File Ref.No.4180/GA - IV - J1/2012/CU

UNIVERSITY OF CALICUT

Abstract

Bachelor of Computer Application – under CUCBCSS UG 2014 – Scheme and Syllabus approved - implemented w.e.f 2014 admission onwards- Orders issued.

G & A - IV - J

U.O.No. 6833/2014/Admn Dated, Calicut University.P.O, 16.07.2014

Read:-1. U.O. No. 3797/2013/CU, dated 07.09.2013 (CBCSS UG Modified Regulations) (File.ref.no. 13752/GA IV J SO/2013/CU).

2. U.O. No. 5180/2014/Admn, dated 29.05.2014 (CBCSS UG Revised Regulations) (File.ref.no. 13752/GA IV J SO/2013/CU).

3. Item No: 1 of the minutes of BOS in Computer Science UG held on 28.1.14

4. Minutes of the meeting of Faculty of Science held on 3.2.14

5. Item II (i). of the minutes of Academic Council held on 20.3.14

ORDER

The Modified Regulations of Choice Based Credit Semester System for UG Curriculum w.e.f 2014 was implemented under the University of Calicut vide paper read as (1). The Revised CUCBCSS UG Regulations has been implemented w.e.f 2014 admission, for all UG programme under CUCBCSS in the University, vide paper read as (2).

As per The paper read as (3) the Scheme and Syllabus of Bachelor of Computer Application under Choice based Credit Semester System (UG) was approved by the Board of Studies in Computer Science UG. As per paper read as(4) above, the Faculty of Science has approved this minutes and as per paper read as 5 above ,the AC has approved it.

Sanction has therefore been accorded for implementing the Scheme and Syllabus of Bachelor of Computer Application with effect from 2014 admission onwards.

Orders are issued accordingly. Revised Syllabus appended.

Muhammed S
Deputy Registrar

To

1. All Affiliated Colleges/SDE/Dept.s/Institutions under University of Calicut.
2. The Controller of Examinations, University of Calicut.

3. The Director SDE, University of Calicut.

Forwarded / By Order

Section Officer

UNIVERSITY OF CALICUT

THENHIPALAM, CALICUT UNIVERSITY P.O

DEGREE OF
BACHELOR OF COMPUTER APPLICATION
(B. C. A)
(CHOICE BASED CREDIT AND SEMESTER SYSTEM)

UNDER THE

FACULTY OF SCIENCE

SYLLABUS

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

BOARD OF STUDIES IN COMPUTER SCIENCE (UG)
THENHIPALAM, CALICUT UNIVERSITY P.O
KERALA, 673 635, INDIA

JULY, 2014

Regulations
For Degree of
Bachelor Computer Applications
(B. C. A)
(Choice Based Credit and Semester System)

Effective from the Academic Year 2014-15

B.C.A. 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 (BCA) at University of Calicut, there is further educational opportunity to go for an MCA or other Master's Programme like MSc(CS), MSc(IT), MBA, etc., at this university or at any other University/Institute. Also after completing the BCA 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 BCA 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 BCA 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 BCA 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 BCA programme is 37

Credits: Each course shall have certain credits. For passing the BCA programme the student shall be required to achieve a minimum of 120 credits of which 38 (14 for common English courses +8 for common languages other than English + 16 credits for General courses) credit shall be from common courses, a minimum of 2 credits for project and 2 credits for the open course. Minimum credits required for core, complementary and open courses put together are 82.

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

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

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

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

Project: Every student of the BCA 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 BCA programme.

ADMISSION

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

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

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

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

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

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

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

REGISTRATION

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

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

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

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

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

COURSE EVALUATION

Total marks for each core, elective and open course, including lab courses and project evaluation cum programme viva voce, shall be 100 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 course (i.e., 20 marks), including lab and project evaluation cum programme viva voce, are for internal examinations.

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

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

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

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

Test paper (50%)	– 10 Marks

Attendance (25%)	– 5 Marks
Assignment/Lab involvement (25%)	– 5 Marks

Attendance of each course will be evaluated as below:

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

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

Punctuality	– 4 Marks
Use of Data	– 4 Marks
Scheme/Organization of Report	– 6 Marks
Viva-Voce	– 6 Marks

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

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

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

EXTERNAL EVALUATION:

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

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

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

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

1. **Section A**: 10 compulsory objective type questions (MCQ/fill in the blank/ matching/one word/etc) of 1 mark each (**Total 10 Marks**)

2. **Section B**: 5 compulsory short answer type questions of 2 Marks each(either a single question or can have subdivisions) (**Total 10 Marks**)

3. **Section C**: 5 short essay type questions of 4 Marks each, to be attempted from a set of 8 questions – at least one question from each unit (either a single question or can have subdivisions) (**Total 20 Marks**)

4. **Section D**: 5 long essay type questions of 8 Marks each, to be attempted from a set of 8 questions – at least one question from each unit (either a single question or can have subdivisions) (**Total 40 Marks**)

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

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

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

No practical examination will be conducted in odd semester. Practical examinations for B.C.A programme shall be conducted in the even semester 4 and 6.

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

1. Section A: One marked question of 30 Marks from Programming Lab Part A is to be attempted (Design - Algorithm/Flowchart/Interface: 10 Marks, Code: 10 Marks and Result: 10 Marks. **Total 30 Marks**)
2. Section B: One marked question of 30 Marks from Programming Lab Part B is to be attempted (Design - Algorithm/Flowchart/Interface: 10 Marks, Code: 10 Marks and Result: 10 Marks. **Total 30 Marks**)
3. Section C: Lab viva voce (**Total 10 Marks**)
4. Section D: Lab Record (**Total 10 Marks**)

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)**

REVALUATION:

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

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

IMPROVEMENT COURSE

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

Improvement and supplementary examinations cannot be done simultaneously.

EVALUATION AND GRADING

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

Indirect Grading System in 7 point scale is as below:

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

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

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

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

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

The Semester Grade Point Average can be calculated as

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

$$i.e., \qquad SGPA = \frac{C1 \times G1 + C2 \times G2 + C3 \times G3 + \cdots}{n}$$

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:

$$CGPA = \frac{Total\ credit\ points\ obtained\ in\ six\ semesters}{Total\ Credits}$$

GRADE CARD

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

a) Name of University

b) Name of college

c) Title of Under-Graduate Programme

d) Semester concerned

e) Name and Register Number of student

f) Code number, Title and Credits of each course opted in the semester

g) Internal marks, External marks, total marks, Grade point (G) and letter grade for each course in the semester

h) The total credits, total credit points and SGPA in the semester (corrected to two decimal places)

i) Percentage of total marks

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

AWARD OF DEGREE

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

GRIEVANCE REDRESSAL COMMITTEE

COLLEGE-LEVEL

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

UNIVERSITY-LEVEL

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

Curriculum for B.C.A Programme (2014 – 15 Academic Year Onwards – As Per the CUCBCSSUG 2013 Regulations)

Total Courses: 37	Total Credits: 120

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
I Semester	1	XXXXA01	Common English Course I	20	80	100	4	0	4	4
	2	XXXXA02	Common English Course II	20	80	100	4	0	4	3
	3	XXXXA03	Additional Language Course I	20	80	100	5	0	5	4
	4	BCA1B01	Problem Solving using C	20	80	100	2	2	4	3
	5	BCA1C01	Mathematical Foundation of Computer Applications	20	80	100	4	0	4	3
	6	BCA1C02	Discrete Mathematics	20	80	100	4	0	4	3
	Total (6 Courses)					**600**			**25**	**20**
II Semester	**7**	XXXXA03	Common English Course III	20	80	100	4	0	4	4
	8	XXXXA04	Common English Course IV	20	80	100	4	0	4	3
	9	XXXXA09	Additional Language Course II	20	80	100	5	0	5	4
	10	BCA2B02	Object Oriented Programming with C++	20	80	100	2	2	4	3
	11	BCA2C03	Computer Oriented Statistical Methods	20	80	100	4	0	4	3
	12	BCA2C04	Numerical Methods in C	20	80	100	4	0	4	3
	Total (6 Courses)					**600**			**25**	**20**

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
III Semester	**13**	XXXXA06	General Course I	20	80	100	4	0	4	4
	14	XXXXA12	General Course II	20	80	100	4	0	4	4
	15	BCA3B03	Database Design & RDBMS	20	80	100	3	2	5	3
	16	BCA3B04	Data Structures Using C++	20	80	100	2	2	4	3
	17	BCA3C05	Financial & Management Accounting	20	80	100	4	0	4	3
	18	BCA3C06	Operations Research	20	80	100	4	0	4	3
	Total (6 Courses)					**600**			**25**	**20**
IV Semester	19	XXXXA13	General Course III	20	80	100	4	0	4	4
	20	XXXXA14	General Course IV	20	80	100	4	0	4	4
	21	BCA4B05	Visual Programming Using C#.Net	20	80	100	5	0	5	3
	22	BCA4B06	Programming Laboratory I - Data Structures Using C++	20	80	100	0	2	2	2
	23	BCA4B07	Programming Laboratory II - RDBMS & C#.Net	20	80	100	0	2	2	2
	24	BCA4C07	E-Commerce	20	80	100	4	0	4	3
	25	BCA4C08	Management Information Systems	20	80	100	4	0	4	3
	Total (7 Courses)					**700**			**25**	**21**

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
V Semester	26	BCA5B08	Android Programming	20	80	100	3	1	4	4
	27	BCA5B09	Java Programming	20	80	100	2	4	6	4
	28	BCA5B10	Computer Networks	20	80	100	3	1	4	4
	29	BCA5B11	Computer Organization and Architecture	20	80	100	5	0	5	4
	30	BCA5B12	Microprocessor and Applications	20	80	100	3	1	4	3
	31	XXX5DXX	Open Course (Other Streams)	10	40	50	2	0	2	2
	Total (6 Courses)					**550**			**25**	**21**
VI Semester	32	BCA6B13	Web Programming	20	80	100	4	0	4	3
	33	BCA6B14	Software Engineering	20	80	100	4	0	4	3
	34	BCA6B15	Operating Systems	20	80	100	5	0	5	4
	35	BCA6B16	Programming Laboratory- III: Java & Web Programming	20	80	100	0	6	6	2
	36	BCA6B17	Project & Programme Viva Voce	10	40	50	0	2	2	2
	37	BCA6B18x	Elective	20	80	100	4	0	4	4
	Total (6 Courses)					**550**			**25**	18

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
Open Courses	XX	BCS5D01	Introduction to Computers & Office Automation	20	80	100	3	0	3	2
	XX	BCS5D02	Introduction to Web Designing	20	80	100	3	0	3	2
	XX	BCS5D03	Introduction to Problem Solving and C Programming	20	80	100	3	0	3	2
Electives	37	BIT6B18a	Computer Graphics	20	80	100	4	0	4	4
	37	BIT6B18b	Multimedia Systems	20	80	100	4	0	4	4
	37	BIT6B18c	Software Testing & Quality Assurance	20	80	100	4	0	4	4

	Mark Distribution	
1	Common: English (4 Courses×100 Marks)	400
2	Additional: Mal/Hindi... (2 Courses×100 Marks)	200
3	Core (Theory & Practical) (17 Courses×100 Marks)	1700
	Project	50
4	Open (1 Course)	50
5	Complementary (8 Courses×100 Marks)	800
Total Marks		**3600**
Total Courses		**37**
Total Credits		**120**

First Semester

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
I Semester	1	XXXXA01	Common English Course I	20	80	100	4	0	4	4
	2	XXXXA02	Common English Course II	20	80	100	4	0	4	3
	3	XXXXA03	Additional Language Course I	20	80	100	5	0	5	4
	4	BCA1B01	Problem Solving using C	20	80	100	2	2	4	3
	5	BCA1C01	Mathematical Foundation of Computer	20	80	100	4	0	4	3
	6	BCA1C02	Discrete Mathematics	20	80	100	4	0	4	3
	Total (6 Courses)					**600**			**25**	**20**

BCA1B01 – Problem Solving Using C

Course Number: 4

Contact Hours: 2T+2L

Number of Credits: 3

Number of Contact Hours: 32T+32L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To impart the students with the basic principles and skill in problem solving using computers.

Objectives of the Course:

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

Prerequisites: Background of the basic science at +2 level

Course Outline

UNIT I (6T)

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

UNIT II (6T+8L)

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

UNIT III (6T+8L)

Decision making, Branching and Looping: Decision making with IF statement, Simple IF statement, If.. .else statement, Nesting of if.. .else and else...if Ladder, switch statement, Conditional operator, go-to statement. Looping: while loop, do-while and for Loops, Nesting of loops, jumps in loop, skipping of loops.

UNIT IV (6T+8L)

Array & Strings: One dimensional array, two dimensional array and multi-dimensional array, strings and string manipulation functions. The Concept of modularization and User defined functions-Multi-function Program, calling functions, various categories of functions, Nesting of functions and recursion, functions and arrays, scope and life-time of variables in functions, multi-file programs. Structures & Union structure definition, giving values to members, structure initialization, comparison of structure variables, arrays of structures, arrays within structures, structures within arrays, structures and functions, Unions, bit-fields.

UNIT V (8T+8L)

Pointers: Understanding pointers, accessing the address of a variable, declaring and initializing pointers, accessing a variable through its pointer, pointer expressions, pointer and arrays, pointer and character string, pointers and functions, pointers and structures, pointer to pointer - dynamic memory allocation. Files: Defining, Opening and closing files - i/o operations on files - error handling on files, random access of files, command line operations. Pre-processor directives: Macro substitution directives - simple macros - macros with arguments - nesting of macros, Compiler control directives.

References:

1. *Programming in Ansi C*, E Balagurusamy, Tata McGraw Hill
2. *Programming with C*, Byran Gotfried
3. *Programming in C*, Kezningham & Ritchie
4. *Let us C*, Yashvant Kanetkar, BPB publications
5. *The spirit of C*, Mullish Cooper, Jasco books
6. *The Complete reference C*, Herbert Schildt, Tata Mc Graw Hill

BCA1C01 – Mathematical Foundation of Computer Applications

Course Number: 5

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To lay mathematical foundations for learning Computer Science.

Objectives of the Course:

- To learn the basic principles of linear algebra and vectors.
- To learn the basic principles of differential and integral Calculus
- To learn the mathematical modelling using ordinary and partial differential equations

Prerequisites: Mathematics at +2 level

Course Outline

UNIT I (12T)

Linear Algebra and Vector Calculus: Matrices: Matrix definition, order of a matrix, types of matrices, addition of matrices, multiplication of matrices, various kinds of matrices, transpose of a matrix, linear system of equations and solutions using gauss elimination, linear independence and rank, determinants, inverse, Eigen values. Vectors: Vectors in 2- and 3-space, dot and cross products,

UNIT II (12T)

Differentiation: Derivative at a point, Derivative of a Function, Differentiation from first principle, Differentiation of important functions, Product rule, Quotient rule, Differentiation of a function of a function (problem based), Higher order derivatives (Definition only)

UNIT III (12T)

Integration: Integral as Anti-derivative, Indefinite integral & constant of integration, Fundamental theorems, Elementary Standard results, Methods of Integration, Integration through Partial Functions, Integration by parts. Definite Integral: Evaluation by Substitution, Properties of definite integrals (Problem Based)

UNIT IV (12T)

Formation of differential equations, order and degree of the differential equation, Ordinary Differential Equations: First-Order ODE's, Solution of first order differential equations by separation of variables, Homogeneous first order differential equations and their solutions, linear first order differential equations and solutions.

UNIT V (12T)

Second Order Linear ODE's, linear second order differential equation with constant coefficients and solutions. Higher Order Linear ODE's, Partial Differential Equations.

References:

1. *Advanced Engineering Mathematics*, Erwin Kreyszig, Wiley
2. *Higher Engineering Mathematics*, John Bird, Elsevier Direct
3. *Skills in Mathematics: Algebra*, S.K.Goyal
4. *Higher Engineering Mathematics*, B S Grewal, Khanna Publishers
5. *Higher Engineering Mathematics*, Ramana, Tata McGraw Hill
6. *Engineering Mathematics*, P Kandasamy, S. Chand Group

BCA1C02 – Discrete Mathematics

Course Number: 6

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To equip the students with basic principles of Discrete Mathematics.

Objectives of the Course:

- To learn the mathematical logic & Boolean Algebra
- To learn the basics of Groups & Rings

Prerequisites: Background of the basic science at +2 level

Course Outline

UNIT I (12T)

Mathematical Logic: Propositions and logical operators, Truth tables, equivalence and implementation, Laws of logic, Quantifiers. Set theory: Introduction, concept of set of theory relation, types of relation, equivalence relation.

UNIT II (12T)

Boolean Algebra and its properties, Algebra of propositions & examples, De-Morgan's Laws, Partial order relations, greatest lower bound , least upper bound, Algebra of electric circuits & its applications. Design of simple automatic control system

UNIT III (12T)

Graph: Simple and multigraph, Incidence and degree, Isomorphism, Sub graphs and Union of graphs, connectedness, Walks, Paths and Circuits, Euler's Formula, Eulerian graph, Hamiltonian graph, Chromatic Graphs, Planer Graphs, Travelling salesman problem, Complete, Regular and Bipartite graphs, Directed Graphs

UNIT IV (12T)

Trees: Properties of trees, pendant vertices. Centre of a tree, rooted and binary trees, spanning trees, spanning tree algorithms, fundamental. circuits; spanning trees of a weighted graph: cutsets and cut-vertices; fundamental cutsets; connectivity and separativity; network. flows; max-flow min-cut theorem.

UNIT V (12T)

Plan on graphs, dual graphs, Kuratowski's two graph, matrix representation of graphs, incidence matrix, directed graphs, digraphs, directed paths and connectedness. Eular digraphs

References:

1. *Elements of Discrete Mathematics*, C. L. Liu, TMH Edition
2. *Discrete Mathematical Structures with applications to Computer Science*, J.K. Tremblay and R Manohar, McGraw Hill
3. *Discrete mathematical Structures*, Kolman, Busby, Ross, Pearson
4. *Graph theory*, Harry, F., Addison Wesley.
5. *Finite Mathematics*, S. Lipchutz, Schaum Series, MGH.
6. *Graph Theory*, Deo. N, PHI

Second Semester

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
II Semester	7	XXXXA03	Common English Course III	20	80	100	4	0	4	4
	8	XXXXA04	Common English Course IV	20	80	100	4	0	4	3
	9	XXXXA09	Additional Language Course II	20	80	100	5	0	5	4
	10	BCA2B02	Object Oriented Programming with C++	20	80	100	2	2	4	3
	11	BCA2C03	Computer Oriented Statistical Methods	20	80	100	4	0	4	3
	12	BCA2C04	Numerical Methods in C	20	80	100	4	0	4	3
	Total (6 Courses)					**600**			**25**	**20**

BCA2B02 – Object Oriented Programming with C++

Course Number: 10

Contact Hours: 2T+2L

Number of Credits: 3

Number of Contact Hours: 30T+30L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To equip the students with principles and concepts of object oriented design.

Objectives of the Course:

- To learn the basic concepts and principles of object oriented design
- To study C++ language

Prerequisites: Basic programming skill

Course Outline

UNIT I (6T+6L)

Introduction to Object Oriented Programming: Principles & Concepts of Object Orientation - Basic Principles of Object Orientation (Abstraction, Encapsulation, Modularity, Hierarchy, Typing, Concurrency, Persistence). Basic Concepts of Object Orientation (Object, Class, Attribute, Operation, State, Behaviour, Identity, Relationships/Association, Polymorphism, Message Passing).

Introduction to C++: Comments - Output operator - Input operator - Cascading of I/O operators. Tokens - keyword, identifiers, constants, strings and operators. Basic data types - User defined data types - Dynamic initialization of variables - Reference variables - Operators in C++ - Scope resolution operators - applications - Member dereferencing operators - Memory Management operators - new and delete. Control Structures - simple if, if else, nested if, switch, while do, break and continue statements.

UNIT II (6T+6L)

Functions: Introduction - Function Prototyping - Call by reference - Return by reference - Inline functions - Default arguments - Const arguments

Classes and Objects: Introduction - Limitations of C structures - Defining a class - Class Vs structures - Creating objects - Accessing class members - Defining member functions - Outside the class definition - Inside the class definition - Outside functions as inline - Nesting of member functions - Private member functions - Memory allocation for objects - Array of objects. Friendly functions.

UNIT III (6T+6L)

Constructors and Destructors: Basic Concepts of constructors - Default constructor - Parameterized constructor - Multiple constructors in a class - Constructor with default arguments - Dynamic initialization of objects - Copy constructor - Dynamic constructors - Destructors.

Function and Operator overloading: Introduction - Rules for overloading operators - Defining operator overloading - Overloading Unary operators - Prefix and Postfix operators overloading - Overloading Binary operators - Overloading relational operators - Overloading using friend functions - Overloading subscript operator. Function overloading.

UNIT IV (6T+6L)

Inheritance - Introduction - Defining derived classes - Types of inheritances - Single - Making a private member inheritable - Multilevel inheritance - Multiple inheritance - Hierarchical inheritance - Hybrid inheritance - Virtual base classes - Abstract classes - Constructors in derived classes - Nesting of classes - Containership

Virtual functions and Run time polymorphism - Introduction - Compile time and Runtime polymorphism - Pointers to objects - this pointer - Pointer to derived classes - Virtual functions - Rules for virtual functions - Pure virtual functions

UNIT V (6T+6L)

Streams: C++ stream classes - put() and get() functions - getline() and write() functions - Overloading << and >> operators - Formatted Console I/O operations - ios class functions - width(), precision(), fill(), setf() and unsetf() - Formatting flags - Manipulators - User defined manipulators.

Files: Introduction - Stream classes for files - Opening files using constructor - Opening files using open() - File modes - Detecting end of file - eof() - Sequential input and output - put() and get() - Reading and writing objects - read() and write() - Random Access files - Manipulating file pointers - seekg(), seekp(), tellg() and tellp() - Error handling during file operations - Command line arguments.

Templates: Generic programming, Class templates, Class templates with multiple parameters, Function templates, Overloading of template functions

References:

1. *The C++ programming language*, Bjarne Stroustrup, Addison Wesley

2. *C++ How to Program*, Deitel and Deitel, Pearson Education Asia
3. *Object oriented programming in C++*, Robert Lafore Galgotia
4. *Object Oriented Programming with C++*, E. Balagurusamy, TMH.
5. *Mastering C++*, K R Venugopal, Tata McGraw-Hill Publication.
6. *Object-Oriented Programming using C++*, B. Chandra, Narosa

BCA2C03 – Computer Oriented Statistical Methods

Course Number: 11

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To train the students with basic statistical methods.

Objectives of the Course:

- To learn the basics of statistics
- To learn probability theory
- To learn the sampling distributions

Prerequisites: Background of the basic Mathematics at +2 level

Course Outline

UNIT I (16T)

Basics statistics: Measures of central tendencies - Mean, Median, Mode, Geometric mean and Harmonic mean. Measures of dispersion - Range, quartile deviation, Lorenz curve. Mean deviation and standard deviation. Curve fitting- Principles of least squares, fitting of straight lines. Correlation (Bivariate case only) Pearson's coefficient of correlation. Rank correlation and Regression analysis.

UNIT II (16T)

Probability theory:Random experiment . Sample point, sample space, events, union, intersection and compliment of events. Different approaches of probability, frequency approach to probability, statistical regularity. Classical definition, numerical examples

UNIT III (16T)

Random variables and probability distribution, Discrete and continuous random variables- density function- distribution- density function- change of variable in unvariate case. Bivariate distributions- definition of bivariate distribution, marginal and conditional distributions, independence of two variables. Mathematical expectation- elementary properties, raw and central moments, moment generating functions, standard distributions- Binomial, Poisson, Normal

UNIT IV (16T)

Sampling distributions, the distribution of mean samples from a Normal population, Definition and statement of the form of the distributions- Chisquare and F and use of their tables

UNIT V (16T)

Estimation of parameters, Desirable properties of point estimates, Maximum likelihood estimator, Interval estimation, Interval estimates of mean and variance of Normal population and proportion of Binomial population, Testing of hypothesis, General principles of testing, Two types of errors, Neyman- Pearson approach

References:

1. *Introduction to Mathematical Statistics*, Hogg R V Craig A T, Macmillan
2. *Mathematical Statistics*, Freund J E, Waple R E, Prentice Hall of India.
3. *Probability and Statistics for Engineers*, Miller I Freund J E, Prentice Hall of India.
4. *Statistics for Management*, Levin R I, Prentice Hall of India

BCA2C04 -- Numerical Methods in C

Course Number: 12

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To train the students with basic Numerical Methods.

Objectives of the Course:

- To learn the floating point arithmetic
- To learn how to solve linear equations
- To learn the numerical differentiation and integration

Prerequisites: Background of the basic Mathematics at +2 level

Course Outline

UNIT I (12T)

Floating Point Arithmetic: Errors, Significant digits and Numerical Instability

UNIT II (12T)

Roots of Algebraic and Transcendental Equations - The Increment Search Method - Bisection Method - Method of False Position - Newton Raphson Method

UNIT III (12T)

Solution to Simultaneous Linear Equations - Direct Method - Crammer's Rule - Gauss Elimination Method - Gauss Jordan Elimination Method - Triangularization Method

UNIT IV (12T)

Interpolation and Approximation - Lagrange & Newton Interpolations - Finite Difference Operators, Interpolating

Polynomials using finite differences - Hermite Interpolation - Least Square Polynomial Approximation of a data

UNIT V (12T)

Numerical Differentiation and Integration - Numerical Differentiation - Methods based on finite differences - Extrapolation Methods - Numerical Integration - Methods based on Interpolation - Composite Rule - Trapezoidal and Simpson's Rule - Romberg Integration - Gauss Quadrature Formulas, Numerical Solution of Ordinary differential equations - Single Step Method - Taylor's Series Method - Euler's Method - Modified Euler's Method - Runge Kutta Methods

References:

1. *Numerical Methods in Engineering*, Salvadori & Baron, PHI
2. *Numerical Methods for Scientific and Engineering Computation*, M.KJain, SRK, Iyengar, R.K.Jain, New Age International

Third Semester

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
III Semester	**13**	XXXXA06	General Course I	20	80	100	4	0	4	4
	14	XXXXA12	General Course II	20	80	100	4	0	4	4
	15	BCA3B03	Database Design & RDBMS	20	80	100	3	2	5	3
	16	BCA3B04	Data Structures Using C++	20	80	100	2	2	4	3
	17	BCA3C05	Financial & Management	20	80	100	4	0	4	3
	18	BCA3C06	Operations Research	20	80	100	4	0	4	3
	Total (6 Courses)					**600**			**25**	**20**

BCA3B03 – Database Design & RDBMS

Course Number: 15

Contact Hours: 3T+2L

Number of Credits: 3

Number of Contact Hours: 48T+32L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To equip the students with principles and concepts of relational database design Objectives of the Course.

Objectives 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 the functional units computer of computers and their functioning along with basic programming knowledge

Course Outline

UNIT I (8T+6L)

Introduction: Purpose of database systems, View of data - Data abstraction, Instances and Schemas, Data models, Database languages, Database administrator, Database users, Database architecture. The Entity-Relationship model: Entity sets, Relationship sets, Attributes, Constraints, Mapping Cardinalities, Keys, ER diagrams, Weak entity sets, Strong entity sets.

UNIT II (8T+6L)

Relational Database Design: First, Second, Third, BCNF, Fourth and Fifth Normal forms. Transactions: ACID properties, States, Concurrent executions.

UNIT III (8T+6L)

Data Definition in SQL: Data types, Creation, Insertion, Viewing, Updation, Deletion of tables, Modifying the structure of the tables, Renaming, Dropping of tables. Data Constraints - I/O constraints, Primary key, foreign key, unique key constraints, ALTER TABLE command.

UNIT IV (8T+6L)

Database Manipulation in SQL: Computations done on table data: 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, Granting and revoking permissions: Granting privileges, Object privileges, Revoking privileges.

UNIT V (8T+6L)

Program 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. *Database System Concepts*, Abraham Silberschatz, Henry F Korth, S.Sudharshan
2. *PL/SQL: The Programming Language of Oracle SQL*, Ivan Bayross.
3. *SQL Bible*, Alex Krigel and Boris M.Trukhnov, Wiley pubs
4. *Microsoft SQL Server 2000 Bible*, Paul Nielsen, Wiley Dreamtech India Pubs.

BCA3B04 – Data Structures Using C++

Course Number: 16

Contact Hours: 2T+2L

Number of Credits: 3

Number of Contact Hours: 32T+32L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To train the students with the implementation of various data structures.

Objectives of the Course:

- To learn the contiguous and non-contiguous data structures and their implementation
- To learn linear and non-linear data structures and their implementation
- To learn the methods of searching, sorting and hashing techniques

Prerequisites: Programming skill in C++ language

Course Outline

UNIT I (6T+6L)

Algorithms (Analysis and Design): Problem solving - Procedure - Top-Down and Bottom-up approaches to algorithm design - Use of algorithms in problem solving: Developing an algorithm - Characteristics of algorithmic language - Design of algorithms - Implementation of algorithm - Verification of algorithm - Efficiency analysis of algorithms: Space, Time complexity, Frequency count - Simple algorithms. Data Representation: Abstract data type (ADT) - Fundamental and derived data types: Declaration - Representation - Primitive data structures: Symbol table - Recursion.

UNIT II (6T+6L)

Arrays: Definition - Terminology - One dimensional array - Memory allocation, Operations, Application - Multidimensional Arrays: Two dimensional Arrays - Sparse matrices - Three dimensional and n-dimensional Arrays - Pointer Arrays.

UNIT III (6T+6L)

Stacks: Introduction - Definition - Representation of stacks - Operations on stacks - Applications of stack. Linked List: Definition - Single Linked List: Representation, Operations - Circular Linked List - Double Linked List: Operations - Circular Double Linked List - Operations Application of Linked Lists: Sparse Matrix Manipulation - Polynomial Representation - Dynamic Storage Management - Memory Representation: Fixed, Variable block storage - Boundary tag system - Deallocation Strategy - Buddy System: Binary Buddy system.

UNIT IV (6T+6L)

Queues: Introduction - Definition - Representation of Queues - using Arrays, Linked list. - Various Queue structures: Circular Queue - De-queue - Priority Queue - Applications of Queues. Trees: Concepts - Representation of Binary tree - Operations on Binary Tree - Types of Binary Trees. Graphs: Introduction - Graph terminologies - Representation of Graphs - Operations on Graphs - Application of Graph Structures.

UNIT V (6T+6L)

Searching and Sorting: Searching - Sequential and Binary Search - Indexed Search - Hashing Schemes - Hashing functions: Division/ Remainder methods - Mid Square method - Folding method - Hash Collision: linear probing - Chaining - Bucketing - Sorting: Selection sort - Bubble sort - Insertion sort - Quick sort - Merge sort - Radix sort - Shell sort - Heap sort - Comparison of time complexity.

References:

1. *Classic Data Structures*, D. Samanta, PHI
2. *Data Structure Made Simple*, Sathish Jain, Shashi Singh, BPB
3. *Fundamentals of Data Structures*, E.Horowiz & S.Sahani, Galgotia
4. *Data Structure Using C and C++*, Aron M Tenenbaum.
5. *An Introduction to Data Structures with Applications*, Tremblay J.P and Sorenson P.G, TMH.
6. *Magnifying Data Structures*, Aprita Gopal, PHI Learning
7. *Data Structures & Algorithms*, R.S.Salaria, Khanna Book Publishing
8. *Data Structures using C and C++*, Y.Langsam et. al., PHI

BCA3C05 - Financial & Management Accounting

Course Number: 17

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To equip the students with fundamental principles of financial & management accounting.

Objectives of the Course:

- To get a general introduction on accounting and its general application.

- To get a general understanding on various tools for financial statement analysis.
- To get a general understanding on accounting procedures up to the preparation of various financial statements.
- To get a general understanding and important tools for managerial decision making.

Prerequisites: Basic Accounting knowledge

Course Outline

UNIT I (12T)

Principles of accounting - Some fundamentals concepts and conventions - Systems of accounting double entry principles - Advantages of Double entry system - personal, real, nominal accounts. Cash book - forms of cash books - subdivisions of Journal - Ledgers - limitations of financial accounting - Trial balance - Final accounts - Trading P/L A/c - Balance sheet

UNIT II (12T)

Invitation to management accounting: Analysis and interpretation of trading accounts and financial statements - Horizontal Vertical analysis - Common size Balance sheet - common size income statement - comparative income and balance sheet - trend analysis.

UNIT III (12T)

Ratio analysis: uses of ratios in interpreting trading accounts and financial statements - different types of ratios - Liquidity ratios - turnover ratios - activity ratios - solvency ratios

UNIT IV (12T)

Fund flow statement - schedule of changes in working capital - fund from operation - cash flow statement - cash from operating activities - cash from financing activities - cash from investing activities

UNIT V (12T)

Marginal costing - Breakeven point - cost volume profit analysis - margin of safety - standard costing - analysis of variance - material - labour - O/H - sales variables - Budget and Budgetary control -

different types of budgets - master budget - sales budget - production budget - flexible budget - cash budget - advantages - preparation

References:

1. *Financial Management*, Pandey I.M Vikas publishing house
2. *Elements of Accounting*, Kellock.J, Heinmann
3. *Advanced Accountancy*, S.N Maheshwari, Vikas Publishing
4. *Cost and Management Accounting*, A.Vinod, Calicut University Central Co-Operative Stores

BCA3C06 - Operations Research

Course Number: 18

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To impart an interdisciplinary approach in mathematical formulation of different models.

Objectives of the Course:

- To get a general introduction in solving linear programming problems.
- To get a general understanding of network analysis technique.
- To get a general understanding of different mathematical models.

Prerequisites: Basic Mathematical knowledge

Course Outline

UNIT I (12T)

Operation research and LPP: Operation Research and Decision making, Advantages of O.R approach in decision making, Application of O.R, uses and limitation of O.R. LPP: Introduction, mathematical formulation the problem, canonical and standard forms of LPP. Simplex method, artificial variable technique - Big *M* and two phase method - problem of degeneracy - concept of duality - dual simplex method.

UNIT II (12T)

Transportation model - North West corner rule, Least cost method, Vogel's approximation method - loops in transportation table - Degeneracy in transportation table - Transhipment problem. Assignment model: Mathematical formulation of the problem - assignment algorithm - impossible algorithms - travelling salesman problem

UNIT III (12T)

Network Scheduling: Concept of network, basic components, PERT and CPM, Rules of network construction, maximal flow problem, project scheduling critical path calculations, advantages of network (PERT/CPM). Sequencing models: processing *n* jobs through two machines, *n* jobs through three machines, two jobs through *m* machines

UNIT IV (12T)

Replacement model: Replacement of items with gradual deteriorates - items deteriorate with value of money, items that fail completely and suddenly, staff replacement problem.

UNIT V (12T)

Inventory model: Deterministic inventory problem - EOQ problem with no shortages, EOQ problem with no shortage and several production runs of unequal length, EOQ production problem with no shortages, EOQ problem with shortages, E)Q problem with one and two price break - ABC analysis.

References:

1. *Operation Research*, Kanti Swarup, Gupta P.K Man Mohan, Sultan Chand & Sons
2. *Operation Research: An Introduction*, Tahah. A, McMillan 1982
3. *Studies in The Mathematical Theory of Inventory and Production*,

Arrow K.J Karlin. S and Scarf, Stanford University Press

4. *Operation Research Methods and Problems*, Macrile Sasiani, Arthur Yospon and Lawrance Friedmon- John wiley&Sans. Inc

Fourth Semester

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
IV Semester	19	XXXXA13	General Course III	20	80	100	4	0	4	4
	20	XXXXA14	General Course IV	20	80	100	4	0	4	4
	21	BCA4B05	Visual Programming Using C#.Net	20	80	100	5	0	5	3
	22	BCA4B06	Programming Laboratory I - Data Structure Using C++	20	80	100	0	2	2	2
	23	BCA4B07	Programming Laboratory II - RDBMS & C#.Net	20	80	100	0	2	2	2
	24	BCA4C07	E-Commerce	20	80	100	4	0	4	3
	25	BCA4C08	Management Information Systems	20	80	100	4	0	4	3
	Total (7 Courses)					**700**			**25**	**21**

BCA4B05 -- Visual Programming Using C#.Net

Course Number: 21

Contact Hours: 5T

Number of Credits: 3

Number of Contact Hours: 75T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic knowledge in Visual Programming.

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

UNIT I (15T)

Getting started with .NET Framework: Benefits of .NET Framework, Architecture of .NET Framework, Components of .NET Framework: CLR, CTS, Metadata and Assemblies, .NET Framework Class Library, Windows Forms, ASP .NET and ASP .NE T AJAX, ADO .NET, Windows workflow Foundation, Windows Presentation Foundation, Windows Communication Foundation, Widows Card Space and LINQ.

UNIT II (15T)

Introducing C#: Need of C#, C# Pre-processor Directives, Creating a Simple C# Console Application, Identifiers and Keywords, Data Types, Variables an d Constants, Value Types, Reference Types, Type Conversions, Boxing and Undoing, Variables and Constants. Expression and Operators: Operator Precedence, Using the ?? (Null Coalescing) Operator, Using the :: (Scope Resolution) Operator and Using the is and as Operators. Control Flow statements: Selection Statements, Iteration Statements and Jump Statements, Namespaces, Classes, Objects and Structures: Namespaces, The System namespace, Classes and Objects: Creating a Class, Creating an Object, Using this Keyword, Creating an Array of Objects, Using the Nested Classes, Defining Partial Classes and Method, Returning a Value from a Method and Describing Access Modifiers. Static Classes and Static Class M embers. Properties: Read-only Property, Static Property, Accessibility of accessors and Anonymous types. Indexers, Structs: Syntax of a struct and Access Modifiers for structs.

UNIT III (15T)

Object- Oriented Programming: Encapsulation: Encapsulation using accessors and mutators, Encapsulation using Properties. Inheritance: Inheritance and Constructors, Sealed Classes and Sealed Methods, Extension methods. Polymorphism: Compile time Polymorphism/ Overloading, Runtime Polymorphism/ Overriding. Abstraction:

Abstract classes, Abstract methods. Interfaces: Syntax of Interfaces, Implementation of Interfaces and Inheritance.

UNIT IV (15T)

Delegates and Events and Exception Handling: Delegates: Creating and using Delegates, Multicasting with Delegates. Events: Event Sources, Event Handlers, Events and Delegates, Multiple Event Handlers. Exception Handling: The try/catch/finally statement, Checked and Unchecked Statements.

Graphical User Interface with Windows Forms: Introduction, Windows Forms, Event Handling: A Simple Event-Driven GUI, Visual Studio Generated GUI Code, Delegates and Event- Handling Mechanism, Another Way to Create Event Handlers, Locating Event Information. Control Properties and Layout, Labels, TextBoxes and Buttons, GroupBoxes and Panels, CheckBoxes and RadioButtons, ToolTips, Mouse-Event Handling, Keyboard-Event Handling. Menus, Month Calendar Control, Date TimePicker Control, LinkLabel Control, ListBox Control, CheckedListBox Control, ComboBox Control, TreeView Control, ListView Control, TabControl Control and Multiple Document Interface (MDI) Windows.

UNIT V (15T)

Data Access with ADO.NET: Understanding ADO.NET: Describing the Architecture of ADO.NET, ADO.NET, ADO.NET Entity Framework. Creating Connection Strings: Syntax for Connection Strings. Creating a Connection to a Database: SQL Server Database, OLEDB Database, ODBC Data Source. Creating a Command Object. Working with DataAdapters: Creating DataSet from DataAdapter, Paging with DataAdapters, Updating with DataAdapters, Adding Multiple Tables to a DataSet, Creating Data View. Using DataReader to work with Databases.

References:

1. *.NET Programming (6-in-1), Black Book*, Kogent Learning Solutions Inc., Wiely- Dream Tech Press.
2. *C# for Programmers*, Paul Deitel and Harvey Deitel, Pearson Education.
3. *Pro C# 5.0 and the .NET 4.5 Framework*, Andrew Trolsen, Wiely-

Appress.

4. *C# Unleashed*, Bart De Smet, Pearson Education- SAMS Series.
5. *Programming in C#*, Hebert Shildt, Tata McGraw Hill.
6. *Professional C#,* Christian Nagel, Bill Evgen, Jay Glynn Wrox Publications.
7. *Beginning with C#.Net*, Wroax publications
8. *C#, How to Program*, Dietel & Dietel, Pearson Education.
9. *Visual C#.Net*, John Sharp & John Jagger, PHI, New Delhi.
10. *Visual Studio .Net*, Francisco, Microsoft Publication
11. *.Net Framework Essentials*, O'Reilly

BCA4B06 – Programming Laboratory I: Data Structures Using C++

Course Number: 22

Contact Hours: 2L

Number of Credits: 2

Number of Contact Hours: 32L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with hands on experience on OOP and data structures

Objectives of the Course:

- To get implementational skill on OOP
- To get used to the implementation of various data structures

Prerequisites: Basic knowledge of OOP

Course Outline

Students are expected work in lab with an objective implementing the following tasks:

1. Simple C++ Programs to implement various Control Structures such as if, switch, do while, for, while, etc

2. Programs to understand Structure & Unions
3. Programs to understand Pointer Arithmetic
4. Programs to understand Functions & Recursion
5. Programs to understand Inline Functions
6. Programs to understand different function call mechanism such as Call by reference & Call by Value
7. Programs to understand Storage Specifiers
8. Use of Constructors & Destructors
9. Use of "this" Pointer
10. Programs to implement inheritance and function overriding such as multiple inheritance and hierarchical inheritance
11. Programs to overload unary & binary operators as member function & non-member function
12. Programs to understand friend function & friend class
13. Programs on Class Templates
14. Operation on dynamic array such as – Creation – Passing to function – Insertion Implementation – Delete Implementation – Search Implementation – Sort Implementation – Separation implementation – Merge Implementation.
15. Operation on linked list such as – Creation – Passing to function – Insertion Implementation – Delete Implementation – Search Implementation – Sort Implementation
16. Operation on doubly linked list such as – Creation – Passing to function – Insertion Implementation – Delete Implementation – Search Implementation – Sort Implementation – Separation implementation – Merge Implementation
17. Implementing basic operation of stack (push, pop) using array implementation
18. Implementing basic operation of stack (push, pop) using linked list implementation
19. Implementing basic operation of Queue (Enqueue, Dequeue) using array implementation
20. Implementing basic operation of Queue (Enqueue, Dequeue) using linked list implementation

21. Implement Binary tree traversal methods: Preorder, In-order, Postorder traversal. Recursive Algorithms for above mentioned Traversal methods
22. Implementing Binary search tree operation (search, addition, deletion).
23. Implementing various searching and sorting techniques

BCA4B07 – Programming Laboratory II: RDBMS & C#.Net

Course Number: 23

Contact Hours: 2L

Number of Credits: 2

Number of Contact Hours: 32L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with hands on experience on OOP and data structures

Objectives of the Course:

- To get implementational skill on RDBMS
- To get implementational skill on .NET environment

Prerequisites: Basic knowledge of .NET and RDBMS

Course Outline

Students are expected work in lab with an objective implementing the following tasks:

RDBMS

1. SQL*Plus/MySQL and SQL: (a). Introduction (b). Logging on to SQL*Plus/MySQL and Leaving SQL*Plus/MySQL (c). Choosing and Describing Tables (d). Elements of the SQL Query (e). Editing SQL Statements (f). The System Dummy Table (g). Selecting Columns (h). Duplicate Information (DISTINCT) (i). Sorting Information

2. SQL Functions: (a). The Concatenation Operator (b). Elements of the SQL Query: Arithmetic (c). Column Aliases (d). String Functions (e). Arithmetic Functions (f). Date Functions (g). Mixed Functions (h). Operator precedence
3. Advanced SQL Functions: (a). Nesting Different Functions (b). Decode Crosstab (c). Decode with ">", "<" & "=" (d). Select with Minus Union and Intersect (e). Handling NULL
4. Filtering Data Using Where: (a). Where Operators (b). Where with Keywords (c). Where and Logical Operators (d). Where and Soundex
5. Retrieving Data from Multiple Tables: (a). Joining Tables (Equi-Joins) (b). Aliases for Table Names (c). Joining Tables (Non-Equi-Joins) (d). Joining Tables (Outer Joins) (e). Joining Tables (Inner Joins) (f). Virtual table
6. Group By and Group By Functions: (a). Group Function Examples (b). Group Function with Having
7. Sub-Queries: (a). Basic Subqueries (b). Multiple Column Subqueries (c). Subqueries with Having (d). Correlated Subqueries
8. Data Definition Language (DDL): (a). Create, Drop Alter Keywords (b). Tables (c). Column (d). Views (e). Synonyms (f). Sequences (g). Object (h). Alter table
9. Integrity Constraints: (a). Types of Constraint (b). Referential Integrity (c). Defining Constraints (d). Integrity Constraints and Data Dictionary (e). Disabled constraints
10. Indexes: (a). Create Index (b). Unique Option (c). When and What to Index (d). Drop Index (e). Validate Index (f). Index Type Overview
11. Data Manipulation Language (DML): (a). Insert (b). Update (c). Delete (d). OPS Commands (Commit, Rollback and Savepoints) (e). Locking tables
12. Data Control Language (DCL): (a). Data Security (b). Grant and Revoke (c). Session control statements (d). System control statements

C#.NET

1. Program in C# to demonstrate different kinds of arrays including

jagged arrays.

2. Program in C# to demonstrate boxing and unboxing
3. Program in C# to implement stack operations
4. Program to demonstrate operator overloading
5. Using try, catch and finally blocks- program in C# to demonstrate error handling
6. Demonstrate use of virtual and override keyword in C# with a simple program.
7. Implement linked lists in C# using the existing collections name space.
8. Program to demonstrate abstract class and abstract methods in C#
9. Program in C# to build a class which implements an interface which already exists
10. Demonstrate arrays of interface types with a C# program
11. Program in C# to override a method
12. Program in C# to ask a user to enter a choice to add, delete, modify or view address using methods for each functionality.
13. Program in C# to demonstrate and verify that the static constructor runs only one time, even though two instances of Class are created, and that it runs before the instance constructor runs.
14. Program in C# to create read only properties
15. Program to create a write only property
16. Program to demonstrate that despite of the internal implementation of the class, its data can be obtained consistently through the use of indexers
17. Program in C# to show that when a struct is passed to a method, a copy of the struct is passed, but when a class instance is passed, a reference is passed
18. Program to implement an Interface
19. Program in C# to perform conversions between enums and their base types
20. Program to string manipulations and calling methods directly by

using delegates

21. Program to invoke an event when a list of documents is changed
22. Program to implement a three-valued logical type with operator overloading. The possible values of this type are DBBool.dbTrue, DBBool.dbFalse, and DBBool.dbNull, where the dbNull member indicates an unknown value.
23. Program to display attributes of two classes order and accounts with the implementation of Author attribute class.
24. Program in C# in which a derived communicate with base class during instantiation.
25. Program in C# to create a base class shape and derived classes i.e., Rectangle, Circle, and Triangle. Invoke the method from base class shape using polymorphism
26. Program in C# to open a file to write and read and handle the exception
27. Program in C# to create an object of type Alpha (oAlpha) and a thread (oThread) that references the Beta method of the Alpha class. Start the thread. The program should wait until the thread is initialized and also stops the execution for milliseconds.
28. Program in C# to create and use a thread pool
29. Program to implement a collection class used with for each.
30. Program in C# to which provides type safety in C# while maintaining interoperability with other languages
31. Program to use indexed properties to perform some text operations on object of a class.
32. Program in C# to using .NET Framework calls to deny the UnmanagedCode permission i.e. for imperative security.
33. Program in C# to use attributes for the security permissions.i.e. for Declarative Security.
34. Program in C# to check for the unmanaged code permission is executed once at load time, rather than upon every call to the unmanaged method.

BCA4C07 -- E-Commerce

Course Number: 24

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Aim of the Course: To provide the students with the basic knowledge in E-Commerce

Objectives of the Course:

- To get a general introduction Electronic Commerce framework
- To get a general understanding on various electronic payment system
- To get a general understanding on Internal information systems
- To get a general understanding on the new age of Information

Prerequisites: Basic knowledge of Commerce

Course Outline

UNIT I (12T)

History of E-commerce and Indian Business Context : E-Commerce, Emergence of the Internet, Emergence of the WWW, Advantages of E-Commerce, Transition to E-Commerce in India, The Internet and India, E-transition Challenges for Indian Corporates. Business Models for E-commerce: Business Model, E-business Models Based on the Relationship of Transaction Parties - E-business Models Based on the Relationship of Transaction Types.

UNIT II (12T)

Enabling Technologies of the World Wide Web: World Wide Web, Internet Client-Server Applications, Networks and Internets, Software Agents, Internet Standards and Specifications, ISP, e-Marketing: Traditional Marketing, Identifying Web Presence Goals, Online Marketing, E-advertising, E-branding.

UNIT III (12T)

e-Security : Information system Security, Security on the Internet, E-business Risk Management Issues, Information Security Environment in India. Legal and Ethical Issues: Cyberstalking, Privacy is at Risk in the Internet Age, Phishing, Application Fraud, Skimming, Copyright, Internet Gambling, Threats to Children.

UNIT IV (12T)

e-Payment Systems: Main Concerns in Internet Banking, Digital Payment Requirements, Digital Token-based e-payment Systems, Classification of New Payment Systems, Properties of Electronic Cash, Cheque Payment Systems on the Internet, Risk and e-Payment Systems, Designing e-payment Systems, Digital Signature, Online Financial Services in India, Online Stock Trading.

UNIT V (12T)

Information systems for Mobile Commerce: What is Mobile Commerce?, Wireless Applications, Cellular Network, Wireless Spectrum, Technologies for Mobile Commerce, Wireless Technologies, Different Generations in Wireless Communication, Security Issues Pertaining to Cellular Technology. Portals for E-Business: Portals, Human Resource Management, Various HRIS Modules

References:

1. *E-Commerce - An Indian Perspective*, P.T.Joseph, S.J., PHI
2. *E-Commerce Strategy, Technologies and Applications,* David Whiteley, Tata Mc-Graw-Hill
3. *Frontiers of Electronic Commerce*, Ravi Kalakota, Andrew B. Whinston, Pearson Education Asia
4. *E – Commerce*, Jeffery F. Rayport, Bernard J. Jaworski, TMCH
5. *E-Commerce - A Managerial Perspective*, P.T. Joseph, PHI

BCA4C08 -- Management Information Systems

Course Number: 25

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic knowledge in Management Information Systems

Objectives of the Course:

- To geta general introduction to Information Systems
- To geta general understanding on the conceptual foundations
- To geta general understanding on organizational & management concepts
- To geta general understanding on developing & implementing application systems

Prerequisites: Basic knowledge in Information Systems

Course Outline

UNIT I (12T)

Introduction to information Systems: Definition of a management information system, MIS as an evolving concept, MIS & other academic disciplines, Subsystems of an MIS, Operating elements of an information system, Management information system support for decision making, MIS structure based on management activity, MIS structure based on organizational function, synthesis of MIS system structure, some issues of MIS

UNIT II (12T)

Conceptual Foundations: Phases in decision Making Process, Concepts of decision Making, Behavioural models of the decision Maker, Behavioural Model of organizational decision making, decision making under psychological stress, Methods for decision among alternatives, relevance of decision making concepts for information system design, Definition of information, quality of information in decision making, value of information other than in decision, General model of the human as an information processor, The Newwell-Simon model, tentative limits on human information processing, Concepts of human cognition & learning, Characteristics of human information processing performance.

UNIT III (12T)

System Concepts: Definition of a System, General model of a system, Types of systems, Subsystems, System concepts & Organizations, System concepts applied to MIS, Concepts of organizational Planning, Planning Process and Characteristics of control process.

UNIT IV (12T)

Organizational structure & management concepts: The basic model of Organizational Structure, Modifications of basic organizational structure, Information processing model of organization structure, Organizational culture & Power, Organizational change, Management theories, organizations as sociotechnical systems, implications of organizational structure & management theory of MIS.

UNIT V (12T)

Developing & implementing application systems: A Contingency approach to choosing an application development strategy, Prototyping approach to application system development, Life cycle approach to application system development, Life cycle definition stage, Life cycle installation & operation stage, Implementation of IS as an organizational change process, Quality in IS, Organizational functions for control & quality assurance, Quality assurance for applications, Quality assurance with user developed systems, Post audit evaluation of Is Applications, Evaluation of existing Hardware & Software, Evaluation of Proposed Hardware & Software, Auditing of IS.

Reference:

1. *Management Information Systems Conceptual Foundations, Structure And Development*, Gordon B Davis, Margrethe H Olson, Tata McGraw Hill

Fifth Semester

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
V Semester	26	BCA5B08	Android Programming	20	80	100	3	1	4	4
	27	BCA5B09	Java Programming	20	80	100	2	4	6	4
	28	BCA5B10	Computer Networks	20	80	100	3	1	4	4
	29	BCA5B11	Computer Organization and Architecture	20	80	100	5	0	4	4
	30	BCA5B12	Microprocessor and Applications	20	80	100	3	1	4	3
	31	XXX5DXX	Open Course (Other Streams)	10	40	50	2	0	3	2
	Total (6 Courses)					**550**			**25**	**21**

BCA5B08 – Android Programming

Course Number: 26

Contact Hours: 3T+1L

Number of Credits: 4

Number of Contact Hours: 45T+12L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To familiarize the students with the basic concepts of Android Programming.

Objectives 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 OOP & Java Programming.

Course Outline

UNIT I (9T+3L)

Introducing the android computing platform, History of android, android 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.

UNIT II (9T+3L)

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

UNIT III (9T+3L)

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.

UNIT IV (9T+3L)

Android menus - creating menus, working with menu groups, respond- ing 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 ac- tivity, implementing base activity classes, tabbed action bar and tabbed listener, debug text view layout, action bar and menu interaction, lsit 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.

UNIT V (9T+3L)

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.

BCA5B09 – Java Programming

Course Number: 27

Contact Hours: 2T+4L

Number of Credits: 4

Number of Contact Hours: 30T+60L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic programming skill in Java.

Objectives 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: Basic knowledge in OOP

Course Outline

UNIT I (6T+12L)

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.

UNIT II (6T+12L)

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, Deprecation, 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.

UNIT III (6T+12L)

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.

UNIT IV (6T+12L)

Collections & Database Connectivity Collections: Vectors, Hashtables, Enumerations, Properties, Collection, Framework Hierarchy, Lists, Sets, Maps, The Collections Class. Networking: Networking Fundamentals, The Client/Server Model, In- etAddress, 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.

UNIT V (6T+12L)

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. *Java Complete Reference*, Herbert Schildt, Tata McGraw hill edition.
2. *J2EE Complete Reference*, Jim Keogh, Tata McGraw hill edition.
3. *Java Enterprise in a Nutshell*, David Flanagan, Jim Farley, William Crawford & Kris Mangnusson, OReill.
4. *Programming With Java - A Primer*, E Balagruswami, Tata Mc Graw Hill,2008

BCA5B10 – Computer Networks

Course Number: 28

Contact Hours: 3T+1L

Number of Credits: 4

Number of Contact Hours: 45T+12L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic knowledge in Networking.

Objectives of the Course:

- To get a general introduction to Computer Networks
- To get a general understanding on different OSI layers

Prerequisites: Basic knowledge in Communication Systems

Course Outline

UNIT I (9T+3L)

Introduction to Computer networks, Topology, categories of networks, Internetwork, Internet, Network Models, Layered model, OSI and TCP/IP models, Physical layer, Switching - Circuit switching, Packet Switching and Message Switching, DTE - DCE

Interface, EIA - 232 interface, X.21 modems.

UNIT II (9T+3L)

Data link layer, Error detection and correction, Types of errors, Single bit 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

UNIT III (9T+3L)

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.

UNIT IV (9T+3L)

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.

UNIT V (9T+3L)

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. *Introduction to Data Communications & Networking*, Behurouz & Forozan,TMH
2. *Computer Networks*, Andrew S. Tanenbaum, PHI
3. *Data and Computer Communications*, William Stallings, VIIth Edition, Pearson Education
4. *Cryptography and Network Security*, Principles and Practices-William Stallings, Prentice Hall of India.

BCA5B11 -- Computer Organization & Architecture

Course Number: 29

Contact Hours: 5T

Number of Credits: 4

Number of Contact Hours: 75T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic knowledge on Computer organization and architecture.

Objectives of the Course:

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

Prerequisites: Basic knowledge of Computer

Course Outline

UNIT I (15T)

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.

UNIT II (15T)

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.

UNIT III (15T)

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

UNIT IV (15T)

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.

UNIT V (15T)

Parallel Processing: Basic Parallel Processing Architecture - Taxonomy- SISD. MISD, SIMD, MIMD structures - CISC vs RISC - Symmetric Multi-processors - 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. *Computer System Architecture*, M. Morris Mano, Prentice-Hall of India, Pvt. Ltd
2. *Computer Organization and Architecture,* William Stallings Prentice- Hall of India, Pvt. Ltd., Seventh edition, 2005.

BCA5B12 -- Microprocessor & Applications

Course Number: 30

Contact Hours: 3T+1L

Number of Credits: 3

Number of Contact Hours: 45T+12L

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic knowledge on Microprocessors and its applications.

Objectives 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

UNIT I (9T+3L)

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

UNIT II (9T+3L)

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

UNIT III (9T+3L)

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.

UNIT IV (9T+3L)

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).

UNIT V (9T+3L)

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

References:

1. *The Intel Microprocessor 8086/8088. 80186, 80286, 80386 and 80486 Architecture Programming and Interfacing,* Barry.B.Brey, Prentice Hall of India Pvt.Ltd.
2. *Microprocessor X86 programming*, K.R. Venugopal, Raj Kumar, BPB publications
3. *IBM PC Assembly Language & Programming,* Abel P, Parson Education Asia 2001
4. *Fundamentals of Microprocessors and Microcomputers, B* Ram, Dhanpat Rai Publications Pvt. Ltd., New Delhi
5. *Microprocessors and Microcomputer Based System Designing,* Mohamad Rafiquzzaman, Universal Bookstall, New Delhi
6. *Microcomputer Systems: The 8086/8088 Family. Architecture, Programming & Designing,* Yu. Cheng Liu, Glenn A Gibson, Prentice Hall of India Pvt. Ltd., New Delhi
7. *Advanced Microprocessor and Peripherals,* Ray A.K., Bhurchandi. K.M, Tata McGraw-Hill, 2002.

Sixth Semester

Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credit
				Intern	Extern	Total	Theory	Lab	Total	
VI Semester	32	BCA6B13	Web Programming	20	80	100	4	0	4	3
	33	BCA6B14	Software Engineering	20	80	100	4	0	4	3
	34	BCA6B15	Operating Systems	20	80	100	5	0	5	4
	35	BCA6B16	Programming Laboratory- III: Java & Web Programming	20	80	100	0	6	6	2
	36	BCA6B17	Project & Programme Viva Voce	10	40	50	0	2	2	2
	37	BCA6BXX	Elective	20	80	100	4	0	4	4
	Total (6 Courses)					**550**			**25**	18

BCA6B13 – Web Programming

Course Number: 32

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic skill in Web programming.

Objectives 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: Basic programming knowledge

Course Outline

UNIT I (12T)

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

UNIT II (12T)

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

UNIT III (12T)

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.

UNIT IV (12T)

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.

UNIT V (12T)

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. *Web Programming with HTML, XHTML, CSS,* Jon Duckett, Wrox.
2. *PHP & MySQL Bible,* Jim Converse & Joyce Park, Wiley.
3. *Internet & World Wide Web How To Program*, Deitel, Harvey M. and Paul J.
4. *HTML 4.0 in Simple Steps*, Kogent Solutions, Wiley
5. *HTML 4 for Dummies*, Ed Tittel & Mary Burmeis- Ter, Wiley
6. *Beginning PHP*, D W Mercer, A Kent, S D Nowicki, Wrox
7. *PHP & MYSQL for Dummies*, Janet Valad, Wiley

BCA6B14 – Software Engineering

Course Number: 32

Contact Hours: 4T

Number of Credits: 3

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic software development principles and skills.

Objectives 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: Basic programming knowledge

Course Outline

UNIT I (12T)

Introduction, Software Engineering Discipline, Evolution and Impact, Programs Vs Software Products, Emergence of Software Engineering, Changes in Software Development Practices, Computer Systems Engineering. Software Life Cycle Models: Use of a Life Cycle Models, Classical Waterfall Model, Iterative Waterfall Model, Prototyping Model, Evolutionary Model, Spiral Model. Software Project Management: Responsibilities of a Software Project Manager, Project Planning, Metrics for Project Size Estimation, Project Estimation Techniques, COCOMO, A Heuristic Estimation Technique, Staff Level Estimation, Scheduling, Organization and Team Structures, Staffing, Risk Management, Software Configuration Management.

UNIT II (12T)

Requirements Analysis and Specification: Requirements Gathering and Analysis, Software Requirements Specification (SRS), Formal System Development Techniques. Software Design: Characteristics of a Good Software Design, Cohesion and Coupling, Neat Arrangement, Software Design Approaches, Object-Oriented Vs Function, Oriented Design.

UNIT III (12T)

Function-Oriented Software Design: Overview of SA/SD Methodology, Structured Analysis, Data Flow Diagrams(DFDs), Structured Design, Detailed Design, Design Overview. Object Modelling Using UML: Overview of Object-Oriented Concepts, UML, UML Diagrams, Use Case Model, Class Diagrams, Interaction Diagrams, Activity Diagrams, State Chart Diagram. Object-Oriented Software Development: Design Patterns, Generalized OOAD Process.

UNIT IV (12T)

User Interface Design: Characteristics of a User Interface, Basic Concepts, Types of User Interfaces, Component-Based GUI Development, User Interface Design Methodology. Coding and Testing: Coding, Code Review, Testing, UNIT Testing, Black-Box Testing, White-Box Testing, Debugging, Program Analysis Tools, Integration Testing, System Testing

UNIT V (12T)

Software Reliability and Quality Management: Software Reliability, Statistical Testing, Software Quality, Software Quality Management System, ISO 9000, SEI Capability Maturity Model. Computer Aided Software Engineering: CASE Environment, CASE support in Software Life Cycle, Characteristics of CASE Tools, Second Generation CASE Tool, Architecture of a CASE Environment. Software Maintenance: Characteristics of Software Maintenance, Software Reverse Engineering, Software Maintenance Process Models, Estimation of Maintenance Cost. Software Reuse: Introduction, Issues in any Reuse Program, Reuse Approach, Reuse at Organization Level.

References:

1. *Fundamentals of Software Engineering*, Rajib Mall, Prentice Hall of India Private Limited

2. *An Integrated Approach to Software Engineering*, Pankaj Jalote, Narosa Pub.
3. *Software Engineering - A Practical Approach*, Roger S. Pressman McGraw Hill - International Ed.
4. *Software Engineering*, Ivan Somervelli.

BCA6B15 – Operating System

Course Number: 34

Contact Hours: 5T

Number of Credits: 4

Number of Contact Hours: 75T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic concepts of Operating Systems.

Objectives 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: Basic knowledge in data structures

Course Outline

UNIT I (15T)

What is an OS, Functions, Structure, Types: Batch, Multiprogramming, Timesharing, Real time, Multiprocessor system, Distributed system, OS as Resource manager, Booting process, POST.

UNIT II (15T)

Processor Management: Functions, Process, Process states, State transition, PCB, Events related to process, Process scheduling, Scheduling objectives, Scheduling levels, Pre-emptive and non- pre-emptive scheduling algorithms, Concurrent processes, Process synchronization, Mutual exclusion and critical section, Solution to mutual exclusion problem: Software, Hardware & Semaphore Solutions, Classical problems of mutual exclusion, Deadlock: Handling deadlock, Prevention, Avoidance, Detection and Recovery.

UNIT III (15T)

Memory Management: Functions, Contiguous: State and Dynamic, Non-contiguous: Segmentation and Paging, Virtual memory, Demand paging, Page replacement policies, Working Set principle.

UNIT IV (15T)

File Management: Information management: File system, Functions, File directory, File system structure, File system design: Symbolic, Basic, Logical and Physical file system layers, File organization, File allocation, Free space management, File protection and security.

UNIT V (15T)

Device Management: Disk scheduling, Disk scheduling policies, Device management: Functions, Techniques for device management: Dedicated, Shared, Virtual, Spooling, Channels and Control unit.

References:

1. *An Introduction to Operating System*, Dietel, Addison Wesly
2. *Operating System*, Madnick S.E., Donovan J.J., McGraw Hill
3. *William Stallings*, Operating System, PHI
4. *System Programming and Operating Systems*, D.M.Dhamdhare, Tata McGraw Hill, 1996
5. *Modern Operating Systems*, Tanenbaum A.S., Prentice Hall Additional References:
6. *Operating System Concepts*, Silberschatz, Galvin & Gagne, John Wiley & Sons

7. *Operating Systems*, Madnick E., Donovan J., Tata McGraw Hill,2001
8. *Operating Systems - A design Oriented Approach*, Charles Crowley, Tata McGraw Hill

BCA6B16 -- Programming Laboratory III: Java & Web Programming

Course Number: 35

Contact Hours: 6L

Number of Credits: 2

Number of Contact Hours: 90T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide hands on experience in Java and PHP programming.

Objectives of the Course:

- To provide hands on experience in Java programming.
- To provide hands on experience in PHP programming.

Prerequisites: Basic programming knowledge in Java and PHP

Course Outline

Programming with JAVA: Lab Exercises

1. Programs to demonstrate the usage of all primitive data types and operators of Java
2. Programs to demonstrate the usage of control statements in Java
3. Programs to demonstrate the usage of arrays in Java
4. Programs to demonstrate the usage of command line arguments
5. Programs to demonstrate the usage of constructors
6. Programs to demonstrate the usage of call by value and call by reference
7. Programs to demonstrate the usage of 'this' operator. Also use

the 'this' keyword as return statement.

8. Programs to demonstrate the usage of to static variables, methods and blocks.
9. Programs to demonstrate the reuse class.
10. Programs to demonstrate the usage of method overriding concepts.
11. Programs to demonstrate the usage of 'super' keyword.
12. Programs to demonstrate the usage of abstract class.
13. Programs to demonstrate the usage of interface
14. Programs to demonstrate the usage of multiple inheritance
15. Programs to demonstrate the usage of recursion
16. Programs to demonstrate the usage of package
17. Programs to demonstrate the usage of automatic type conversions apply to overriding.
18. Programs to demonstrate the usage of try and catch block.
19. Programs to demonstrate the usage of multiple catch statements
20. Programs to demonstrate the sub class exception precedence over base class
21. Programs to demonstrate the usage of try/catch with finally clause
22. Programs to demonstrate the usage of throws clause
23. Program for creation of user defined exception
24. Program to create a text file and check whether that file is exists.
25. Program to rename the given file, after renaming the file delete the renamed file. (Accept the file name using command line arguments.)
26. Program to create a directory and check whether the directory is created
27. Program to open one application using process class
28. Program using modifiers
29. Program to illustrate creation of threads using runnable class.
30. Program to get the reference to the current thread by calling currentThread() method.

31. Program to create two threads. In this class use one constructor to start the thread and run it. Check whether these two threads are run are not.
32. Create a multithreaded program by creating a subclass of Thread and then creating, initializing, and staring two Thread objects from your class. The threads will execute concurrently and display Java is hot, aromatic, and invigorating to the console window.
33. An applet program to display the "Hello World" in the browser.
34. An Applet program that automatically display the text with Font Style, Font type
35. An Applet program that automatically display the text with Font Style, Font type using getParameter Method.
36. Program that displays the menu bar and when You click the options it has to display a dialog box stating which option has been clicked.
37. Program that has menu bar and also a quit option and if the user clicks the quite option the applet should quit.
38. Program to create a dialogbox and menu
39. Program to create a grid layout control
40. Program to create a border layout control
41. Program to create a padding layout control
42. Program to give the example for button control
43. Program to give the example for panel control.
44. Program that will display check boxes and option buttons they are numbered from1 to. Use a textbox to display the number those corresponding boxes or button checked.
45. Program to create a simple calculator
46. Program as above with combo box and list boxes instead
47. Program that displays the x and y position of the cursor movement using Mouse
48. Program to create a canvas
49. Program that displays the x and y position of the cursor movement using Keyboard

50. Program to create a text box control
51. Program to create an analog clock.
52. Program to create a Applet life cycle

Web Programming: Lab Exercises

53. Program to demonstrate different formats of text in XHTML
54. Program to demonstrate Anchor Tag in XHTML
55. Program to demonstrate Tables in XHTML
56. Program to demonstrate Cell Spacing and Cell Padding in a XHTML Table
57. Program to demonstrate different forms of Lists- Ordered, Unordered, Nested and description lists
58. Program to demonstrate Simple Frame using XHTML
59. Program to demonstrate Mixed Frames(combining Horizontal & Vertical frames)
60. Demonstration of Navigation through various frames
61. Program to demonstrate Form Fields
62. Program to demonstrate Character Entities
63. Program to demonstrate Internal Style Sheet
64. Program to demonstrate External CSS
65. Program to demonstrate Inline CSS
66. Program to demonstrate Border Colors using CSS
67. Program to demonstrate Text Alignments using CSS
68. HTML program to give different colours for different heading tags.
69. Using CSS invert the behaviour of the <h1> to <h6> tags.
70. Create a sample code to illustrate the procedure of creating user defined classes in CSS.
71. Demonstration of Simple Java Script program to display Date
72. Program to demonstrate Alert, Confirm and Prompt Message Boxes
73. Program to handle various events using Java Script

74. Program to handle Form Validation using Java Script
75. Create a java script program to accept the first, middle, last names of user and print them.
76. Write a java script program to add two numbers.
77. Write a java script program to find the factorial of given number.
78. Write a java Script program to print all prime numbers.
79. Write a java script program to sort the array (Bubble Sort).
80. Write a java script program to “Wish a user” at different hours of a day.
81. Prompt a user for the cost price and selling price of an article and output the profit and loss percentage.
82. Create a web page of customer profile for data entry of customer’s in a Hotel. The profile should include Name, Address, Age, gender, Room Type (A/C, Non-A/C or Deluxe), Type of payment (Cash, Credit/Debit Card or Coupons).
83. Create an Online Bio-Data Form for the Current Employees in the organization.
84. Design the simple Calculator.
85. HTML program using FRAMESET Tag to first divide the web page into two columns, and right column bottom row having the main page with text. The left host column with some other images.
86. HTML program using Java script to analyse examination result of a class of 10 students. If no. of students passed in that class in greater than no. of students failed then display the text 'Good Result'.
87. HTML program using Java script to demonstrate (a) Alert Box (b) Prompt dialogue
88. HTML program using Java script to perform comparison between two numbers entered by user, using relational operators.
89. HTML program using Java script to calculate the product of 3 integers.
90. PHP programs involving various control structures like: if, else, elseif/else if, while, do-while, for, foreach, switch, break, continue,

etc

91. PHP programs involving the following: declare, return, require, include, require-once, include_once and goto.
92. Programs to demonstrate PHP Array functions such as PHP Array Sorting, PHP Key Sorting, PHP Value Sorting, PHP MultiArray Sorting, PHP Array Random Sorting, PHP Array Reverse Sorting, Array to String Conversion, Implode() function, String to Array, Array Count, Remove Duplicate Values, Array Search, Array Replace, Array Replace Recursive, Array Sub String Search
93. PHP programs to demonstrate the following (a) use of regular expression to compare two strings, (b) Extract domain name from URL and (c) Find the number of rows from a mysql database for your query.
94. PHP program to generate a Guestbook which will allow your website visitor to enter some simple data about your website.
95. PHP program for Email Registration
96. PHP program for making application form and performing degree admission on-line.

BCA6B17 -- Project & Programme Viva-Voce

Course Number: 36

Contact Hours: 2L

Number of Credits: 2

Number of Contact Hours: 32L

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

Aim of the Course: To provide practical knowledge on software development.

Objectives of the Course:

- To provide practical knowledge on software development process

Prerequisites: Basic programming and system development knowledge.

Course Outline

The objective of the BCA 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 BCA 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 BCA final semester project. The courses studied by the students during the BCA 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 bonafied project report shall be submitted in hard bound complete in all aspects.

Open Courses

BCS5D01 - Introduction to Computers & Office Automation

Course Number: XX

Contact Hours: 3T

Number of Credits: 2

Number of Contact Hours: 30T

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

Aim of the Course: To provide the students with the basic knowledge on Computers and office automation.

Objectives of the Course:

- To get a general introduction to office automation packages
- To get a general introduction to Internet

Prerequisites: Basic knowledge Computers and Internet

Course Outline

UNIT I (7T)

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

UNIT II (7T)

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

UNIT III (8T)

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

UNIT IV (8T)

Presentation using (OpenOffice Impress/MS-Power Point): Presentations, Creating, Manipulating & Enhancing Slides, Organizational Charts, Charts, Word Art, Layering art Objects, Animations and Sounds, Inserting Animated Pictures or Accessing through Object, Inserting Recorded Sound Effect or In-Built Sound Effect.

References:

1. *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.

BCS5D02 - Introduction to Web Designing

Course Number: XX

Contact Hours: 3T

Number of Credits: 2

Number of Contact Hours: 30T

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

Aim of the Course: To provide the students with the basic skills on Web designing.

Objectives of the Course:

- To get a general introduction to Internet
- To achieve basic Web designing skills

Prerequisites: Basic knowledge Computers and Internet

Course Outline

UNIT I (7T)

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

UNIT II (7T)

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

UNIT III (8T)

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

UNIT IV (8T)

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

References:

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

BCS5D03 - Introduction to Problem Solving and C Programming

Course Number: XX

Contact Hours: 3T

Number of Credits: 2

Number of Contact Hours: 30T

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

Aim of the Course: To provide the students with the basic programming skills.

Objectives of the Course:

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

Prerequisites: None

Course Outline

UNIT I (7T)

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

UNIT II (7T)

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

UNIT III (8T)

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

UNIT IV (8T)

Array & Strings - One dimensional array, two dimensional array and 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, bit-fields.

References:

1. *Programming in ANSI C*, E. Balaguruswami.
2. *The C Programming Language*, Brian W. Kernighan & Dennis M. Ritchie.
3. *Let us C*, Yashvant P. Kanetkar.
4. *Programming with C*, Byran Gotfried, Schaums Outline series.

Elective Courses

BCA6B18a - Computer Graphics

Course Number: 37

Contact Hours: 4T

Number of Credits: 4

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with the basic knowledge of Computer Graphics.

Objectives of the Course:

- To understand the basic knowledge of graphics devices
- The awareness of 2D
- To get the basic concepts graphics algorithms
- To learn the concepts of 3D

Prerequisites: Basic knowledge in Information Systems

Course Outline

UNIT I (12T)

Overview of Computer Graphics: Historical background of Computer Graphics; Applications of Computer Graphics; Popular Graphics Software; Display devices: Pixel, Resolution, Aspect Ratio; Raster-Scan Systems and Display : CRT, Refresh Rate and Interlacing; Bit Planes, Colour Depth and Colour Palette, Frame Buffer, Video Controller, Raster-Scan Display Processor, Lookup Table, RGB Colour Model, Colour CRT monitors; Random-Scan Displays; Flat Panel Display: LCD, Plasma Panel; Graphics Monitors and Workstations; Popular Graphics Input Devices; Hard-Copy Devices

UNIT II (12T)

Coordinate Representations; Graphics Primitives: Line Drawing Algorithms- DDA Algorithm, Bresenham's Algorithm; Different Line

Styles; Circle-Generating Algorithms - Properties of Circles, Circle Drawing using Polar Coordinates, Bresenham's Circle Drawing Algorithm; Ellipse Generating Algorithms; Anti-aliasing;

UNIT III (12T)

Geometric Transformations: Scaling, Translation, Rotation; Matrix Representations and Homogeneous Coordinates; Rotation Relative to an Arbitrary Point; Reflection; Shearing; Coordinate Transformation; Inverse Transformation; Affine Transformation; Raster Transformation; Composite Transformations; Fixed-point Scaling; Input Techniques: Pointing, Positioning, Rubber-band method, Dragging;

UNIT IV (12T)

Two-Dimensional Viewing: Window-to-Viewport Coordinate Transformation; Zooming; Panning; Clipping: Point Clipping, Line Clipping- Cohen-Sutherland line clipping, Mid-point Subdivision Line Clipping; Polygon Clipping – Sutherland-Hodgeman Polygon Clipping; Text Clipping;

UNIT V (12T)

Graphics in Three Dimensions: Displays in Three Dimensions, 3-D Transformations; 3-D Viewing: Viewing Parameters, Projections, Parallel and Perspective projection; Hidden Surfaces: Z-Buffer Method, Painter's Algorithm;

References:

1. *Computer Graphics*, Donald Hearn, M. Pauline Baker, PHI.
2. *Computer Graphics*, Apurva A. Desai, PHI
3. *Theory and Problems of Computer Graphics* (Shaums Series), Plastock R. and XiangZ
4. *Principles of Interactive Computer Graphics*, Newmann & Sproull, McGraw Hill
5. *Computer Graphics Principles & Practice*, Foley etc. Addison Wesley
6. *Procedural Elements of Computer Graphics*, Rogers, McGraw Hill
7. *Introduction to Computer Graphics and Multimedia*, Anirban

Mukhopadhyay, Arup Chattopadhyay, Vikas

8. *Computer Graphics*, Zhigang Xiang, Roy Plastock, TMH
9. *Fundamentals of Computer Graphics and Multimedia*, D.P. Mukherjee, PHI

BCA6B18b - Multimedia Systems

Course Number: 37

Contact Hours: 4T

Number of Credits: 4

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with an introductory knowledge on Multimedia technology and devices.

Objectives of the Course:

- To get a general introduction and basic skills on Multimedia techniques and tools.

Prerequisites: Basic knowledge of +2 level Mathematics

Course Outline

UNIT I (12T)

Multimedia Definition, Use Of Multimedia, Delivering Multimedia, Text: About Fonts and Faces, Using Text in Multimedia, Computers and Text, Font Editing and Design Tools, Hypermedia and Hypertext.

UNIT II (12T)

Images: Plan Approach, Organize Tools, Configure Computer Workspace, Making Still Images, Colour, Image File Formats. Sound: The Power of Sound, Digital Audio, Midi Audio, Midi vs. Digital Audio, Multimedia System Sounds, Audio File Formats –Vaughan's Law of Multimedia Minimums, Adding Sound to Multimedia Project.

UNIT III (12T)

Animation: The Power of Motion, Principles of Animation, Animation by Computer, Making Animations that Work. Video: Using Video, Working with Video and Displays, Digital Video Containers, Obtaining Video Clips, Shooting and Editing Video.

UNIT IV (12T)

Making Multimedia: The Stage of Multimedia Project, The Intangible Needs, The Hardware Needs, The Software Needs, An Authoring Systems' Needs. Multimedia Production Team

UNIT V (12T)

Planning and Costing: The Process of Making Multimedia, Scheduling, Estimating, RFPs and Bid Proposals. Designing and Producing, Content and Talent: Acquiring Content, Ownership of Content Created for Project, Acquiring Talent

References:

1. *Multimedia: Making It Work*, Tay Vaughan
2. *Multimedia Computing, Communication & Applications*, Ralf Steinmetz & Klara Nahrstedt, Pearson Education

BCA6B18c - Software Testing & Quality Assurance

Course Number: 37

Contact Hours: 4T

Number of Credits: 4

Number of Contact Hours: 60T

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course: To provide the students with an introductory knowledge on software testing and quality assurance techniques.

Objectives of the Course:

- To get a general introduction and basic skills on software testing and quality assurance techniques and tools

Prerequisites: Basic knowledge of Software Engineering

Course Outline

UNIT I (12T)

Phases of Software project - Quality Assurance, Quality control - Testing, Verification and Validation - Process Model to represent Different Phases - Life Cycle models. White-Box Testing: Static Testing - Structural Testing Challenges in White-Box Testing.

UNIT II (12T)

Black-Box Testing: What is Black, Box Testing?, Why Black, Box Testing?, When to do Black, Box Testing?, How to do Black, Box Testing?, Challenges in White Box Testing, Integration Testing: Integration Testing as Type of Testing, Integration Testing as a phase of Testing, Scenario Testing, Defect Bash.

UNIT III (12T)

System and Acceptance Testing: system Testing Overview, Why System testing is done? Functional versus Non, functional Testing, Functional testing, Non, functional Testing, Acceptance Testing, Summary of Testing Phases.

UNIT IV (12T)

Performance Testing: Factors governing Performance Testing, Methodology of Performance Testing, tools for Performance Testing, Process for Performance Testing, Challenges. Regression Testing: What is Regression Testing? Types of Regression Testing, When to do Regression Testing, How to do Regression Testing, Best Practices in Regression Testing.

UNIT V (12T)

Test Planning, Management, Execution and Reporting: Test Planning, Test Management, Test Process, Test Reporting, Best Practices. Test Metrics and Measurements: Project Metrics, Progress Metrics, Productivity Metrics, Release Metrics.

References:

1. *Software Testing Principles and Practices*, Srinivasan Desikan & Gopalswamy, Ramesh, Pearson Education.
2. *Effective Methods of Software Testing*, William E. Perry, Wiley
3. *Software Testing*, Renu Rajani and Pradeep Oak, TMH
4. *Software Testing Tools,* K. V. K. K. Prasad, Dreamtech Press
5. *Introducing Software Testing*, Lauise Tamres, Pearson Education