VIMALA COLLEGE (AUTONOMOUS)

(NAAC Re-accredited (3rd Cycle): A Grade, CGPA-3.50)

College with potential for Excellence

(Affiliated to University of Calicut)

B.Sc. DEGREE PROGRAMME IN COMPUTER SCIENCE
(Choice Based Credit and Semester System)

SYLLABUS and SCHEME

(2016 ADMISSION ONWARDS)

VIMALA COLLEGE
ENGINEERING COLLEGE P O, THRISSUR
KERALA-680009
INDIA

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) there is further educational opportunity to go for an MCA or other Master's Programme like MSc (Computer Science), MSc (IT), MBA, etc. 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 which 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 37.

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.

Course Evaluation

Total marks for each core and common course including lab courses shall be 100 marks. Total marks for open course and project work 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

20% of the total marks in each core and common course (i.e., 20 marks), 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 are:

Table1: Components of Internal Evaluation of Theory Courses

Sl.No	Components	Marks
1	Attendance(25%)	5
2	Test papers: I & II (50%)	5+5
3	Assignment/ Seminar/Viva (25%)	5
	Total	20

Table2: Components of Internal Evaluation of Lab Courses

Sl.No	Components	Marks
1	Attendance(25%)	5
2	Test papers: I & II (50%)	5+5
3	Assignment/ Lab involvement (25%)	5
	Total	20

Table 3: Percentage of Attendance and Eligible Marks

% of Attendance	Marks
Above 90%	5
85-89%	4
80-84%	3
76-79%	2
75%	1

Table 3: Pattern of Test Papers

Duration	Pattern	Total Number of questions	Number of questions to be answered	Marks for each question	Marks
1.5 Hours	One Word	6	6	1	6
	Short answer	3	3	2	6
	Paragraph	5	3	4	12
	Essay	4	2	8	16
				Total Marks*	40

*90% and above = 5, 80 to below 90% = 4.5, 70 to below 80% = 4, 60 to below 70% = 3.5, 50 to below 60% = 3, 40 to below 50% = 2, 35 to below 40% = 1, below 35% = 0

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 – 4 Marks
Use of Data – 4 Marks
Scheme/Organization of Report – 6 Marks
VivaVoce – 6 Marks

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

EXTERNAL EVALUATION

There shall be Theory and Lab examinations for each course at the end of each semester. 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 except project work.
The model of the question paper for external theory courses examination of 3 Hrs duration shall be

Table 1: Pattern of Question Paper

Duration	Pattern	Total number of questions	Number of questions to be answered	Marks for each question	Marks
3 Hours	One word or one phrase or true or false	10	10	1	10
	Short answer(one or two Sentence)	5	5	2	10
	Paragraph/half page	8	5	4	20
	Essay	8	5	8	40
				Total	80

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. The external examination in practical courses shall be conducted by two examiners, one internal and an external. The project evaluation with programme viva voce will be conducted by two examiners, one internal and an external, at the end of the sixth semester.

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

1. **Section A** One chosen 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 chosen 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)**

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

1. Relevance of the Topic, Statement of Objectives, Methodology (Reference/ Bibliography) **(Total 16 Marks)**
2. Presentation, Quality of Analysis/Use of Statistical tools, Findings and recommendations **(Total 24 Marks)**
3. Project cum Programme Viva Voce **(Total 40 Marks)**

EVALUATION AND GRADING

Mark system is followed 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.

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

$$SGPA = \frac{\text{Sum of the credit points of all courses in a semester}}{\text{Total credits in that semester}}$$

$$ie.,\ SGPA = \frac{C1*G1+C2*G2+C3*G3+\ldots\ldots}{n}$$

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

$$CGPA = \frac{\text{Total credit points obtained in all semesters}}{\text{Total credits}}$$

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.

B.Sc. DEGREE PROGRAMME (COMPUTER SCIENCE CORE) COURSE SRTUCTURE

SEMESTER I

Course No.	Course Code	Title	Hours		Marks		Credit
			T	P	Int.	Ext.	
01	VEG1A01	Communication Skills in English	5	0	20	80	4
02	VEG1A02	Critical Reasoning, Writing and Presentation	4	0	20	80	3
03	VHD1ACS1 or VML1ACS1	Communication Skill in Languages other than English	4	0	20	80	4
04	VCS1B01	Problem Solving Using C	2	2	20	80	3
05	VMT1C01	Complementary Mathematics I	4	0	20	80	3
06	VST1C01	Optional Complementary I	4	0	20	80	3
		Total	25		600		20

SEMESTER II

Course No.	Course Code	Title	Hours		Marks		Credit
			T	P	Int.	Ext.	
07	VEG2A03	Reading Literature in English	4	0	20	80	3
08	VEG2A04	Reading on Indian Constitution Secularism and Sustainable Environment	4	0	20	80	4
09	VHD2ACS2/ VML2ACS2	Literature in Languages other than English	2	0	20	80	3
10	VCS2B02	Data Structures Using C	2	0	20	80	3
11	VCS2BPL1	Programming Laboratory I :Programming in C & Data Structures Using C	0	2	20	80	2
12	VMT2C02	Complementary Mathematics II	4	0	20	80	3
13	VST2C02	Optional Complementary II	4	0	20	80	3
		Total	25		700		22

SEMESTER III

Course No.	Course Code	Title	Hours		Marks		Credit
			T	P	Int.	Ext.	
14	VGC3CS1	1.Basic Numerical Skills 2.General Informatics 3.Fundamentals of Digital Electronics	4	0	20	80	4
15	VGC3CS2		4	0	20	80	4
16	VCS3B03	Visual Programming Using VB.NET	2	1	20	80	3
17	VCS3B04	OOPs Concepts using C++	1	3	20	80	3
18	VMT3C03	Complementary Mathematics III	5	0	20	80	3
19	VST3C03	Optional Complementary III	5	0	20	80	3
		Total	25		600		20

SEMESTER IV

Course No.	Course Code	Title	Hours		Marks		Credit
			T	P	Int.	Ext.	
20	VGC4CS3	Entrepreneurship Development	4	0	20	80	4
21	VGC4CS4	Basics of Audio & Video Media	4	0	20	80	4
22	VCS4B05	Fundamentals of Database Management System and RDBMS	3	2	20	80	3
23	VCS4BPL2	Programming Laboratory II: C++ & RDBMS	0	2	20	80	2
24	VMT4C04	Complementary Mathematics IV	5	0	20	80	3
25	VST4C04	Optional Complementary IV	5	0	20	80	3
		Total	25		600		19

SEMESTER V

Course No.	Course Code	Title	Hours		Marks		Credit
			T	P	Int.	Ext.	
26	VCS5B06	Computer Organization and Architecture	5	0	20	80	4
27	VCS5B07	Java Programming	3	3	20	80	4
28	VCS5B08	Web Programming Using PHP	3	3	20	80	4
29	VCS5B09	Principles of Software Engineering	4	0	20	80	4
30	VCS5D01 VCS5D02 VCS5D03	Open Course	2	0	10	40	2
	VCS5GS	General Seminar	0	2			
		Total	25		450		18

SEMESTER VI

Course No.	Course Code	Title	Hours		Marks		Credit
			T	P	Int.	Ext.	
31	VCS6B10	Android Programming	4	1	20	80	4
32	VCS6B11	Computer Networks	4	1	20	80	4
33	VCS6B12	Fundamentals of Operating Systems	5	0	20	80	4
34	VCS6BPL3	Programming Laboratory III: Java & PHP Programming	0	2	20	80	2
35	VCS6BPL4	Programming Laboratory IV: Android & Network Programming	0	2	20	80	2
36	VCS6E01 VCS6E02 VCS6E03	Elective Course	4	0	20	80	3
37	VCS6PR	Project Work	0	2	10	40	2
		Total	25		650		21

LIST OF OPEN COURSES

Course No	Course Code	Title
30	VCS5D01	Introduction to Computers & Office Automation
30	VCS5D02	Web Designing
30	VCS5D03	Introduction to Problem Solving and C Programming

LIST OF ELECTIVE COURSES

Course No	Course Code	Title
36	VCS6E01	An Introduction to Operational Research
36	VCS6E02	System Software
36	VCS6E03	Microprocessor & Applications

MARK DISTRIBUTION

1	Common: English (4 Courses×100 Marks)	400
2	Additional: Mal/Hindi... (2 Courses×100 Marks)	200
3	General Common Course (4 Courses×100 Marks)	400
4	Core (Theory & Practical) (17 Courses×100 Marks)	1700
5	Project& Viva-Voce	50
6	Open (1 Course)	50
7	Complementary (8 Courses×100 Marks*)(*Number of complementary courses and their mark distribution may vary for complementary courses with laboratory work)	800
	Total Marks	**3600**

COURSE SRTUCTURE

Semester	Course Code	Course Title	Total hours	Hours /week	Credit	Marks
I	VEG1A 01	Common Course I – English	90	5	4	100
	VEG1A 02	Common Course II – English	72	4	3	100
	VHD1ACS1/ VML1ACS1	Common Course III – Language other than English	72	4	4	100
	VCS1B01	Problem Solving Using C	36	2	3	100
		Core Practical I	36	2	*	
	VMT1C01	1st Complementary Course I Mathematics	72	4	3	100
	VST1C01	2nd Complementary Course I	72	4	3	100
		Total	**450**	**25**	**20**	**600**
II	VEG2A 03	Common Course IV – English	72	4	4	100
	VEG2A 04	Common Course V – English	90	5	3	100
	VHD2ACS2/ VML2ACS2	Common Course VI – Language other than English	72	4	4	100
	VCS2 B02	Core Course II -Data Structures Using C	36	2	3	100
	VCS2BPL1	Core Course Practical I	36	2	2	100
	VMT2C02	1st Complementary Course I Mathematics	72	4	3	100
	VST2C02	2nd Complementary Course I	72	4	3	100
		Total	**450**	**25**	**22**	**700**
III	VGC3CS1	Common Course X	72	4	4	100
	VGC3CS2	Common Course X	72	4	4	100
	VCS3 B03	Core Course III -Visual Programming Using VB.NET	36	2	3	100
		Core Course III –Mini Project	18	1	*	
	VCS3 B04	Core Course IV -OOPs Concepts using C++	18	1	3	100
		Core Course IV -Practical II	54	3	*	
	VMT3C03	1st Complementary Course III – Mathematics	90	5	3	100
	VST3C03	2nd Complementary Course III	90	5	3	100
		Total	**450**	**25**	**20**	**600**
IV	VGC4CS1	Common Course X	72	4	4	100
	VGC4CS2	Common Course X	72	4	4	100
	VCS4B05	Core Course V- Fundamentals of Database Management System and RDBMS	54	3	3	100
		Core Course V- Practical II	36	2	*	
	VCS4BPL2	Core Course Practical II	36	2	2	100
	VMT4C04	1st Complementary Course IV – Mathematics	90	5	3	100
	VST4C04	2nd Complementary Course IV	90	5	3	100

		Total	**450**	**25**	**19**	**600**
V	VCS5 B06	Core Course VI - Computer Organization and Architecture	90	5	4	100
	VCS5 B07	Core Course VII - Java Programming	54	3	4	100
		Core Course VII- Practical III	54	3	*	
	VCS5 B08	Core Course VIII -Web Programming Using PHP	54	3	4	100
		Core Course VIII - Practical III	54	3	*	
	VCS5 B09	Core Course IX - Principles of Software Engineering	72	4	4	100
	VCS5D01 VCS5D02 VCS5D03	Open Course – (*course from other streams*):	36	2	2	50
	VCS5GS	General Seminar	36	2	*	
		Total	**450**	**25**	**18**	**450**
VI	VCS6 B10	Core Course X -Android Programming	72	4	4	100
		Core Course X - Practical IV	18	1	*	
	VCS6 B11	Core Course XI -Computer Networks	72	4	4	100
		Core Course XI – Practical IV	18	1	*	
	VCS6 B12	Core Course XII - Fundamentals of Operating Systems	90	5	4	100
	VCS6 E01 VCS6 E02 VCS6 E03	Core Course XII -(Elective)	72	4	3	100
	VCS6B PL3	Core Course Practical III	36	2	2	100
	VCS6BPL4	Core Course Practical IV	36	2	2	100
	VCS6PR	Project Work	36	2	2	50
		Total	**450**	**25**	**21**	**650**
		Total Credit			**120**	**3600**

COURSE STRUCTURE COMPUTER SCIENCE(CORE)

Credit Distribution

	Common course			***Complementary course***			
Semester	***English***	***Additional Language***	***Core course***	***Mathematics***	***Statistics***	***Open course***	***Total***
I	4+3	4	3	3	3	-	20
II	4+3	4	3+2*	3	3	-	22
III	-	-	4+4+3+3	3	3	-	20
IV	-	-	4+4+3+2*	3	3	-	19
V	-	-	4+4+4+4	-	-	2	18
VI	-	-	4+4+4+2*+2*+3+2**	-	-	-	21
Total	**14**	**8**	**72**	**12**	**12**	**2**	**120**

Practical*, Project**

Core Course Structure
Total Credits: 56 (Internal: 20%; External: 80%)

Semester	Course Code	Course Title	Total hours	Hours /week	Credit	Marks
I	VCS1B01	Problem Solving Using C	36	2	3	100
		Core Practical I	36	2	*	
II	VCS2 B02	Core Course II -Data Structures Using C	36	2	3	100
	VCS2BPL1	Core Course Practical I	36	2	2*	100
III	VCS3 B03	Core Course III -Visual Programming Using VB.NET	36	2	3	100
		Core Course III –Mini Project	18	1	-	
	VCS3 B04	Core Course IV -OOPs Concepts using C++	18	1	3	100
		Core Course IV -Practical II	54	3	**	
IV	VCS4B05	Core Course V- Fundamentals of Database Management System and RDBMS	54	3	3	100
		Core Course V- Practical II	36	2	**	
	VCS4BPL2	Core Course Practical II	36	2	2**	100
V	VCS5 B06	Core Course VI - Computer Organization and Architecture	90	5	4	100
	VCS5 B07	Core Course VII - Java Programming	54	3	4	100
		Core Course VII- Practical III	54	3	***	
	VCS5 B08	Core Course VIII -Web Programming Using PHP	54	3	4	100
		Core Course VIII - Practical III	54	3	***	
	VCS5 B09	Core Course IX - Principles of Software Engineering	72	4	4	100
	VCS5GS	General Seminar	36	2	-	
VI	VCS6 B10	Core Course X -Android Programming	72	4	4	100
		Core Course X - Practical IV	18	1	***	
	VCS6 B11	Core Course XI -Computer Networks	72	4	4	100
		Core Course XI – Practical IV	18	1	***	
	VCS6 B12	Core Course XII - Fundamentals of Operating Systems	90	5	4	100
	VCS6 E01 VCS6 E02 VCS6 E03	Core Course XII -(Elective)	72	4	3	100
	VCS6B PL3	Core Course Practical III	36	2	2	100
	VCS6BPL4	Core Course Practical IV	36	2	2	100
	VCS6PR	Project Work	36	2	2	50
		Total			**56**	**1750**

* Exam will be held at the end of 2nd semester
** Exam will be held at the end of 4th semester
***Exam will be held at the end of 6th semester

Common Course Structure

Total Credits: 16 (Internal: 20%; External: 80%)

Semester	Course Code	Course Title	Total hours	Hours /week	Credit	Marks
III	VGC3CS1	Common Course I	72	4	4	100
	VGC3CS2	Common Course II	72	4	4	100
		Total	**144**	**8**	**8**	**200**
IV	VGC4CS1	Common Course I	72	4	4	100
	VGC4CS2	Common Course II	72	4	4	100
		Total	**144**	**8**	**8**	**200**
		Total	**288**	**16**	**16**	**400**

OPEN COURSE STRUCTURE

(FOR STUDENTS OTHER THAN B.Sc. Computer Science)
Total Credits: 2 (Internal 20%; External 80%)

Semester	Course Code	Title	Hours/week	Total Hours	Total Marks
V	VCS5D01	Introduction to Computers & Office Automation	2	36	50
	VCS5D02	Web Designing			
	VCS5D03	Introduction to Problem Solving and C Programming			

OPEN COURSE: EVALUATION SCHEME

The evaluation scheme contains two parts: *viz.,* internal evaluation and external evaluation.
Maximum marks from each unit are prescribed in the syllabus.

1. INTERNAL EVALUATION

20% of the total marks are for internal evaluation. The departments shall send only the marks obtained for internal examination to the college.

Table1: Components of Evaluation

Sl.No	Components	Marks
1	Attendance	2.5
2	Test papers: I & II	2.5 + 2.5
3	Assignment / Viva	2.5
	Total Marks	10

Table2: Percentage of Attendance and Eligible Marks

% of Attendance	Marks
Above 90%	2.5
85-89%	2
80-84%	1.5
76-79%	1
75%	0.5

Table 3: Pattern of Test Papers (Internal)

Duration	Pattern	Total number of questions	Number of questions to be answered	Marks for each question	Marks
1 Hour	One word or one phrase or true or false	4	4	1	4
	Short answer	2	1	2	2
	Paragraph	4	2	3	6
	Essay	2	1	8	8
				Total	20

*Marks: 80% and above = 2.5, 60 to below 80% = 2, 50 to below 60% = 1.5, 40 to below 50% = 1, 35 to below 40% = 0.5, below 35% = 0.

2. EXTERNAL EVALUATION

External evaluation carries 80% marks. External examination will be conducted at the end of 5th semester.

Table 1: Pattern of Question Paper

Duration	Pattern	Total number of questions	Number of questions to be answered	Marks for each question	Marks
2 Hours	One word or one phrase or true or false	6	6	1	6
	Short answer	5	5	2	10
	Paragraph	6	4	4	16
	Essay	3	1	8	8
				Total	40

FIRST SEMESTER

VCS1B01: Problem Solving Using C

Course Number: 4
Contact Hours per Week: 4 (2T + 2P)
Number of Credits: 3
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

- Background of the basic science at +2 level

Course Outline

Module I [7T+7P]

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.

Module II [7T+7P]

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.

Module III [7T+7P]

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, goto statement. Looping while loop, do-while, and for Loops, Nesting of loops, jumps in loop, skipping of loops.

Module IV [8T+7P]

Array & Strings: One dimensional array, two dimensional array and multidimensional array,

strings and string manipulation functions. The Concept of modularization and User defined functions, Multifunction Program, 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.

Module V [8T+7P]

Pointers and Files 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 Defining, Opening and closing files I/O operations on files error handling on files random access of files command line operations. Preprocessor directives, Macro substitution directives, simple macros, macros with arguments, nesting of macros, Compiler control directives.

References

1. E. Balaguruswami, Programming in ANSI C
2. Brian W. Kernighan & Dennis M. Ritchie, The C Programming Language
3. Yashavant P. Kanetkar, Let us C
4. Byran Gotfried, Schaums Outline series Programming with C
5. Ashok N. Kamthane, Programming in C, Pearson, 2nd Edition.

SECOND SEMESTER

VCS2B02: Data Structures Using C

Course Number: 10
Contact Hours per Week: 2T
Number of Credits: 2
Number of Contact Hours: 36 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

Module I [7T]

Introduction to data structures - need for data structures, definition, categories of data structures; Contiguous Data Structures: Arrays: Structure of arrays, Representation of arrays, Operations on one dimensional arrays, Multidimensional arrays, String representation and manipulation. Sparse matrix representation.

Module II [7T]

Non-Contiguous Linear Data Structures: Non-contiguous Data Structures: Lists: Representation and Traversing of linked list, Operations with linked list, Doubly linked list, Circular list, Stacks: Definition, Operation on stack, Implementation using arrays and linked lists, Evaluation of arithmetic expressions, Conversion of Expressions - Prefix, Infix and Postfix expressions. Queues: Definition, Implementations using arrays and linked lists, Circular queue, priority queue, dequeue. Applications of queues.

Module III [7T]

Trees - Definition, Basic terminology, Binary trees, Representation of binary trees, Sequential representation of binary trees, Linked representation of binary trees, Traversals, Threaded binary tree. Binary Search Trees: Definition, Insertion, Deletion, Traversal and Searching BST, AVL Trees, B-Tree and B+ Tree, Heap tree: Insertion and deletion.

Module IV [7T]

Graphs: Graphs: Representation of graphs, Adjacency Matrix, Adjacency List, Adjacency Multi-list, Graph search methods (BFS and DFS), Minimal Spanning Tree, Prims Algorithm and Kruskals Algorithms, Shortest path problem, Dijkstras Algorithm

Module V [8T]

Searching and Sorting: Searching: Linear search, Binary search, Comparison of different methods, Sorting: Insertion, Bubble, Selection, Quick and Merge sort methods, Comparisons, Hashing: Different hashing functions, Methods for collision handling.

References:

1. Horowitz & S. Sahni, Fundamentals of data structures
2. E. Aron M, Tenenbaum, Data Structure Using C and C++

VCS2BPL1: Programming Laboratory I
Programming in C &Data Structure Using C

Course Number: 11
Contact Hours per Week: 2P
Number of Credits: 3
Number of Contact Hours: 36 P.Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- To make the students learn programming environments.
- To practice procedural concepts.
- 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 C Programming.

Course Outline:

Part A: C Programming
Experiments should include but not limited to:

Lab 1: Simple C Programs like Computation of area of a circle, rectangle etc.
Lab 2: C Programs using Loops (like pyramid printing, factorial computation, number reversing etc.)
Lab 3: Programs involving Arrays (Searching, sorting and vector operations etc.)
Lab 4: Two dimensional arrays (like Matrix operations)
Lab 5: String Manipulations
Lab 6: Programs involving Structures (like addition of Two Complex numbers, student record creation and manipulation etc.)
Lab 7: Programs involving Union
Lab 8: Programs involving functions

Lab 9: Recursion (recursive function to compute a factorial, reverse string etc)
Lab 10: Pointers - simple programs to learn concept of pointers, array operation using pointers etc.
Lab 11: File operations
Lab 12: Command line arguments copy a file, delete a file etc.

Projects:

1. Design a Scientific Calculator and include as many functions as possible
2. Prepare rank list for finding the topper in the considering the grade obtained in various subjects.
3. Develop a telephone directory.
4. Write a program for sorting a list of input elements. User should be able to give an option for ascending order or descending order. Use command line arguments.

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).

Part B: Data Structure Using C

Lab 1: Experiments should include but not limited to:
Lab 2: Array operations and string operations
Lab 3: Implementation of sparse matrix
Lab 4: Stacks operation using array
Lab 5: Stacks: adding, deleting elements using linked list
Lab 6: Circular Queue: Adding & deleting
Lab 7: Implementation of linked lists: inserting, deleting, inverting a linked list.
Lab 8: Implementation of queues using linked lists
Lab 9: Implementation Polynomial addition, Polynomial multiplication using linked lists.
Lab 10: Implementation of doubly linked list
Lab 11: implementation of queue using array
Lab 12: Implementation of searching Techniques: Linear and Binary search.
Lab 13: Sorting techniques: Bubble Sort, Insertion Sort, Quick Sort, and Merge Sort.
Lab 14: Simple Hashing Functions.

THIRD SEMESTER

VCS3B03: Visual Programming Using VB.NET

Course Number: 16
Contact Hours per Week: 3 (2T + 1P)
Number of Credits: 3
Number of Contact Hours: 54 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objectives of the Course

- To get a general understanding on .Net Frame Work
- To get a general understanding on ADO.Net

Prerequisites

- Basic knowledge of OOP

Course Outline

Module I [7T+3P]

Introduction to visual programming - Concept of event driven programming - Introduction to VB.Net environment, The .NET Framework and the Common Language Runtime. Building VB.NET Applications, The Visual Basic Integrated Development - Basic Language - Console application and windows application, Data types, Declaring Variables, scope of variables, operators and statements.

Module II [7T+3P]

Making Decisions with If . . . Else Statements, Using Select Case, Making Selections with Switch and Choose, Loop statements - Do Loop, for, while – The With Statement - Handling Dates and Times - Converting between Data Types - Arrays - declaration and manipulation - Strings & string functions – Sub Procedures and Functions.

Module III [7T+4P]

Windows Applications - Forms - Adding Controls to Forms, Handling Events, MsgBox , InputBox , Working with Multiple Forms, Setting the Startup Form, SDI & MDI Forms, Handling Mouse & Keyboard Events, Common controls (Text Boxes, Rich Text Boxes, Labels, Buttons, Checkboxes, Radio Buttons, Group Boxes, List Boxes, Checked List Boxes, Combo Boxes, Picture Boxes, Scroll Bars, Tool Tips, Timers) properties - methods

Module IV [7T+4P]

Object-Oriented Programming - Creating and using Classes & objects – Handling Exceptions - On Error GoTo - Raising an Exception - Throwing an Exception - Using Structured Exception Handling - Debugging and tracing.

Module V [8T+4P]

Data Access with ADO.NET - Accessing Data with the Server Explorer - Accessing Data with Data Adaptors and Datasets - Creating a New Data Connection - Creating and populating Dataset - Displaying Data in a Data Grid - Selecting a Data Provider - Data Access Using Data Adapter Controls – Binding Data to Controls - Handling Databases in Code - Binding to XML data.

References:

1. Steven Holzner, Visual Basic .NET Black Book
2. Rebecca Riordan, VB.NET for Developers, Keith Franklin, SAMS
3. Jason Beres, Sams Teach Yourself Visual Studio .NET 2005 in 21 Days
4. Jesse Liberty, Learning Visual Basic .NET
5. Tim Anderson, Visual Basic .Net programming in Easy Steps, DreamTech Press.
6. .NET Programming (6-in-1), Black Book, Kogent Learning Solutions Inc.,
7. Wiely- Dream Tech Press
8. Francisco, Visual Studio .Net, Microsoft Publication.
9. .Net Framework Essentials, O'Reilly

VCS3B04: OOPs Concepts Using C++

Course Number: 17
Contact Hours per Week: 4 (1T + 3P)
Number of Credits: 3
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Objectives of the Course

- To learn basic concepts of OOPS.
- To learn Object Oriented Programming.
- To learn C++.

Prerequisites

- Basic knowledge in C

Course Outline

Module I [3T+10P]
OOP Concepts - Characteristics of OOPS - Object oriented languages. C++ Fundamentals: C++ data types, operators, expressions. Control structures: Branching and looping statements-break and continue statements. User defined functions - Inline functions.

Module II [3T+11P]
Classes and objects: Defining classes, defining member functions, creating objects. Passing and returning objects to and from functions- Friend functions- Constructors: default constructors, parameterized constructors, constructors with default arguments, copy constructors- destructors.

Module III [4T+11P]
Overloading and polymorphism: Operator overloading, function overloading, constructor overloading. Inheritance: Single, multiple, multilevel, hierarchical and hybrid inheritance.

Module IV [4T+11P]
Pointers - Dynamic memory management- new and delete operators, Pointers to functions - Pointers to objects-Accessing members- this pointer- Pointers to derived objects - Virtual functions - Virtual base classes.

Module V [4T+11P]
Templates: Generic Functions, Applying Generic Functions, Generic Classes, class templates, function templates.

References:

1.Robert C Lafore ,Object oriented programming with C++
2. Bjarne Stroustrup, `The C++ Programming Language', Addison Wesley, 1999.
3. Herbert Schildt, C++: The Complete Reference
4. Object Oriented Programming with C++, E Balagurusamy

FOURTH SEMESTER

VCS4B05: Fundamentals of Database Management System and RDBMS

Course Number: 22
Contact Hours per Week: 5 (3T + 2P)
Number of Credits: 3
Number of Contact Hours: 90 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

Module I [10T+7L]
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. 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

Module II [11T+7L]
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. SQL Data Definition in SQL creation, updation , deletion of tables, modifying the structure of tables, renaming, dropping of tables. Constraints. Database Manipulation in SQL Select command, Logical operators, Range searching, Pattern matching, Grouping data from tables in SQL, GROUP BY, HAVING clauses, Joins Joining Multiple Tables, Joining a Table to itself. Views Creation, Renaming the column of a view, destroys view.

Module III [11T+7L]
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.

Module IV [11T+7L]
Transaction Management and Concurrency Control Transaction Properties (ACID), states, Commit, Rollback; Concurrency ControlLost update problems, Locks, two phase locking.

Module V [11T+8L]
Programming with SQL: Data types, Using set and select commands, procedural flow, if, if /else, while, goto, global variables, Security: Locks, types of locks, levels of locks. Cursors: Working with cursors, Error Handling, Developing stored procedures, create, alter and drop, passing and returning data to stored procedures, using stored procedures within queries, building user defined functions, creating and calling a scalar function, implementing triggers, creating triggers, multiple trigger interaction.

References:

1. Abraham Silberschatz, Henry F Korth, S.Sudharshan, Database System Concepts
2. Ivan Bayross, PL/SQL: The Programming Language of Oracle SQL.
3. Alex Krigel and Boris M.Trukhnov, SQL Bible, Wiley pubs
4. Paul Nielsen, Microsoft SQL Server 2000 Bible, Wiley Dreamtech India Pubs.
5. CJ Date, Introduction to Database Systems, Addison Wesley.
6. Ramkrishnan, Database Management Systems, McGraw Hill.

VCS4BPL2: Programming Laboratory II: C++ & RDBMS

Course Number: 23
Contact Hours per Week: 2 (0T + 2P)
Number of Credits: 2
Number of Contact Hours: 36 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- To practice Visual Programming using VB.NET.
- To learn practical database design.
- To create & manipulate various database objects.
- To practice SQL Commands.

Prerequisites

- Theoretical knowledge in Visual Programming.
- Theoretical knowledge in DBMS & SQL.

Course Outline:

Part A: C++ Lab

Lab 1. Programs covering Basics of C++, Control Structures etc.,
Lab 2. Programs to demonstrate Class and Objects,f unction calling,
Lab 3. Programs to demonstrate Constructor and Destructor,
Lab 4. Programs to demonstrate Inheritance,
Lab 5. Programs to demonstrate Operator Overloading,
Lab 6. Programs to demonstrate Polymorphism,
Lab 7. Programs to demonstrate I/O and streaming,
Lab 8. Programs to demonstrate Exception Handling,

Part B: DBMS Lab

Lab 1: 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/PL/SQL.

Create the following tables with the specified constraints:

1. ***Department:***
    a. Department name - Not NULL unique
    b. Department number - Primary Key
    c. ManagerId - Refers to employee-id of employee table.
    d. Manager
    e. date of joining - Not NULL.
2. ***Employee:***
    a. First name - Not NULL
    b. Middle initials
    c. Last name - Not NULL
    d. Employee id - Primary Key
    e. Date of Birth - Not NULL
    f. Address
    g. Gender - M or F
    h. Salary - Range of 5000 to 25000
    i. Date of Joining
    j. Department number - Refers to Department Number of Department table.
3. ***Department location:***
    a. Department number - Refers to Department number of department table.
    b. Department location - Not NULL.
    c. Department number & Department location are combined Primary Key
4. ***Project:***
    a. Project name-Not NULL.
    b. Project number-Primary Key.
    c. Project location-Not NULL.
    d. Department number-Refers to department number of Department table.
5. ***Works-on:***
    a. Employee-id - Not NULL refers to employee-id of employee table.
    b. Project number- Not NULL refers to Project number of Project table.
    c. Hours - Not NULL.
    d. Employee-id & Project number are combined primary key.
6. ***Dependent:***
    a. Employee-id - Refer to employee table employee id field
    b. Dependent name - Gender - M or F
    c. Date of Birth - Not NULL
    d. Relationship - Not NULL

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

1. List the department wise details of all the employees.
2. Find out all those departments that are located in more than one location.
3. Find the list of projects.
4. Find out the list of employees working on a project.
5. List the dependents of the employee whose employee id is 001

Lab 2: 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:

1. ***Book Records:***
    a. Accession Number
    b. ISBN Number
    c. Books:
    d. ISBN Number
    e. Author
    f. Publisher
    g. Price
2. ***Members:***
    a. Member Id
    b. Member Name
    c. Maximum Number of books that can be issued
    d. Maximum Number of days for which book can be issued
3. ***Book Issue:***
    a. Member Id
    b. Accession Number
    c. Issue Date
    d. 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:

1. Insert data in all the three tables (use insert).
2. Insert appropriate description associated with each table and the column (use comment).
3. Display the structure of the tables.
4. Display the comments that you have inserted
5. 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.

Lab 3: 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.

1. Get the list of ISBN-Number, Book name, available copies of the books of which available copies are greater than zero.
2. 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.
3. 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.
4. Get the list of books issued to each member.
5. Write query to know the maximum and average price of the books.
6. 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.
7. Get the list of member codes of those members who have more than two books issued.
8. Find the details of the books presently issued to a member
9. Create the history of issue of a book having a typical accession number.
10. To set the width of the book name as 35.

Lab 4: 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.

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

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

Lab 5: 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 to them, use any inbuilt function and operators like IN, ANY, ALL, EXISTS.

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

operator.

2. List the members who have got issued at least one book (use IN / ANY operator).
3. List the books which have maximum Price using ALL operator.
4. Display Book Name, Member Name, Issue date of Book. Create a view of this query of the currently issued books

Lab 6: Create a table of Employee (emp-number, name, dept, 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

1. Display all records from employee table where department is not found in department table.
2. 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.
3. Display those employee records who have salary less than the salary of person whose empcode= A100.
4. Create another table: SalesData (RegionCode, City, Salesperson- Code, SalesQty).
5. Display records where salesperson has achieved sales more than average sales of all sales persons of all the regions.

Lab 7: 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.

Lab 8: Create a form for storing bio data of students. Create the supporting tables to store the data.

Lab 9: Design a suitable form for storing basic information and salary details of employees of an organization. Design and implement the necessary tables

Lab 10:

1. Write a procedure/trigger on department code so such that the validity of the code is checked and the name of department is automatically displayed on selection of department code. Assume, design and create the necessary tables and constraints
2. Write a procedure/ trigger to generate Order Number automatically in any of the order tables created in Lab 7

FIFTH SEMESTER

VCS5B06: Computer Organization and Architecture

Course Number: 26
Contact Hours per Week: 5 (4T + 0P)
Number of Credits: 5
Number of Contact Hours: 90 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- To learn basic Architecture of a Computer
- To learn basic Computer Organization.

Prerequisites

- Basic knowledge of Computer.

Course Outline

Module I [18T]
Basic Computer Organization and Design: Instruction Codes , Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory reference Instructions, Input, Output and Interrupt, Design of Basic Computer, Design of Accumulator logic.

Module II [18T]
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. Computer Arithmetic: Addition, Subtraction, Multiplication, Division algorithms - Floating point arithmetic operations, Decimal arithmetic operations.

Module III [18T]
Memory Organization: Memory Hierarchy, Main memory (RAM/ROM chips), Auxiliary memory, Associative memory, Cache memory, Virtual Memory, Memory Management Hardware, hit/miss ratio, magnetic and optical storage devices

Module IV [18T]
Input-Output Organization: Peripheral devices, I/O interface, Modes of Transfer, Priority Interrupt, Direct Memory Access, Input-Output Processor, and Serial Communication. I/O Controllers, Asynchronous data transfer, Strobe Control, Handshaking.

Module V [18T]
Parallel Processing: Basic Parallel Processing Architecture - Taxonomy- SISD. MISD, SIMD, MIMD structures - CISC Vs RISC - Symmetric Multiprocessors - Cache coherence and MESI protocol - Clusters - Non Uniform Memory Access. Pipelining: Basic Concepts of pipelining, Instruction Pipelining. Hazards, Reservation Tables, Collision, Latency, Dynamic pipeline, Vector processing & Vector processors

References:

1. M. Morris Mano, Computer System ArchitecturePHI
2. William Stallings, Computer Organization and Architecture, PHI.

VCS5B07: Java Programming

Course Number: 27
Contact Hours per Week: 6 (3T +3P)
Number of Credits: 4
Number of Contact Hours: 108 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- To have a 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

Module I [10T+10P]

Introduction to Java: History, Versioning, The Java Virtual Machine, Writing a Java Program, Packages, Simple Java Programs.Language Components: Primitive Data Types, Comments, The for Statement, The if Statement, The while and do while Statements, The switch Statement, The break Statement, The continue Statement, Operators - Casts and Conversions, Keywords.

Module II [11T+11P]

Object-Oriented Programming: Defining New Data Types, Constructors, The String Class, String Literals, Documentation, Packages, The StringBuffer Class, Naming Conventions, The Date Class, The import Statement, The StringTokenizer Class. Methods: Introduction - Method Signatures, Arguments and Parameters, Passing Objects to Methods, Method Overloading, Static Methods, The Math Class, The System Class, Wrapper Classes Arrays: Processing Arrays,
Copying Arrays, Passing Arrays to Methods, Arrays of Objects, The Arrays Class, Command Line Arguments, Multidimensional Arrays. Encapsulation: Constructors, The this Reference, Data Hiding, public and private Members, Access Levels, Static Data Members Inheritance & Polymorphism: Inheritance, extends keyword, Polymorphism, The Object Class, Method Overloading & Overriding. Abstract Classes and Interfaces: Abstract Classes, Abstract Class Example, Extending an Abstract Class, Interfaces.

Module III [11T+11P]

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, Triggering Exceptions with throws, Suppressing Exceptions with throw, Developing user defined Exception Classes-The finally Block.

Module IV [11T+11P]
Collections & Database Connectivity Collections: Vectors, Hashtables, Enumerations, properties, Collection, Framework Hierarchy, Lists, Sets, Maps, The Collections Class. Networking: Networking Fundamentals, The Client/Server Model , InetAddress, URLs, Sockets, Writing Servers, Client/Server Example.
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, Error Checking and the SQLException Class, The Statement Interface, The ResultSet Interface, ResultSetMetaData, Transaction Management.

Module V [11T+11P]
Applets, Events and GUI Applications: Introduction to GUI Applications – Applets - Types of Applet, Applet Skeleton, Update Method, Html Applet tag and passing parameter to applet. Event Handling: The Delegation Event Model, Event Classes, Event Listener Interfaces, Adapter Classes, Inner Classes.
Java Desktop Applications, Introduction to the AWT, Overview of the AWT, Structure of the AWT, The AWT hierarchy, Working with: Color, Button, Canvas, Checkbox, Choice, Frame, Label, List, Scroll bar, TextArea, TextField, Font, FontMetrics, Graphics, Image, Menu Component, MenuBar, MenuItem, Checkbox MenuItem, Menu, Point, Polygon, Rectangle, Layout Manager, Menu Component, Containers, Components, Event handling, Simple Graphics Drawing Lines, Rectangles,etc.

References:

1. Herbert Schildt, Java Complete Reference, TMH
2. Jim Keogh, J2EE Complete Reference, TMH
3. David Flanagan, Jim Farley, William Crawford, Kris Mangnusson, Java Enterprise in a Nutshell, OReill.
4. Patrick N,Schildt H, Java 2 The Complete Reference, TMH

VCS5B08: Web Programming Using PHP

Course Number: 28
Contact Hours per Week: 6 (3T + 3P)
Number of Credits: 4
Number of Contact Hours: 108 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- To learn client side and server side scripting.
- To learn PHP Programming.
- To learn how to develop dynamic websites.
- To learn how to interact with databases through internet.

Prerequisites

- Knowledge in Programming

Course Outline

Module I [10T+10L]

www, W3C, Web Browser, Web server, Web hosting, Web Pages, DNS, URL, Introduction e-documents - Static, Active & Dynamic. Web programming - client- side scripting and server-side scripting. HTML: Introduction to HTML, Basic formatting tags: heading, paragraph, underline break, bold, italic, underline, superscript, subscript, font and image. Different attributes like align, color, bgcolor, font face, border, size. Navigation Links using anchor tag: internal, external, mail and image links.Lists: ordered, unordered and definition, Table tag, HTML Form controls: form, text, password, textarea, button, checkbox, radio button, select box, hidden controls, Frameset and frames CSS: Introduction to Cascading Style Sheet (CSS), CSS Syntax, Comments, Id and Class, Background - Background Color, Background Image -Text - Text Color, Text Alignment, Text Decoration, Text Transformation, Text Indentation - CSS Font - Font Families, Font Style, Font Size -Setting Text Size - Using Pixels and Em - CSS Lists -
Different List Item Markers, Unordered List, Ordered List, An Image as The List Item Marker - CSS Tables - Table Borders, Collapse Borders, Table Width and Height, Table Text Alignment, Table Padding, Table Color CSS Positioning - Static Positioning, Fixed Positioning, Relative Positioning, Absolute Positioning, Overlapping Elements - Float - Horizontal Align - Image Gallery – Image Opacity/Transparency - Image Sprites

Module II [11T+11L]

Javascript: Introduction, Client side programming, script tag, comments, variables, Document Methods: write and writeln methods, alert, Operators: Arithmetic, Assignment, Relational, Logical, Javascript Functions, Conditional Statements, Loops, break and continue. Events Familiarization: onLoad, onClick, onBlur, onSubmit, onChange

Module III [11T+11L]

PHP: Introduction to PHP, Server side scripting, Role of Web Server software, including files, comments, variables and scope, echo and print, Operators: Logical, Comparison and Conditional operators, Branching statements, Loops, break and continue, PHP functions.

Module IV [11T+11L]

Working with PHP: Passing information between pages, HTTP GET and POST method, String functions: strlen, strops, strstr, strcmp, substr, str_replace, string case, Array constructs: array(),list() and foreach(), PHP advanced functions:Header , Session, Cookie, Object-Oriented Programming using PHP: class, object, constructor, destructor and inheritance.

Module V [11T+11L]

PHP & MySQL: Features of MySQL, data types, Introduction to SQL commands - SELECT, DELETE, UPDATE, INSERT, PHP functions for MySQL operations: mysqLconnect, mysql_select_db, mysqLquery, mysql_fetch_row, mysql_fetch_array, mysql_fetch_object, mysqLresult, Insertion and Deletion of data using PHP, Displaying data from MYSQL in webpage. Introduction to AJAX, Implementation of AJAX in PHP, Simple examples like partial page update, Concept of master page, applying templates.

References:

1. Jon Duckett, Beginning Web Programming with HTML,XHTML, CSS, Wrox.
2. Jim Converse & Joyce Park, PHP & MySQL Bible, Wiley
3. Deitel, Harvey M. and Paul J., 3/E, Internet & World Wide Web How To Program, 2004

4. Kogent Solutions, HTML 4.0 in Simple Steps, Wiley
5. Ed Tittel & Mary Burmeis – Ter, HTML 4 for Dummies, Wiley
6. D W Mercer,A Kent,S D Nowicki, Beginning PHP, Wrox.
7. Janet Valad, PHP & MYSQL For Dummies, Wiley

VCS5B09: Principles of Software Engineering

Course Number: 29
Contact Hours per Week: 4 (4T + 0P)
Number of Credits: 4
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

Module I [15T]
Introduction to Software Engineering, Software Engineering Approach, Software Process, Characteristics of a Software Process, Software Development Process Models, Process Management Process

Module II [14T]
Software Requirements, Problem analysis, Requirements Specification, Functional specification with use case, Validation.

Module III [14T]
Function oriented Design Principles, Module Level Concepts, Structured Design Methodology, Verification, Object oriented analysis and design, Object Oriented Concepts, Design Principles, Unified Modelling Language, Design Methodology for Object oriented design. Verification-Design walkthrough, Critical Design Review, Consistency checks

Module IV [14T]
Coding - Programming Principles and Guidelines, Coding Process, Verification Code Inspection, Unit Testing

Module V [15T]
Testing - Testing Fundamentals, Black Box Testing, White Box Testing, Testing Process

References:

1. Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Pub.
2. Roger S. Pressman, Software Engineering - A Practical Approach, McGrawHill
3. Ivan Somervelli, Software Engineering

SIXTH SEMESTER

VCS6B10: Android Programming

Course Number: 31
Contact Hours per Week: 5 (4T + 1P)
Number of Credits: 4
Number of Contact Hours: 90 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

Module I [14T+3P]
Introducing the android computing platform, History of android, an- droid software stack, 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 real device, Structure of android application, Application life cycle.

Module II [14T+3P]
Understanding android resources - String resources, Layout resources, Resource reference syntax, Defining own resource IDs - Enumerating key android resources, string arrays, plurals, Colour resources, di- mension resources, image resources, Understanding content providers - android built in providers, exploring databases on emulator, architec- ture 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 – basics of intents, available intents, exploring intent composi- tion, Rules for Resolving Intents to Their Components, ACTION PICK, GET CONTENT, pending intents

Module III [14T+4P]
User interfaces development in android - building UI completely in code, UI using XML, UI in XML with code, Android's common controls - Text controls, button controls, checkbox control, radio button controls, image view, date and time controls, map view control, understanding adapters, adapter views, list view, grid view, spinner control, gallery control, styles and themes, Understanding layout managers - linear layout manager, table layout manager, relative layout manager, frame layout manager, grid layout manager.

Module IV [15T+4P]
Android menus - creating menus, working with menu groups, responding to menu items, icon

menu, sub menu, context menu, dynamic menus, loading menu through XML, popup menus, Fragments in an- droid - structure of fragment, fragment life cycle, fragment transaction and back stack, fragment manager, saving fragment state, persistence of fragments, communications with fragments, startActivity() and set- TargetFragment(), using dialogs in android, dialog fragments, working with toast, Implementing action bar - tabbed navigation action bar activity, implementing base activity classes, tabbed action bar and tabbed listener, debug text view layout, action bar and menu interaction, list navigation action bar activity, spinner adapter, list listener, list action bar, standard navigation action bar activity, action bar and search view, action bar and fragments.

Module V [15T+4P]

Persisting data - Files, saving state and preferences - saving application data, creating, saving and retrieving shared preferences, preference framework and preference activity, preference layout in XML, native preference controls, preference fragments, preference activity, persisting the application state, including static 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. Pro Android 4, Satya Komatineni & Dave MacLean, Apress.
2. Professional Android 4 Application Development, Retomeier, Wrox.
3. Programming Android, Zigurd Mednieks, Laird Dornin, G. Blake Meike, and Masumi Nakamura, O'Reilly

VCS6B11: Computer Networks

Course Number: 33
Contact Hours per Week: 5 (4T +1P)
Number of Credits:4
Number of Contact Hours: 90 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

Module I [14T+3P]

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.

Module II [14T+3P]

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 bit 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, Bluetooth

Module III [15T+4P]

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, Multicast routing protocols, The Dijkstra Algorithm.

Module IV [15T+4P]

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; Network management: SNMP, Network security, Cryptography.

Module V [14T+3P]

Network Administration, IP address - Configuring network host - setting hostname - assigning IP address, configuring the Network Interface card, Setup a LAN with more than two systems, Setting up Internet services File Transfer Protocol(FTP), Trivial File Transfer Protocol(TFTP),Simple Mail Transfer Protocol(SMTP) and Post Office Protocol(POP), Setting up Intranet Services, Network File System(NFS), Network Information Service(NIS) and Dynamic Host Configuration Protocol(DHCP), Samba printing and Web server.

References

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

VCS6B12: Fundamentals of Operating Systems

Course Number: 32
Contact Hours per Week: 5 (5T +0P)
Number of Credits: 4
Number of Contact Hours: 90 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

Module I [18T]
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.

Module II [18T]
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, Dead lock Prevention, Dead lock detection, dead lock avoidance.

Module III [18T]
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, File and Database System, File System, Functions of organization, Allocation and Free space management.

Module IV [18T]
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, Thrashing.

Module V [18T]
Protection and security: policy and mechanism, authentication, authorization, Mobile OS: Concepts, history, features, architecture, future scope.
Unix/Linux commands: date, echo, now, cat, cp, chmod, cut, mv, rm, ls, more, wc, mkdir, cd, pwd, who, whoami, grep, sort, tail, touch, cmp, echo, exit, nice

References:

1. Silberschatz, Galvin, Gagne, Operating System Concepts, John Willey & Sons.
2. Nutt G.J, Operating Systems: A Modern Perspective, Addison Wesley.
3. William Stallings, Operating Systems, Internals and Design Principles, - PHI.

VCS6BPL3: Programming Laboratory III: Java & PHP Programming

Course Number: 34
Contact Hours per Week: 2 (0T +2P)
Number of Credits: 2
Number of Contact Hours: 36 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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 java program to display n prime numbers
2. Write java program to display Fibonacci series upto n
3. Write program to find the roots of a quadratic equation
4. Write java program to compute the mean and SD of 3 numbers. Mean=(a+b+c)/3, SD=√((a-m)2+(b-m)2+(c-m)2)/3
5. Write a java program to count the number of even numbers, odd numbers, positive numbers, negative numbers and zeros in an array.
6. Write a java program to display transpose of a matrix.
7. Write java program to find the sum of the digits and reverse of a given number using class and objects.
8. Write a program in java with class rectangle with the data fields length, breadth. Calculate the area. Create two objects of rectangle and compare their area.
9. Write a java program to find the volume of cube, rectangular box, cylinder using function overloading.
10. Write a java program to create a class complex. Create two objects and find the sum of two complex numbers. Read the real and imaginary part using constructor.
11. Find the area of a square and a rectangle using overloaded constructors.
12. Write a java program to create a class student. Write methods to read and display the student details. Create a derived class result. Write methods to read mark of 5 subjects and display the total and grade. Create object to read and display the result of a student.

13. Write a program to create a class worker and create derived classes daily worker and salaried worker. Worker class should contain methods to read and display the personal details. Class daily worker should contain methods to read the number of days worked and wage per day and display. Class salaried worker should contain methods to read the basic salary and allowance and display. Create two objects of daily worker and salaried worker to display the work details and salary.
14. Design two interfaces Item and Salesman. Item should contain data members Item code, Item name, Unit price. Salesman should contain data members Salesman number, name, address. Create a class productsale contain members to read the quantity sold and display the sales details.
15. Write a java program for generating two threads. One for odd numbers and one for even numbers.
16. Create a package for finding the factorial of a number.
17. Write a java program to enable arithmetic exceptions.
18. Write a program to create a class account with members deposit(), withdraw() and getbalance(). While withdrawing, check the balance. If the balance is < 500, raise a user defined exception.
19. Write java program for finding the length of a given string and the number of vowels in it. Find the first occurrence of a character.
20. Create an applet program to read amount purchased and type of cloth(mill cloth or hand loom cloth). Find the discount rate, based on the conditions.

Amount	mill	handloom
0 – 100		5%
101-200	5%	7.5%
201-300	7.5%	10%
Above 300	10%	15%

21. Write a java program to override method greatest() for finding the greatest of 2 numbers and 3 numbers.
22. Write an applet program for currency denomination
23. Write an applet to display a rectangle with specified coordinate and colour passed as parameter from the HTML file.
24. Create an AWT application to add, remove, search and count items in a listbox.
25. Create and AWT application to list the names of authors in a choice box. On selecting an author, list the books of the selected author in a list box.
26. Create a menu driven program to display different shapes.
27. Write simple client/server program using socket
28. Create an applet for a displaying smiling face.
29. Write a java program to insert and display book details using jdbc and mysql. The book table should contain bookid, bookname, author, publisher, price.
30. Create a table in mysql “product” with productno, name, description and unitprice. Insert 5 records. Create a java program to search for a particular record and delete it.

Part B: PHP Programming

Lab 1: Setting-up the environment

- Setup WAMP/XAMPP Server or Setup Apache, MySQL and PHP separately in your PHP Lab.

- Simple PHP program that displays a welcome message.
- Write a php program to generate a random number between 1 and 100.
- Modify above program to accept range of the random number from HTML interface.

Lab 2: Programs involving various control structures like:

- 'if, else, elseif/else if'
- Alternative Syntax for 'if, else, elseif/else if'

Lab 3: Programs involving various control structures like:

- while, do-while, for, foreach, switch, break, continue.
- Try alternative syntax for while, do-while, for, foreach, switch.

Lab 4: Programs involving the following.

- declare, return.
- require, include, require- once, include_once and goto.

Lab 5: Programs to demonstrate PHP Array functions.

- PHP Array Sorting,
- PHP Key Sorting,
- PHP Value Sorting,
- PHP MultiArray Sorting,
- PHP Array Random Sorting

Lab 6: Programs to demonstrate PHP Array functions.

- PHP Array Reverse Sorting,
- Array to String Conversion,
- Implode() function,
- String to Array, Array Count,
- Remove Duplicate Values

Lab 7: Programs to demonstrate PHP Array functions.

- Array Search,
- Array Replace,
- Array Replace Recursive,
- Array Sub String Search

Lab 7: Demonstrate the following.

- Use of regular expression to compare two strings.
- Extract Domain name from URL.
- Find the number of rows from a mysql database for your query.

Lab 8: Generate a Guestbook which will allow your website visitor to enter some simple data about your website.

Lab 9: Develop a PHP program for Email Registration.

Lab10: Develop a project for making Application form and performing Degree Admission On-line.

VCS6BPL4: Programming Laboratory IV: Android& Network Programming

Course Number: 35
Contact Hours per Week: 2 (0T +2P)
Number of Credits:2
Number of Contact Hours: 36 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

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

Prerequisites

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

Course Outline

Part A: Android Programming

Lab 1: Programs to understand basic arithmetic operations
Lab 2: Programs to understand basic logic operations
Lab 3: Programs to understand loops and control statements
Lab 4: Programs to understand GUI in android
Lab 5: Android application for adding two numbers
Lab 6: Develop simple user interface to display message
Lab 7: Create two menu items-opening a file-saving a file
Lab 8: Text view controls to represent each row in a list view
Lab 9: Implementation of background image
Lab 10: Starting another activity from your own activity using intent
Lab 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
Lab 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
Lab 13: Implementation of array adapter
Lab14: 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.
Lab 15: Create an earth quake viewer
Lab 16: Create mobile applications
Lab 17: Program to implement simple calculator
Lab 18: Program to Get IP Address
Lab 19: Program to Home And Lock Screen Widget (Temperature Widget)
Lab 20: Program to Device/Battery Temperature Sensor
Lab 21: Program to Audio Demo AudioTrack , AudioRecord
Lab 22: Program to Blocking Incoming call Android
Lab 23: Program to create simple login screen.

Part B: Network Programming

1. Program to implement a simple client server socket program
2. Program to implement echo server
3. Program to implement RSA algorithm
4. Program to implement RMI Implementation
5. Program to implement File transfer
6. Program to create a simple servlet that displays a welcome message.
7. Program to pass employee name and phone number to a servlet and prints it
8. Program to implement get method
9. Program to implement post method
10. Program to display cookie name and value
11. Program to implement session tracking
12. Program to implement URL connection
13. Program to implement datagram client and server
14. Program to implement TCP/IP server socket
15. Program to implement a substitution cipher algorithm
16. Program to implement a transposition cipher algorithm
17. Program to compute the Hamming code and detect the error

VCS6PR: Project

Course Number: 37
Contact Hours per Week: 2P
Number of Credits: 2
Number of Contact Hours: 36 Hrs.
Course Evaluation: Internal – 10 Marks + External – 40 Marks

Aim of the Course

- 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.

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

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

OPEN COURSES

VCS5D01: Introduction to Computers & Office Automation

Course Number: XX
Contact Hours per Week: 2 (2T +0P)
Number of Credits: 2
Number of Contact Hours: 36 Hrs.
Course Evaluation: Internal – 10 Marks + External – 40 Marks

Aim of the Course

- To learn Office Automation.

Prerequisites

- Basic knowledge in Computer & Internet.

Course Outline

Module 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.

Module 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, Tables, Printing, Styles, linking and embedding object, Template.

Module III [7T]
Electronic Spread Sheet (OpenOffice Calc/MS-Excel) - Introduction to Spread Sheet, Creating & Editing Worksheet, Formatting and Essential Operations, Formulas and Functions, Charts.

Module 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.

Module V [7T]
Internet - History of Internet – Intranet and Extranet – DNS – Connections-Dial-up, ISDN, DSL, WiFi, Wireless, and Satellite Communications--E-mail, Chat, Forum, Blog, Browsers – Search Engines

References

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

VCS5D02: Web Designing

Course Number: XX
Contact Hours per Week: 2 (2T +0P)
Number of Credits: 2
Number of Contact Hours: 36 Hrs.
Course Evaluation: Internal – 10 Marks + External – 40 Marks

Aim of the Course
- To learn Web designing.

Prerequisites
- Basic knowledge in Computer & Internet.

Course Outline

Module I [9T]
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

Module II [9T]
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.

Module III [9T]
Javascript: Introduction and advantages of javascript, java script syntax, writing javascript in html, javascript operators, arrays and expressions, programming contructs - 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.

Module IV [9T]
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.

References

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

VCS5D03: Introduction to Problem Solving and C Programming

Course Number: XX
Contact Hours per Week: 2 (2T +0P)
Number of Credits: 2
Number of Contact Hours: 36 Hrs.
Course Evaluation: Internal – 10 Marks + External – 40 Marks

Aim of the Course

- 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

Module I [9T]
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.

Module II [9T]
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.

Module III [9T]
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.

Module IV [9T]
Array & Strings - One dimensional array, two dimensional array and multi- dimensional 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. E. Balaguruswami, Programming in ANSI C
2. Brian W. Kernighan & Dennis M. Ritchie, The C Programming Language
3. Yashvant P. Kanetkar, Let Us C
4. Byran Gotfried, Programming with C, Schaums Outline Series

ELECTIVE COURSES

VCS6E01: An Introduction to Operational Research

Course Number: 37
Contact Hours per Week: 4 (4T +0P)
Number of Credits: 3
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- To learn basics of Operational Research

Prerequisites

- Basic knowledge in Mathematics.
- Basic knowledge in Computer.

Course Outline

Unit I: [14T]
Operations Research (OR) and its Scope, Modeling in OR, Scientific Method in Operations Research, Scope and Limitations of Operational Research, Linear Programming: Definition, mathematical formulation, standard form, Solution space, solution –feasible, basic feasible, optimal, infeasible, multiple, redundancy, degeneracy, Standard and canonical form of LPP , Solution of LP Problems -Graphical Method, Simplex Method.

Unit-II: [14T]
Artificial variable techniques-Two Phase method; Charnes M Method, Degeneracy, Special cases in LPP. Duality in LP, Dual Simplex Method, Economic interpretation of Dual.

Unit-III: [15T]
Transportation Problem (T.P.) : Matrix form of T.P., the transportation table, Initial basic feasible solutions (different methods like North West corner, Row minima, Column minima, Matrix minima & Vogel's Approximation method ,MODI method for finding optimal solution for TP), Loops in T.P. table and their properties, Optimal solutions, Degeneracy in T.P., Unbalanced T.P.

Unit-IV: [15T]
Assignment Problem, Hungarian Method for Assignment Problem, Transhipment and Travelling salesmen problem. Meaning of combinatorial problems, The Branch and Bound technique, Branch and Bound Algorithm in case of assignment Problem.

Unit-V: [14T]
Network Analysis: Objectives; Steps involved; Significance; Basic Terms, Concepts of network, construction of network, Time estimates, CPM calculation, and PERT calculations.

References

1) An Introduction to Operational Research, C R Kothari
2) Operations Research ,R. Panneerselvam
3) Operations Research: Theory and Applications K Sharma

VCS6E02: System Software

Course Number: 37
Contact Hours per Week: 4 (4T +0P)
Number of Credits: 3
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

Module I [14T]
System software: General concept, Assemblers, loaders, linkers, macros, compilers, interpreters, operating system, Design of assemblers.

Module II [15T]
Macros and macro processors, Macro definitions and instructions, Macro calls, Features of Macros, Design of Macroprocessors

Module III [15T]
Loading, linking and relocating Loader schemes- Binders, linking loaders, overlays, dynamic binders-Dynamic loading and dynamic linking – Relocatability of programs.

Module IV [14T]
Compilers - Phases of a compiler - Lexical, syntax, intermediate code generation, optimization, code generation, symbol table and error correcting routines – Passes of a compiler.

Module V [14T]
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

VCS6E03: Microprocessor & Applications

Course Number: 37
Contact Hours per Week: 4 (4T +0P)
Number of Credits: 3
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course

- 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

Module I [14T]
16-Bit Microprocessor, 8086, Architecture, Pin Configuration, 8086 Minimum and Maximum mode configurations.

Module II [14T]
Addressing modes, 8086 Instruction set (Data transfer, Arithmetic, Branch, Processor control & String instruction), 8086 interrupts.

Module III [15T]
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.

Module IV [15T]
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).

Module V [`14T]
Advanced Microprocessors Introduction to 80186, 80286, 80386, 80486 and Pentium processors, General introduction to BIOS and DOS interrupts.

References

1. Barry.B.Brey, The Intel Microprocessor 8086/8088. 80186, 80286, 80386 and 80486 Architecture Programming and Interfacing, Prentice Hall of India Pvt.Ltd.1995.
2. K.R. Venugopal, Raj Kumar,Microprocessor, X86 programming, BPB publications, New Delhi
3. Abel P, IBM PC Assembly Language & Programming, Parson Education Asia
4. B Ram, Fundamentals of Microprocessors and Microcomputers, Dhanpat Rai Publications Pvt. Ltd., New Delhi
5. Mohamad Rafiquzzaman, Microprocessors and Microcomputer Based System Designing, Universal Bookstall, New Delhi
6. Yu. Cheng Liu, Glenn A Gibson, Microcomputer Systems The 8086/8088 Family. Architecture, Programming & Designing, Prentice Hall of India Pvt. Ltd., New Delhi
7. Ray A.K.Bhurchandi.K.M, Advanced Microprocessor and Peripherals, Tata McGrawHill.

LIST OF COMMON COURSES

VGC3CS11: BASIC NUMERICAL SKILLS

Contact Hours per Week: 4 (4T)
Number of Credits: 4
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Module I [14T]

Sets and set operation - Venn Diagrams - Elements of Co-ordinate system.
Matrices, Fundamental ideas about matrices and their operational rules – Matrix multiplication - Inversion of square matrices of not more than 3rd order- solving system of simultaneous liner equations.

Module II [14T]

Theory of equations: meaning, types of equations –simple linear and simultaneous equations (only two variables) eliminations and substitution method only. Quadratic equation factorization and formula method ($ax^2 + bx + c = 0$ form only) problems on business application.

Module III [15T]

Progressions: Arithmetic progressions finding the 'n'th term of an AP and also sum to 'n' terms of an AP.Insertion of Arithmetic means in given terms of AP and representation of AP. Geometric progression: finding nth term of GP. Insertion of GMs in given GP and also representation of GP - Mathematics of Finance - simple and compound interest. (Simple problems only)

Module IV [14T]

Meaning and Definitions of Statistics - Scope and Limitations – Statistical enquiries – Scope of the problem - Methods to be employed - types of enquiries - Presentation of data by Diagrammatic and Graphical Method - Formation of Frequency Distribution.

Module V [15T]

Measures of Central tendency - Arithmetic Mean, Median, Mode, Geometric and Harmonic mean, Measures of variation and standard, mean and quartile deviations - Skewness and Kurtosis and

Lorenz curve. Analysis of Time Series: Methods of Measuring - Trend and Seasonal variations - Index number - Unweighted indices - Consumers price and cost of living indices.
(Theory and problems may be in the ratio of 20% and 80% respectively)

References

1. Sundaresan and Jayaseelan - An Introduction to Business Mathematics and Statistical Methods
2. Dr. A K Arte & R V Prabhakar: A textbook of Business Mathematics.
3. Sanchethi and Kapoor, Business Mathematics.
4. Gupta S.P. - Statistical Methods
5. Navaneethan P. - Business Mathematics
6. Statistics - R.S.N. Pillai, Mrs. Bhagavathi
7. P.R. Vittal - Business Mathematics and Statistics

VGC3CS12 –GENERAL INFORMATICS

Contact Hours per Week: 4 (4T)
Number of Credits: 4
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Module I [14T]
Computers and Operating System**:** Features of New generation personal computers and peripherals-Computer Networks-Types of networks-Components of networks-Topology-internet-Uses of internet-Introduction to software-License-Open source-Overview of operating systems and major application software.

Module II [14T]
Information-IT and its Components-IT and Internet-IT Applications-E-Governance-Emerging Trends in IT-EDI-Mobile Computing-SMS,MMS-Wireless Applications-Blue Tooth-GPS-Infra Red Communication-Smart Card-DNA Computing-Cloud Computing

Module III [15T]
Data, Information and Knowledge-Knowledge Management –Internet as a Knowledge Repository-Academic Search Techniques-Academic Websites-Basic Concepts of IPR-Use of IT in Teaching and Learning-Academic service-INFLIBNET-NICENET-BRNET

Module IV [14T]
IT and Society-Issues and concerns-Digital Divide-Free Software Movement-IT and industry-Opportunities and Threats-Cyber Ethics-Security, Privacy Issues-Cyber Laws-Cyber Addictions-Information Overload-Guidelines for Proper Usage of Computers and Internet-E-waste & Green Computing-Unicode-IT & Regional-Languages

Module V [15T]
Introduction to Linux-Linux Distributions-Operating Systems & Linux –History of Linux and UNIX-Linux Software –Software Repositories-Linux office & Database Softwares-GNOME

Desktop Environment-GNOME Panels-GNOME Preferences-Desktop Publishing in Linux-Using Open Office.Org Suite

VGC3CS13: Fundamentals of Digital Electronics

Contact Hours per Week: 4 (4T)
Number of Credits: 4
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Module I [14T]

Number Systems and Codes, Decimal numbers, binary numbers, binary arithmetic, 1's and 2's complements, octal numbers, hexadecimal numbers, inter conversions of number systems, Digital codes: Binary coded decimal(BCD), Gray code, Excess-3 code, ASCII code, error detection and error correction codes, Hamming code.

Module II [15T]

Logic Gates Positive and negative logic, NOT gate, OR gate, AND gate, NAND gate, NOR gate, EX-OR and EX-NOR gates, Universal gates. Boolean Algebra: Boolean operations, logic expressions, rules and laws of Boolean algebra,DeMorgan's theorems, 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)

Module III [14T]

Arithmetic and Combinational Logic Circuits Half adder, full adder, parallel binary adder, decoders, BCD to 7-segment decoder, multiplexers and demultiplexers, multiplexer and demultiplexer trees.

Module IV [15T]

Sequential Logic Circuits: SR latch, SR flip flop, JK flip flop, Master Slave JK flip flop, D type flip flop, T type flip flop. Shift register: serial in - serial out, serial in – parallel out, parallel in - serial out, parallel in-parallel out configurations.

Module V [14T]

Ring counter, Johnson’s counter, asynchronous counters, synchronous counters, up/down asynchronous counter.

References:

1. Rajaraman V. & Radhakrishnan, An Introduction to Digital Computer Design, PHI.
2. Thomas L Floyd, Digital Fundamentals, Universal Book Stall
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

VGC4CS11: ENTREPRENEURSHIP DEVELOPMENT

Contact Hours per Week: 4 (4T)
Number of Credits: 4
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Module I [18T]
Entrepreneur and fundamentals of Entrepreneurship: Entrepreneurial competencies-Factors affecting entrepreneurial growth-Role of entrepreneur in economic Development-Challenges of Women Entrepreneurs.

Module II [18T]
Micro small and Medium Enterprises-Legal Framework-Licenses-Role of Promotional Institutions with Special Reference to KINFRA,KITCO,MSME & DICs-Concessions-Incentives and Subsidies.

Module III [18T]
Project Management-Feasibility and Viability analysis-Technical-Financial-Network-Appraisal and Evaluation-Project Report Preparation.

Module IV [18T]
Identification of Business Opportunities in the context of Kerala-Role of ED Clubs-Industrial Policies-Skill Development for Entrepreneurs. Business incubation-meaning-Setting up of Business incubation Centres.

VGC4CS12: BASICS OF AUDIO & VIDEO MEDIA

Contact Hours per Week: 4 (4T)
Number of Credits: 4
Number of Contact Hours: 72 Hrs.
Course Evaluation: Internal – 20 Marks + External – 80 Marks

Module I [18T]

Concept of wave motion-Speech, music and noise- Different types of noise-Concept of S.N.R-Mechanism of hearing of human ear.- Concept of stereophony- Nature of sound reflections, refractions, diffractions, absorption noise - general considerations on acoustics of studio reverberation, acoustics of auditorium - growth and decay of sound in enclosures, acoustic material.

Module II [18T]
Features and working principle of various types of microphones and loudspeakers. Basics of Digital Video Recording techniques -Principles of VCD, D.V.D and Blue ray Disc recorders and players -Introduction to Video compression techniques and standards (MPEG-1,2&4, H.26

standards)

Module III [18T]

Introduction to multimedia, Concept of hypertext/ hypermedia, applications of multimedia, multimedia authoring, multimedia hardware, Components of multimedia: text, audio, image, video, various file types: JPEG, MPEG audio, BITMAP, GIF, SVG, PNG, MIDI

Module IV [18T]

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

References:

1. Audio & Video Systems 2nd Edition- R.G Gupta-T.M.H
2. Standard handbook of Audio Engineering- Jerry Whitaker and Blair Benson- Publisher: McGraw-Hill Professional; 2nd Edition.
3. Modern recording Techniques 6th Edition-David Miles Huber & Robert E Runstein-Focal Press

4.Television Engineering and Video Systems-R.G Gupta-T.M.H

5.D. P. Mukherjee , Fundamentals Of Computer Graphics And Multimedia

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