade for a pass in a course is permitted to write the examination along with the next batch.

After the successful completion of a semester, Semester Grade Point Average (SGPA) of a student in that semester is calculated using the formula given below. For the successful completion of a semester, a student should pass all courses. However, a student is permitted to move to the next semester irrespective of SGPA obtained.

The Semester Grade Point Average can be calculated as

$$\text{SGPA} = \frac{\text{Sum of the Credit Points of all courses in a semester}}{\text{Total Credits in that Semester}}$$

i.e.

$$\text{SGPA} = \frac{\text{C1 x G1} + \text{C2 x G2} + \text{C3 x G3} \dots .}{\text{n}}$$

where G1, G2, … are grade points of different courses; C1, C2, ... are credits of different courses of the same semester and n is the total credits in that semester.

The Cumulative Grade Point Average (CGPA) of the student is calculated at the end of a programme. The CGPA of a student determines the overall academic level of the student in a programme and is the criterion for ranking the students. CGPA can be calculated by the following formula

The Cumulative Grade Point Average (CGPA) can be calculated as:

$$\text{CGPA} = \frac{\text{TOTAL CREDIT PONTS OBTAINED IN SIX SEMESTERS}}{\text{TOTAL CREDITS}}$$

ATTENDANCE

A student shall be permitted to appear for the semester examination, only if he/she secures not less than 75% attendance in each semester. Attendance shall be maintained by the concerned Department Condonation of shortage of attendance to a maximum of 9 days in a semester subject to a maximum of two times during the whole period of the degree programme may be granted by the University. Benefit of attendance may be granted to students who attend the approved activities of college/university with prior concurrence of the Head of the institution. Participation in such activities may be treated as presence in lieu of their absence on production of participation attendance certificate. It should be limited to a maximum of 9 days in a semester.

The condonation of shortage of attendance shall be granted according to the existing prescribed norms.

CURRICULUM FOR B.Sc. COMPUTER SCIENCE

(COMPLEMENTARY)

(2017-18 ACADEMIC YEAR ONWARDS – AS PER THE CUCBCSSUG 2014 REGULATIONS)

Total Courses: 5				Total Credits: 12						
Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credits
				Intern	Extern	Total	Theor	Lab	Total	
1	1	CSC1C01	Computer Fundamentals	16	64	80	2	2	4	3
II	2	CSC2C02	Fundamentals of System Software, Networks & DBMS	16	64	80	2	2	4	3
III	3	CSC3C03	Problem solving using C programming	16	64	80	3	2	5	2
IV	4	CSC4C04	Data Structures Using C Programming	16	64	80	3	2	5	2
IV	5	CSC4C05	Programming Lab: C & Data Structures	16	64	80	0	0	0	2
Total (5 Courses)						400				12

CSC1C01 – Computer Fundamentals

Semester:1
Course Number: 1
Contact Hours:2T+2L
Number of Credits:3
Number of Contact Hours:60
Course Evaluation: Internal – 16 Marks + External – 64 Marks

Aim of the Course:

To impart the students with fundamental principles and operations of various units of computer and to impart them with the basic skill in application packages.

Objectives of the Course:

To learn the basics of computer hardware units and how they work together
To acquire basic skill with office packages

Prerequisites:

Background of the basic science at +2 level

Course Outline

UNIT I (6T+6L)

Number systems- Non-positional number systems and positional number systems (Binary, Octal and Hexadecimal), Converting from one number system to another- decimal to a new base, converting to decimal from another bases, converting from base other than ten to base other than ten, short cut method for converting from binary to octal, octal to binary, binary to hexadecimal and hexadecimal to binary, Computer Codes (BCD, EBCDIC, ASCII) error detecting and correcting codes, parity bit, Hamming Code, computer arithmetic ,importance of binary, binary addition and subtraction.

UNIT II (6T+6L)

Boolean Algebra and Logic circuits- fundamental concepts of Boolean Algebra, postulates, Principle of duality, theorems of Boolean Algebra, Boolean functions, minimization, complement, canonicals forms, conversion between canonical forms. Logic Gates- AND, OR, NOT, NAND, NOR, XOR and XNOR, logic circuits, converting expression to logic circuit, universal NAND and NOR gates, Exclusive OR and equivalence functions, Design of Combinational circuits (Half Adder, Subtractor and Full Adder)

UNIT III (6T+6L)

Basic Computer Organization-Input Unit, Output Unit, Storage Unit (Direct, Sequential and Random Access), CPU organization, Control Unit (micro programmed and hardwired control), primary storage, memory hierarchy, storage locations and addresses, storage capacity, bit, byte, nibble, RAM, ROM, PROM and EPROM, cache memory, registers. Secondary storage devices (Magnetic tape, Hard disk and CD drive)

UNIT IV (6T+6L)

I/O devices - Input Devices-identification and its use, keyboard, pointing devices (mouse, touch pad and track ball), Video digitizer, remote control, joystick, magnetic stripes, scanner, digital camera, microphone, sensor, and MIDI instruments, Output Devices identification and its use, monitor, printer (laser, inkjet, dot-matrix), plotter, speaker, control devices (lights, buzzers, robotic arms, and motors)

UNIT V (6T+6L)

Planning a Computer program, purpose of program planning, algorithm, flowchart - symbols, sample flowcharts, advantages and limitations

References:

1. Pradeep K. Sinha and PritiSinha, Computer Fundamentals, BPB
2. Peter Nortorn, Introduction to Computer, TMH
3. Rajaraman, V, Fundamental of Computers, Prentice Hall India
4. B. Ram,Computer Fundamentals

Lab List

Word Processing

- Paragraph formatting
- Newspaper style Document
- Table creation
- Mail merge
- Page formatting & printing

Spreadsheet

- Worksheet entries, including formulas
- Formatting cells

- Chart creation
- Functions

Presentation Software

- Creating presentation
- Animations
- Sound
- Inserting picture

CSC3C02 – Fundamentals of System Software, Networks & DBMS

Semester: 2
Course Number: 2
Contact Hours: 2T+2L
Number of Credits: 3
Number of Contact Hours: 60
Course Evaluation: Internal – 16 Marks + External – 64 Marks

Aim of the Course:

To impart the students with the basic concepts of system software, Computer Networks and Database.

Objectives of the Course:

To learn the basic concepts of various system software
To learn the basics of Computer Networks
To learn the basics of Databases

Prerequisites:

Background of the basic science at +2 level

Course Outline

UNIT I [6 T+6L]

System software - classification of programming languages (Machine, assembly & High level), Characteristics and Comparison, language processors (Assembler, Interpreter and Compiler), Operating Systems- Functions, types of OS (batch, multiprogramming, time sharing, real time and distributed)

UNIT II [6 T+6L)

Computer networks- goals of networking, network topologies, types of networks (LAN, MAN and WAN), network model, OSI model- 7 layers, Internet Layer- 5 layers, Communication Media-Guided (Twisted Pair, Coaxial Cable and Fiber Optic) and Unguided (microwave, satellite)

UNIT III [6 T+6L]

Database Management Systems-definition, structure of Database, data models (Record based Data model, Network model: - Basic Components, Record types, data types, links, relationships,Hierarchical model and Relational model)

UNIT IV [6 T+6L]

Structured query language - Create, insert, select, update, delete, alter, drop commands

UNIT V [6 T+6L]

HTML-hypertext, hyper media, understanding basic HTML tools- HTML editor, web browser, General structure of HTML document, different types of elements-doc type, comment element, structural element, HTML tags and attributes: <html>, <body>, <head>,
<title>, <h1>,... ,<h6>, <br>, <table>, <img>, <hr>, adding links, background image to the body, creating lists.

References:

1. P. K Sinha, Fundamentals of Computers
2. D. M Dhamdhere,Operating System: A concept based Approach
3. Behrouz A Forouzan, Data Communication & Networking, MC Graw Hill
4. Joel Sklar,Principles of Web Page Design, Vikas Publications

Lab List

HTML

1. Simple HTML document creation
2. HTML document with tables

3. HTML document with various lists
4. HTML document with links to different parts of the same
5. documents and to separate documents

MySQL

1. Table creation
2. Data insertion and deletion
3. Data retrieval
4. Alteration of tables

CSC2C03 – Problem Solving Using C Programming

Semester: 3
Course Number: 3
Contact Hours: 3T+2L
Number of Credits: 2
Number of Contact Hours: 76 Hrs.
Course Evaluation: Internal – Internal – 16 Marks + External – 64 Marks

Aim of the Course:

To equip the students with the basic concepts of problem solving using computers.

Objectives of the Course:

To learn the concepts of programming.
To learn the C language

Prerequisites:

Background of the basic science at +2 level

Course Outline

UNIT I [9 T+6L]

Introduction to C- Structure of C program, Character Set, Keywords, Identifiers, Data Types, Qualifiers, Variables, Declarations, Symbolic Constants, Expressions, Statements, Different Types of Operators (Arithmetic, Logical, Relational & Equality, Unary and Conditional), Operator Precedence and Associativity, Library Functions, Comments, I/O functions-(Formatted scanf() &printf(), getchar (), putchar (), getche(), gets(), puts())

UNIT II [9 T+6L]

Control Statements- Selection Statements (if, if-else, else if ladder, switch), iteration (while, do while, for), jumping (goto, break, continue), Nested Control Statements

UNIT III [9 T+6L]

Structured Data types - Arrays (One dimensional and Two Dimensional), Character and String Functions, Structure (Definition,Processing-period Operator), Union

UNIT IV [9 T+6L]

User defined Functions - Advantages, Definition, Accessing functions, formal and Actual Parameters, Recursion, Storage Classes- Automatic, External, Static and Register Variable, Argument Passing Mechanism

UNIT V [10T+6L]

Pointers and data files- Pointers, advantages, declaration, operations on pointers, pointers and one dimensional arrays, dynamic memory allocation. Data files (sequential), file handling functions (fopen(), fclose(), fputc(), fgetc(), fgets(), fputs(), fscanf(), fprintf()

References:

1. E Balagurusamy, *Programming in Ansi C*, Tata McGraw Hill
2. ByranGotfried, *Programming with C*, Schaum Series
3. Kezningham& Ritchie,*Programming in C*
4. YashvantKanetkar, *Let us C*, BPB publications
5. Mullish Cooper, *The spirit of C*, Jasco books
6. Herbert Schildt, *The Complete reference C*, Tata McGraw Hill

CSC4C04 – Data Structure Using C

Semester: **4**

CourseNumber: 4

ContactHoursperWeek:3(3T+2L)

NumberofCredits:2

NumberofContactHours:76

CourseEvaluation: Internal – 16 Marks+External – 64 Marks

Objective

Tointroducetheconceptofdatastructures
Tomakethestudentsawareofvariousdatastructures
Toequipthestudentsimplementfundamentaldatastructures

Prerequisites

KnowledgeinCProgrammingLanguage

CourseOutline

Module I [10 T+6L]

Primitive Data types & Abstract Data Types(ADT) - Introduction to data structures – definition - characteristics of data structures - categories of data structures – algorithm - space complexity and time complexity of an algorithm.

Module II [6 T+6L]

Arrays & Singly Linked Lists - 1D, 2D and Multi-dimensional arrays – operations on arrays - Sparse matrix Representation

Module III [9 T+6L]

Lists- Linked List- Definition –Creation- Operations ,Basics of Doubly Linked List, Circular Linked List, Header Linked List

Module IV [11 T+6L]

Stack & Queues – Definition & Operations on stack - Implementation of Stack using arrays and linked lists - Applications of Stacks - Polynomial Addition

Queues – Definition, Implementations of queue using arrays and linked lists – basics of Circular queue, Dequeue - Priority queues - Applications of queues.

Module V [10 T+6L]

Searching and Sorting: Searching: Linear search & Binary search. Sorting – Linear sort - Bubble sort - Selection sort - Insertion sort - Quick sort - Merge sort – Comparisons and implementations.

TEXT BOOKS

25. SeymourLipschutz, “DataStructures”,TataMcGraw-HillPublishingCompanyLimited,Schaum’sOutlines,NewDelhi.
26. YedidyanLangsam,MosheJ.Augenstein,andAaronM.Tenenbaum, “DataStructuresUsingC”,PearsonEducation.,NewDelhi.
27. HorowitzandSahani, “FundamentalsofdataStructures”,GalgotiaPublicationPvt.Ltd.,NewDelhi.

REFERENCEBOOKS

1. Trembley, J.P. And Sorenson P.G., “An Introduction to Data Structures With Applications”, Mcgraw- Hill International Student Edition, New York.
2. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, Addison- Wesley, (An Imprint of Pearson Education), Mexico City.

CSC4C05 – Programming Lab: C &Data structure

Semester: 4

Course Number: 5

Number of Credits: 2

Course Evaluation: Internal – 16 Marks + External – 64 Marks

Aim of the Course:

To provide practical skill in Programming

Objectives of the Course:

To develop C Programming skills

To make the students equipped to solve mathematical or scientific problems using C

To learn how to implement various data structures.

Prerequisites:

Background of the basic computing knowledge

Course Outline

LAB LIST

C programming

Write programs and draw flowchart to do the following

1. Find roots of a quadratic equation
2. Find the area and nature of a triangle
3. Find the sum of digits and reverse of a number
4. Find the factorial of a number
5. Find Sin(x)
6. Find Cos(x)
7. Display pyramid using '*'
8. Check for leap year
9. To display count of +ves, -ves and zeros in a set of N numbers
10. Find first n prime numbers
11. Find LCM and HCF of 2 numbers
12. To print Armstrong numbers within range

13. Evaluate the series 1 + x + x^2/2! + x^3/3! + …. + x^n/n!
14. Convert a decimal number to a new base
15. Find the decimal equivalent of a number(base other than 10)
16. Calculate percentage of marks obtained for N students appeared for examination in M subjects.
17. To calculate standard deviation of N numbers.
18. To merge two arrays
19. To find N^{th} Fibonacci number
20. To find row and column totals of a matrix
21. Matrix addition, multiplication and transpose
22. To find the trace of a square matrix
23. To sort n numbers
24. Find the strings end with a particular character
25. Find the number of words in a given sentence
26. To check whether given string is palindrome or not
27. Swapping of two numbers using function
28. Reverse a string using recursion
29. Find the number of vowels in a string
30. To find length of a string using pointer
31. To count the occurrence of a word in a sentence.
32. To generate mark list of N students in a class using array of structures.
33. To insert an element at the correct position in a sorted array
34. To store and read from a text file
35. Write odd and even numbers into two files

Data Structure Programming

1. Sort a given list of strings
2. Reverse a string using pointers.
3. Search an element in a 1-dimensional array
4. Search an element in a 2-dimensiolnal array
5. Implement Pascals Triangle using 2-dimensional array.
6. Merge two sorted array into one sorted array.
7. Search an element in the array using recursive binary search.
8. Implement sparse matrix
9. Implement polynomial using arrays
10. Implement singly linked list of integers.
11. Delete an element from a singly linked list
12. Implement a doubly linked list of integers
13. Implement a circular linked list.
14. Implement polynomial using linked list

15. Addition of 2 polynomials
16. Implement Stack using array
17. Implement Stack using linked list
18. Implement Queue using array
19. Implement Queue using linked list
20. Implement linear sort
21. Implement bubble sort
22. Implement selection sort.
23. Implement insertion sort.
24. Implement quick sort.
25. Implement merge sort.

All lab works should be neatly recorded in a Laboratory Record Book in written form. However Program results can be pasted in the left hand side of the fare record. The laboratory record should have a minimum of:

- 20 lab exercises from C Programming
- 15 lab exercises from Data structure Programming

All students should maintain a rough record (observation note book) too, in which they write all the works to be carried out in the lab prior to his/her entry into the lab. He/She may also note down the i/p and o/p for program verification in the rough record.