ST. JOSEPH'S COLLEGE (AUTONOMOUS), DEVAGIRI, CALICUT

SYLLABUS

FOR

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

(UNDER SJCBCSSUG 2019 SYSTEM)

(EFFECTIVE FROM 2019 ADMISSION)

REGULATIONS

FOR THE DEGREE OF
BACHELOR OF SCIENCE IN COMPUTER SCIENCE
(Choice Based Credit and Semester System for Under Graduate Curriculum 2019)
EFFECTIVE FROM THE ACADEMIC YEAR 2019-20

B.Sc. COMPUTER SCIENCE PROGRAMME OBJECTIVE

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

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

PROGRAMME STRUCTURE

Duration: The duration of the B.Sc Computer Science Programme shall be six semesters distributed over a period of three 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 UG programme shall include six types of courses, viz; Common Courses (Code A), Core courses (Code B), Complementary courses (Code C), Open Course (Code D) , Elective Course (Code E) and Audit courses (Code F).

Course code: Each course shall have a unique alphanumeric code number, which includes the code of the syllabus revision, abbreviation of the subject in three letters, the semester number (1 to 6) in which the course is offered, the code of the course (A to E) and the serial number of the course (01,02 ……) and the code for indicating whether the course is a theory course(T) or a practical course(P). For example: ENG2A03T represents a common course of serial number 03 offered in the second semester and PHY2B02T representing second semester Core course 2 in Physics programme.

Credits(C): Each course shall have certain credits. A student is required to acquire a minimum of 140 credits for the completion of the UG programme, of which 120 credits are to be acquired from class room study and shall only be counted for SGPA and CGPA. Out of the 120 credits, 14 credits for common courses (English), 8 credits for additional language courses and 16 credits for General courses. The maximum credits for a course shall not exceed 5. Audit courses shall have 4 credits per course and a total of 16 credits in the entire programme. The maximum credit acquired under extra credit shall be 4. If more Extra credit activities are done by a student, it may be mentioned in the Grade card. The credits of audited courses or extra credits are not counted for SGPA or CGPA.

Letter Grade or simply **Grade (G):** in a course is a letter symbol (O, A+, A, B+, B, C, P, F, I and Ab). Grade shall mean the prescribed alphabetical grade awarded to a student based on his/her performance in various examinations.

Credit Point (P): of a course is the value obtained by multiplying the grade point (G) by the credit (C) of the course: P=G x C.

Extra Credit is the additional credit awarded to a student over and above the minimum credits required in a programme, for achievements in co-curricular activities and social activities conducted outside the regular class hours, as decided by the University. For calculating CGPA, extra credits will not be considered.

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 Department concerned. Condonation of shortage of attendance to a maximum of 10% in the case of single condonation and 20% in the case of double condonation in a semester shall be granted by the college after remitting the required fee.

Benefits of attendance may be granted to students who attend the approved activities of the college with the 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 (within two weeks) in curricular/extracurricular activities (maximum 9 days in a semester). Students can avail of condonation of shortage of attendance in a maximum of four semesters during the entire programme (Either four single condonations or one double condonation and two single condonations during the entire programme) . If a student fails to get 65% attendance, he/she can move to the next semester only if he/she acquires 50% attendance. In that case, a **provisional registration** is needed. Such students can appear for supplementary examination for such semesters after the completion of the programme. Less than 50% attendance requires Readmission. Readmission is permitted only once during the entire programme.

Strike off the roll: A student who is continuously absent for 14 days without sufficient reason and proper intimation to the Principal of the college shall be removed from the roll.

Ability Enhancement courses/Audit courses: These are courses which are mandatory for a programme but not counted for the calculation of SGPA or CGPA. There shall be one Audit course each in the first four semesters. These courses are not meant for class room study. The students can attain only pass (Grade P) for these courses. At the end of each semester there shall be examination conducted by the college from a pool of questions (Question Bank). The students can also attain these credits through online courses like SWAYAM, MOOC etc (optional). The list of courses in each semester with credits is given below.

Course with credit	Credit	Semester
Environment Studies	4	1
Disaster Management	4	2
*Human Rights/Intellectual Property Rights/ Consumer Protection	4	3
*Gender Studies/Gerontology	4	4

* College can opt any one of the courses.

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 head of the college(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. In addition, maximum of 6 marks per semester can be awarded to the students of UG Programmes, for participating in the College Fitness Education Programme (COFE).

Project: Every student of a UG degree programme shall have to work on a project of 2 credits under the supervision of a faculty member or shall write a theory course based on Research Methodology as per the curriculum. College shall have the liberty to choose either of the above.

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

ADMISSION

The admission to all programmes will be as per Rules and Regulations of the college. The eligibility criteria for admission shall be as announced by the college 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 the college 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 college 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 college websites.

There shall be a uniform **calendar** prepared by the college for the registration, conduct/schedule of the courses, examinations and publication of results. The college

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. College transfer may be permitted in Second and Fourth semester also without change in complementary course within a period of two weeks from the date of commencement of the semester concerned.

Complementary change at the time of college transfer is permitted in the third semester if all conditions are fulfilled.

CBCSS regular students can join distance education stream/Private Registration in any semester in the same programme or different one. If core and complementary courses are different, they have to undergo them in the new stream. The marks/grace obtained for common courses will be retained.

A student registered under distance education stream/Private Registration in the CBCSS pattern may be permitted to join the regular college (if there is a vacancy within the sanctioned strength) in the third and fifth semester with the same programme only. If there is a change in complementary courses, it can be done with following conditions:

i) the external and internal marks/grade obtained in the previous semesters for the earlier complementary courses will be cancelled.

ii) the students have to write the external examinations for the previous semester for the new complementary courses along with the subsequent batch.

iii) An undertaking to the effect that "the internal evaluation for the previous semesters of the new complementary courses will be conducted", is to be obtained from the Principal of the college in which the student intends to join.

Provision for **credit transfer** is subject to common guidelines prepared by the faculty concerned.

There shall be provision for **Readmission** of students in SJCBCSSUG 2019.

The Principal can grant readmission to the student, subject to the conditions detailed below and inform the matter of readmission to the Controller of Examinations within one month of such readmission.

This readmission is not to be treated as college transfer. There should be a gap of at least one semester for readmission.

The candidate seeking readmission to a particular semester should have registered for the previous semester examination. Readmission shall be taken within two weeks from the date of commencement of the semester concerned. For readmission, the vacancy should be within the sanctioned strength in the parent college. If there is a change in complementary courses, it can be done with following conditions: i) the external and internal marks/grade obtained in the previous semesters for the earlier complementary courses will be cancelled. ii) the students have to write the external examinations for the previous semester for the new complementary courses along with the subsequent batch iii) An undertaking to the effect that "the internal evaluation for the previous semesters of the new complementary courses will be conducted", is to be obtained from the Principal of the college.

If change in scheme occurs while readmission, provision for credit transfer is subject to common guidelines prepared by Board of Studies/ Faculty concerned. For readmission to SJCBCSS UG 2019 involving scheme change, the Principal concerned shall report the matter of readmission to Controller of Examinations with the details of previous semesters and course undergone with credits within two weeks in order to fix the deficiency/excess papers.

REGISTRATION

Each student shall make an online registration for the courses he/she proposes to take, in consultation with the Faculty Adviser within two weeks from the commencement of each semester.

It is mandatory that the students who got admission under SJCBCSSUG 2019 in SDE/Private shall register for the examinations of the concerned semesters in the same year itself.

A student shall be normally permitted to register for the examination if he/she has required minimum attendance. If the student has a shortage of attendance below 65% in a semester, the student shall be permitted to move to the next semester (if the attendance is more than 50% - Provisional registration) and can write the examination for the entire

courses of the semester in which shortage of attendance occurs as supplementary examination only after the completion of the entire programme. In such cases, a request from the student may be forwarded through the Principal of the college to the Controller of Examinations within two weeks of the commencement of the semester. If the attendance is less than 50%, the student is ineligible to continue the programme and has to seek readmission. **There will not be any Repeat semester in SJCBCSSUG 2019**.

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.

For open courses there shall be a minimum of 10 and maximum of 75 students per batch. For other courses existing pattern will be followed.

Those students who have followed the UG Programmes in annual pattern or Choice based Credit & Semester System pattern can cancel their earlier registration and register afresh for SJCBCSSUG 2019 scheme in the same discipline or a different one.

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

EXAMINATION

There shall be End Semester Examination at the end of each semester. Practical examinations shall be conducted by the College as prescribed by the Board of studies. External viva-voce, if any, shall be conducted along with the practical examination/project evaluation. The model of question papers may be prepared by the concerned Board Of Studies. Each question should aim at – (1) assessment of the knowledge acquired (2) standard application of knowledge (3) application of knowledge in new situations. Different types of questions shall possess different marks to quantify their range.

Project evaluation shall be conducted at the end of sixth semester. 20% of marks are awarded through internal assessment.

Improvement course: Improvement of a particular semester can be done only once. The student shall avail of the improvement chance in the succeeding year after the successful completion of the semester concerned. The students can improve a maximum of two courses in a particular semester. The internal marks already obtained will be carried forward to determine the new grade/mark in the improvement examination (for regular students). If the

candidate fails to appear for the improvement examination after registration, or if there is no change in the results of the improved examination, the mark/grade obtained in the first appearance will be retained.

Improvement and supplementary examinations cannot be done simultaneously.

Moderation: Moderation is eligible as per the existing rules of the Academic Council.

COURSE EVALUATION

Total marks for each complementary, elective course is 75 and lab courses shall be **100** marks and Industrial Vist & Project Evaluation cum Programme viva- voce shall be **100** marks and open course shall be **75** marks. The evaluation scheme for each course shall contain two parts (1) Internal evaluation (2) external evaluation 20% weight shall be given to the internal evaluation. The remaining 80% weight shall be for the external evaluation.

INTERNAL EVALUATION

The internal assessment shall be based on a predetermined transparent system involving written test, assignments, seminars and attendance in respect of theory courses and on test/record/viva/attendance in respect of lab courses. 20% of the total marks in each course are for internal examinations.. Internal assessment of the project will be based on its content, method of presentation, final conclusion and orientation to research aptitude.

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

Serial no.	Components	Marks(For Courses With Max. Marks 75)	Marks(For Courses With Max. Marks 100)
1	Attendance	3	4
2	Assignment	3	4
3	Seminar	3	4
4	Test Paper	6	8
Total		**15**	**20**

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

Test paper (50%)   –   10 Marks

Attendance (20%)   –   4 Marks

Assignment/Lab involvement (30%) –   6 Marks

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

For the test paper marks, at least one test paper should be conducted. If more test papers are conducted, the mark of the best one should 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 college Principal after obtaining the signature of both course teacher and Head of the Department.

The Split up of marks for Test paper and Class Room Participation (CRP) for internal evaluation are as follows.

Split up of marks for Test paper:

	Theory	Practical
85% -100%	6	10
65% -85%	5	8
55% - 65%	4	6
45% - 55%	3	5
35%- 45%	2	4
Less than 35%	1	2

Attendance of each course will be evaluated as below

	Theory	Practical
85 % and above	3	4
75% ≤CRP <85%	2	2
50% ≤CRP <75%	1	1

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

Originality	–	3 Marks
Methodology	–	3 Marks
Scheme/Organization of Report	–	4.5 Marks
Viva-voce	–	4.5 Marks

EXTERNAL EVALUATION

There shall be end semester examinations for each course at the end of each semester. Practical examinations shall be conducted by the college at the end of second, fourth and sixth semesters. External evaluation of Project, Industrial Visit Report and 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., 60 marks, for each theory course.

The model of the question paper for external examination for theory courses of 2Hrs. duration. The students can answer all the questions in Sections A and B. But there shall be Ceiling in each section.

1. **Section A**: Short answer type carries 2 marks each - 12 questions **(Ceiling - 20)**
2. **Section B:** Paragraph/ Problem type carries 5 marks each - 7 questions **(Ceiling - 30)**
3. **Section C:** Essay type carries 10 marks (1 out of 2)

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. The external examination in practical courses shall be conducted by two examiners, one internal and an external. The project evaluation with Programme viva voce will be conducted by two examiners, one internal and an external (appointed by the University), at the end of the sixth semester. No practical examination will be conducted in odd semester. Practical examinations for B.Sc Computer Science Programme shall be conducted in the even semester 2, 4 and 6.

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

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

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

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

Number of students in one batch for practical examination must be limited to 14.

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

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

1. Relevance of the Topic, Statement of Objectives (**Total 10 Marks**)

2. Methodology (Reference/ Bibliography, Presentation, quality of Analysis/Use of Statistical Tools) (**Total 10 Marks**)

3. Findings and recommendations (**Total 15 Marks**)

4. Project cum Programme Viva Voce (**Total 15 Marks)**

5. Report of Industrial visit **(Total 10 Marks)**

External Examiners will be appointed by the College from the list of VI Semester Board of Examiners in consultation with the Chairperson of the Board.

Audit course: The students can attain only pass (Grade P) for these courses. At the end of each semester (up to fourth semester) there shall be examination conducted by the college from a pool of questions. The students can also attain the credits through online courses like SWAYAM, MOOC etc.

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/College.

COURSE IMPROVEMENT

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

improvement examination appeared, the marks/grades obtained in the first appearance will be retained. Improvement and supplementary examinations cannot be done simultaneously.

EVALUATION AND GRADING

Evaluation (both internal and external) is carried out using Mark system. The Grade on the basis of total internal and external marks will be indicated for each course, for each semester and for the entire programme.

Indirect Grading System in 10-point scale is as below:

Ten Point Indirect Grading System

Percentage of Marks (Both Internal &External put together)	Grade	Interpretation	Grade point Average (G)	Range of grade points	Class
95 and above	O	Outstanding	10	9.5 -10	First Class with Distinction
85 to below 95	A+	Excellent	9	8.5 -9.49	
75 to below 85	A	Very good	8	7.5 -8.49	
65 to below 75	B+	Good	7	6.5 -7.49	First Class
55 to below 65	B	Satisfactory	6	5.5 -6.49	
45 to below 55	C	Average	5	4.5 -5.49	Second Class
35 to below 45	P	Pass	4	3.5 -4.49	Third Class
Below 35	F	Failure	0	0	Fail
Incomplete	I	Incomplete	0	0	Fail
Absent	Ab	Absent	0	0	Fail

SGPA Calculation

Example – 1

Semester I Course Code	Course Name	Grade Obtained	Grade point (G)	Credit (C)	Credit point (CXG)
xxxxxxx	Xxxxxxx	A	8	4	32
xxxxxxx	Xxxxxxxxxx	C	5	3	15
xxxxxxx	Xxxxxxxxxx	A+	9	4	36
xxxxxxx	Xxxxxxxxxx	B+	7	3	21
xxxxxxx	Xxxxxxxxx	P	4	3	12
xxxxxxx	Xxxxxxxxx	C	5	4	20

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

$$\text{SGPA} = \frac{(32+15+336+21+12+20)}{21} = \frac{136}{21}$$

SGPA = 6.476

Percentage of marks of semester I = (SGPA/10) x 100 = 64.76 %

Note: The SGPA is corrected to three decimal points and the percentage of marks shall be approximated to two decimal points.

Example: 2

Semester II Course Code	Course Name	Grade Obtained	Grade point (G)	Credit (C)	Credit point (CXG)
xxxxxxx	Xxxxxxx	A	8	4	32
xxxxxxx	Xxxxxxxxxx	C	5	3	15
xxxxxxx	Xxxxxxxxxx	A+	9	4	36
xxxxxxx	Xxxxxxxxxx	B+	7	3	21
xxxxxx*	Xxxxxxxxx	F	0	3	0
xxxxxxx	Xxxxxxxxx	C	5	4	20

*Failed course

Note: In the event a candidate failing to secure **'P'** grade in any Course in a semester, consolidation of SGPA and CGPA will be made only after obtaining **'P'** grade in the failed Course in the subsequent appearance.

CGPA Calculation

$$\text{CGPA} = \frac{\textbf{Total Credit points obtained in six semesters}}{\textbf{Total Credits acquired (120)}}$$

Example

$$\text{CGPA} = \frac{\textbf{136 + 145 + 161 + 148 + 131 + 141}}{\textbf{120}} = \frac{\textbf{862}}{\textbf{120}}$$

$$\text{CGPA} = \textbf{7.183}$$

$$\text{Total percentage of marks} = \frac{\text{CGPA}}{10} \times 100 = \frac{7.183}{10} \times 100 = \textbf{71.83}$$

$$\text{CGPA of Core Courses} = \frac{\textbf{Total Credit points obtained for Core Courses}}{\textbf{Total Credits acquired for Core Courses}}$$

Similarly CGPA of Complementary courses, Open courses, English Common courses and Additional Language Common courses may be calculated and the respective percentage may be calculated. All these must be recorded in the Final Grade Card.

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 UG 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) Letter grade in each course in the semester

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

The final Grade 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. The final grade card shall show CGPA (corrected to three decimal places), percentage of marks (corrected to two decimal places) and the overall letter grade of a student for the entire programme. The final grade card shall also include the CGPA and percentage of marks of common courses, core courses, complementary courses and open courses separately. This is to be done in a 10-point indirect scale. The final Grade card also contains the list of Audit courses passed and the details of Extra credits.

Evaluation of Audit courses: The examination shall be conducted by the college itself from the Question Bank. The Question paper shall be of 100 marks of 3 hour duration. For SDE/Private students it may be of MCQ/ fill in the blank type questions or online question paper may be introduced.

SOCIAL SERVICE PROGRAMME (SSP)

In this programme, a student has to complete 12 days of social service. This has to be completed in the first four semesters; 3 days in each semester. For the regular programme the student has to work in a Panchayath or Local body or in a hospital/ poor home or old age home or in a Pain &palliative center or any social work assigned by the College authorities. Students who engaged in College Union activities and participate in sports and cultural activities in Zonal level have to undergo only 6 days of SSP during the entire programme. The whole documents regarding the student should be kept in the college and the Principal should give a Certificate for the same. The list of students (successfully completed the programme) must be sent to the Controller of Examinations before the commencement of the fifth semester examinations. A College level Co-ordinator and a Department level Co-ordinator shall be appointed for the smooth conduct of the programme.

AWARD OF DEGREE

The successful completion of all the courses (common, core, complementary and open courses) prescribed for the B.Sc. Computer Science Programme with P grade (35 %) shall be the minimum requirement for the award of B.Sc Computer Science Programme degree.

GRIEVANCE REDRESSAL COMMITTEE

Department level: The College shall form a Grievance Redressal Committee in each department comprising of course teacher, one senior teacher and elected representative of students (Association Secretary) 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.

College level: There shall be a college level grievance redressal committee comprising of student adviser, two senior teachers, two staff council members (one shall be elected member) and elected representative of students (College Union Chairperson) as members and Principal as Chairman.

A Steering Committee consisting of one governing body members of whom shall be a teacher, Controller of Examinations, two teachers from different disciplines (one each from science and arts), two Chairpersons of Board of Studies shall be formed to resolve the issues, arising out of the implementation of SJCBCSSUG 2019. The governing body member who is also a teacher shall be the Convener of the committee. The quorum of the committee shall be four and meeting of the committee shall be held at least thrice in an academic year. The resolutions of the committee will be implemented by the Principal in exigency and this may be ratified by the Academic Council.

PROGRAMME SPECIFIC OUTCOMES

PSOs	PROGRAMME SPECIFIC OUTCOMES
PSO1	An ability to understand the core concepts of computer science and to enrich problem solving skills to analyse, design and implement software-based systems of varying complexity
PSO2	To interpret real-time problems with analytical skills and to arrive at cost effective and optimal solution using advanced tools and techniques.
PSO3	An understanding of social awareness and professional ethics with practical proficiency in the broad area of programming concepts by lifelong learning to inculcate employment and entrepreneurship skills
PSO4	Learn to analyse and design a real world problem implementation and understand the dynamic behavior of a system.
PSO5	Demonstrate the ability to think critically and analyse problems; find effective solutions.
PSO6	Understand the impact of the professional software engineering solutions in social and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PSO7	Recognition of the need for and ability to engage in continuing professional development.
PSO8	An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computational systems in a way that demonstrates comprehension of the trade-offs involved in design choices.
PSO9	An understanding of professional, ethical, legal, security, and social issues and responsibilities for the computing profession.
PSO10	Recognize the need for, and develop the ability to engage in continuous learning in the context of technological change.

B.Sc. COMPUTER SCIENCE PROGRAMME STRUCTURE

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

Semester I									
No	**Course Code**	**Course Name**	**C**	**Marks**			**Hrs/wk**		
				I	**E**	**T**	**L**	**P**	**T**
01	GENG1A01T	Common English	3	15	60	75	4	0	4
02	GENG1A02T	Common English	4	20	80	100	5	0	5
03	GMAL1A07T/ GHIN1A07T	Languages other than English	4	20	80	100	5	0	5
04	GBCS1B01T	Computer Fundamentals & HTML	3	15	60	75	1	2	3
05	GMAT1C01T	Complementary I: Mathematics I	3	15	60	75	4	0	4
06	GSTA1C01T	Complementary II: Introductory Statistics	3	15	60	75	4	0	4
Total			**20**			**500**	**23**	**2**	**25**

Semester II

No	Course Code	Course Name	C	Marks			Hrs/wk		
				I	E	T	L	P	T
07	GENG2A03T	Common English	3	15	60	75	4	0	4
08	GENG2A04T	Common English	4	20	80	100	5	0	5
09	GMAL2A08T/ GHIN2A08T	Languages other than English	4	20	80	100	5	0	5
10	GBCS2B02T	Problem Solving using C	3	15	60	75	1	2	3
11	GBCS2B03P	Programming Laboratory I: HTML and Programming in C	4	20	80	100	Lab Exam of GBCS1B01T & GBCS2B02T		
12	GMAT2C02T	Complementary I: Mathematics II	3	15	60	75	4	0	4
13	GSTA2C02T	Complementary II: Probability Theory	3	15	60	75	4	0	4
		Total	**24**			**600**	**23**	**2**	**25**

Semester III

No	Course Code	Course Name	C	Marks			Hrs/wk		
				I	E	T	L	P	T
14	GBCS3A01T	Python Programming	4	20	80	100	4	0	4
15	GBCS3A02T	Data Communication and Optical Fibres	4	20	80	100	4	0	4
16	GBCS3B04T	Data Structures using C	3	15	60	75	3	4	7
17	GMAT3C03T	Complementary I: Mathematics III	3	15	60	75	5	0	5
18	GSTA3C03T	Complementary II: Probability Distributions and Sampling Theory	3	15	60	75	5	0	5
		Total	**17**			**425**	**21**	**4**	**25**

Semester IV

No	Course Code	Course Name	Credit	Marks			Hrs/wk		
				I	E	T	L	P	T
19	GBCS4A03T	Microprocessors Architecture and Programming	4	20	80	100	4	0	4
20	GBCS4A04T	Sensors and Transducers	4	20	80	100	4	0	4
21	GBCS4B05T	Database Management System and RDBMS	3	15	60	75	3	4	7
22	GBCS4B06P	Programming Laboratory II: Data Structures and RDBMS	4	20	80	100	Lab Exam of GBCS3B04T & GBCS4B05T		
23	GMAT4C04T	Complementary I: Mathematics IV	3	15	60	75	5	0	5
24	GSTA4C04T	Complementary II: Statistical Inference and Quality Control	3	15	60	75	5	0	5
Total			**21**			**525**	**21**	**4**	**25**

Semester V

No	Course Code	Course Name	Credit	Marks			Hrs/wk		
				I	E	T	L	P	T
25	GBCS5B07T	Computer Organization and Architecture	3	15	60	75	4	0	4
26	GBCS5B08T	Java Programming	3	15	60	75	3	3	6
27	GBCS5B09T	Web Programming Using PHP	3	15	60	75	3	3	6
28	GBCS5B10T	Principles of Software Engineering	3	15	60	75	4	0	4
29	GXXX5DXXT	Open Course	3	15	60	75	3	0	3
	GBCS6B16D	Industrial Visit & Project Work	Evaluation at the end of 6th Sem.				0	2	2
Total			**15**			**375**	**17**	**8**	**25**

No	Course Code	Course Name	Credit	Marks			Hrs/wk		
		Semester VI							
				I	E	T	L	P	T
30	GBCS6B11T	Android Programming	3	15	60	75	4	3	7
31	GBCS6B12T	Operating Systems	3	15	60	75	4	3	7
32	GBCS6B13T	Computer Networks	3	15	60	75	5	0	5
33	GBCS6B14P	Programming Laboratory III: Java and PHP Programming	4	20	80	100	Lab exam of GBCS5B08T & GBCS5B09T		
34	GBCS6B15P	Programming Laboratory IV: Android and Linux shell Programming	4	20	80	100	Lab exam of GBCS6B11T & GBCS6B12T		
35	GBCS6EXXT	Elective Course	3	15	60	75	4	0	4
36	GBCS6B16D	Industrial Visit & Project Work	3	15	60	75	0	2	2
Total			**23**			**575**	**17**	**8**	**25**

List of Electives		
No	**Course Code**	**Course Name**
1	GBCS6E01T	System Software
2	GBCS6E02T	Machine Learning
3	GBCS6E03T	Discrete Structures
4	GBCS6E04T	Computer Graphics
5	GBCS6E05T	Technical Writing
6	GBCS6E06T	Fundamentals of Life Skill Education

List of Open Courses (For students of other UG programmes)		
No	**Course Code**	**Course Name**
1	GBCS5D01T	Introduction to Computers and Office Automation
2	GBCS5D02T	Web Designing
3	GBCS5D03T	Introduction to Problem Solving and C Programming
4	GBCS5D04T	Introduction to Data Analysis using spread sheet

List of Audit Courses				
Semester	**Course Code**	**Course Name**	**Credits**	**Marks**
1	GAUD1F01T	Environment Studies	4	100
2	GAUD2F02T	Disaster Management	4	100
3	GAUD3F03T	Human Rights	4	100
	GAUD3F04T	Intellectual Property Rights		
	GAUD3F05T	Consumer Protection		
4	GAUD4F06T	Gender Studies	4	100
	GAUD4F07T	Gerontology		
TOTAL			16	400

SEMESTER I
Core Course
GBCS1B01T: COMPUTER FUNDAMENTALS AND HTML

Course Number: 4
Contact Hours per Week: 3 (1L + 2P)
Number of Credits: 3
Number of Contact Hours: 48 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites

- Background of the basic science at +2 level

Course Outcome

COs	COURSE OUTCOMES
CO1	To equip the students with fundamentals of Computer
CO2	To learn the basics of Computer organization
CO3	To equip the students to write algorithm and draw flow chart for solving simple problems
CO4	To learn the basics of Internet and webpage design

Course Outline

Unit I [4 L]
Concepts of Hardware and Software: Computer Languages, Language Translators, Features of good language, Basics Computer Organization: Von Neumann Model, Input Unit, Output Unit, Storage Unit, Control Unit, Memory Hierarchy, Primary Storage, Cache Memory, Registers, Secondary Storage Devices, Basics of Hardware Components – SMPS, Motherboard, Add-on Cards, Ports, Memory, Adapters, Network cables, Basic Computer Configuration

Unit II [5 L]
Number Systems and Boolean Algebra – Decimal, Binary, Octal and Hexadecimal Numbers, Arithmetic involving Number Systems, Inter Conversions of Number Systems, 1"s and 2's Complements, Complement Subtractions, Digital Codes – Binary Coded Decimal (BCD), ASCII Code ,Unicode, Gray Code, Excess-3 Code. Boolean Algebra: Boolean Operations, Logic Expressions, Postulates, Rules and Laws of Boolean Algebra, DeMorgan's Theorem, Minterms, Maxterms, SOP and POS form of Boolean Expressions for Gate Network, Simplification of Boolean Expressions using Boolean Algebra and Karnaugh Map Techniques (up to 4 variables)

Unit III [3 L]

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

Unit IV [2L + 16P]

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

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

Unit V [2L+16P]

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

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

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

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

SEMESTER II
Core Course
GBCS2B02T: PROBLEM SOLVING USING C

Course Number: 10
Contact Hours per Week: 3(1L+2P)
Number of Credits: 3
Number of Contact Hours: 48Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites

- Knowledge of basic computer operations

Course Outcome

COs	COURSE OUTCOMES
CO1	To equip the students with fundamental principles of Problem Solving aspects.
CO2	To learn the concept of programming
CO3	To study C language
CO4	To equip the students to write programs for solving simple computing problems

Course Outline

Unit I [3L+ 3P]
Introduction to C Programming - Overview and Importance of C, C Program Structure, Sample programs.

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

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

Unit II [2L+ 6P]
C Operators - Arithmetic operators, relational operators, and logical operators, assignment operators, increment and decrement operators, conditional operators, special operators, arithmetic expressions, evaluation of expressions, precedence of arithmetic operators, Type conversion in expressions, operator precedence and associativity, Mathematical Functions, I/O operations - Library functions.

Unit III [3L + 7P]

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

Unit IV [4L+ 8P]

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

The Concept of modularization and User defined functions - Definition - Multifunction Program, proto-types, Passing arguments, calling functions, various categories of functions, Nesting of functions and recursion, functions and arrays, scope and lifetime of variables in functions, multi-file programs.
Structures & Union structure definition - giving values to members, structure initialization, comparison of structure variables, arrays of structures, arrays within structures, structures within arrays, structures and functions, Unions, bit fields.

Unit V [4L+8P]

Pointers - Understanding pointers, accessing the address of a variable, declaring and initializing pointers, accessing a variable through its pointer, pointer expressions, pointer and arrays, pointer and character string, pointers and functions, pointers and structures, pointer to pointer dynamic memory allocation.
Files: Creating, Processing, Opening and Closing a data file, command line operations

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

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

GBCS2B03P: PROGRAMMING LABORATORY I: HTML AND PROGRAMMING IN C
(Lab Exam of GBCS1B01T & GBCS2B02T)

Course Number: 11
Number of Credits: 4
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Prerequisites:

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

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To make the students learn web designing
CO2	To make the students learn programming environments.
CO3	To practice procedural programming concepts.
CO4	To make the students equipped to solve mathematical or scientific problems using C

Course Outline:

Part A: HTML lab work
List of Exercises:

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

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

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

4) Design a page to display the information in table format. Display the list of colleges offering B.Sc. Computer Science along with the details College Name, Address, Contact no. Address column will consist of sub columns as House Name, Post, City and pin code.
(Purpose: - Introduction of table tags along with the sub columns and other supportive tags like caption, cell spacing, cell padding etc.)

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

6) Design web pages which display the product images and its information with it. The products are computer, printers and laptop. The information displayed of product should be configuration/ technical details, price etc.
(Purpose: - Study image tag. Display image in tabular form along with the other text information.)

7) Design a page to display the B.Sc. Computer Science syllabus by using List tag.
(Purpose: - Introduce list tag to display data in ordered or unordered format as main, sub main, sub-sub main, etc by using nos. or special types of bullets.)

8) Create HTML pages using Hyper Links

   i. File Link
   ii. Single Page Link

9) Create a hyperlink to show the information and syllabus of B.Sc. Computer Science. When click on the links each page should display the objective of respective course, Lesson plan, Course durationetc.
(Purpose: - Introduce anchor tag to create links between pages. One can able to transfer the control to next page, previous page or to a specific page like Home page.)

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

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

12) HTML pages with Tables

   i. Use Tables to layout HTML pages
   ii. Use <span> and <div> tags to provide a layout to the above page instead of table layout
   iii. Use Frames to divide the page contents into different parts
   iv. Embed Audio and Video into your HTML webpage

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

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

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

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

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

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

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

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

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

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

      i. Checkbox,

      ii. Radio button,

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

      i. Select box,

      ii. Hidden controls

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

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

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

Part B: C Programming

Write programs to do the following:

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

6. Solve the problem of Towers of Hanoi using recursion.
7. Find Sin(x) and Cos(x) in the range 0° to 180° (interval 30°) using functions.
8. Create a pyramid using „*‟.

9. Display the multiplication tables up to the number N.
10. Find the number of words in a sentence.
11. Perform matrix addition, subtraction, multiplication, inverse, and transpose using pointers and functions.
12. Replace a part of the string with another string.
13. Find the power of a number using structure and union.
14. Find the average of prime numbers in a group of N numbers using function.
15. Find the sum of the series $S = 1 + (½)^2 + (1/3)^3 + ...$ to 0.0001% accuracy.
16. Display the rightmost digit in a floating point number.
17. Create a pattern with the number N.

    e.g. N = 39174   Pattern: 3 9 1 7 4

                                                         9 1 7 4

                                                         1 7 4

                                                         7 4

                                                         4

18. Display the short form of a string. E.g. Computer Science : CS
19. Currency conversion (any four currencies)
20. Find the currency denomination of a given amount.
21. Prepare sales bill using array of structures.
22. Addition and subtraction of complex numbers using structure.
23. Find the amstrong numbers within a given range.
24. Check for palindrome string/number.
25. Check for leap year.
26. Find the number of special characters in a given string.
27. Store and read data from a text file.
28. Write odd and even numbers into separate files.
29. Swapping of two numbers using call-by-reference method.
30. Copy the contents of one file into another one using command line parameters.
31. Base conversion of numbers.
32. Calculate the percentage of marks obtained for N students appeared for examination in M subjects using array of structures.
33. Display a table of the values of function $y = exp(-x)$ for x varying from 0.0 to N in steps of 0.1
34. Design a Scientific Calculator and include as many functions as possible.
35. Merge two numeric arrays in sorted order.
36. Fill upper triangle with 1, lower triangle with -1 and diagonal elements with 0.
37. Count the occurrence of different words in a sentence.
38. Convert an input amount into words.
39. Convert a time in 24 hour clock to a time in 12 hour clock using structure.

40. Change the date/time format using structure.

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

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

SEMESTER III
General Course
GBCS3A01T: PYTHON PROGRAMMING

Course Number: 14
Contact Hours per Week: 4L
Number of Credits: 4
Number of Contact Hours: 64 Hrs
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Prerequisites:

- Knowledge of C language.

Course Outcomes:

COs	COURSE OUTCOMES
CO1	Understand various statements, data types and functions in Python
CO2	Develop programs in Python programming language
CO3	Understand the basics of Object oriented programming using Python
CO4	To learn basics of Python programming
CO5	To learn decision making, looping and functions in Python
CO6	Understand Object Oriented Programming using Python

Course Outline:

Unit I [16L]
Introduction to python, features, IDLE, python interpreter, Writing and executing python scripts, comments, identifiers, keywords, variables, data type, operators, operator precedence and associativity, statements, expressions, user inputs, type function, eval function, print function.

Unit II [16L]
Boolean expressions, Simple if statement, if-elif-else statement, compound boolean expressions, nesting, multi way decisions. Loops: The while statement, range functions, the for statement, nested loops, break and continue statements, infinite loops.

Unit III [16L]

Functions, built-in functions, mathematical functions, date time functions, random numbers, writing user defined functions, composition of functions, parameter and arguments, default parameters, function calls, return statement, using global variables, recursion.

Unit IV [16L]

String and string operations, List- creating list, accessing, updating and deleting elements from a list, basic list operations. Tuple- creating and accessing tuples in python, basic tuple operations. Dictionary, built in methods to access, update and delete dictionary values. Set and basic operations on a set.

References:

1. E. Balaguruswamy, Introduction to Computing and Problem Solving Using Python
2. Richard L. Halterman, Learning To Program With Python
3. Martin C. Brown, Python: The Complete Reference.

SEMESTER III
General Course

GBCS3A02T: DATA COMMUNICATION AND OPTICAL FIBERS

Course Number: 15
Contact Hours per Week: 4L
Number of Credits: 4
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To introduce the concept of data communication
CO2	Understand various types of communication and communication medium
CO3	To equip the students implement communication

Course Outline:

Unit I [16L]:
Introduction- Components, Networks, Protocols and standards, Basic Concepts: Line Configuration, Topology Transmission mode, analog and digital signals, Encoding and modulating-analog-to-digital conversion, digital to analog conversion, digital data transmission, DTE-DCE interface, modems, cable modems. Transmission media: guided media, unguided media, and transmission impairment.

Unit II [16L]
Multiplexing: Many to one/ one to many, frequency division multiplexing, wave division multiplexing, TDM, multiplexing applications: the telephone system, Cellular System, Mobile Communication-GSM, Mobile Services, GSM system Architecture, Radio Interface in GSM

Unit III [16L]
Data link Control: Line Discipline, flow control, error control, Data link Protocols: Asynchronous Protocols, synchronous protocols, character oriented protocols, bit – oriented protocols, link access procedures. Local Area Networks: Ethernet, token bus, token ring, FDDI, Comparison, Switching-circuit switching, packet switching, message switching, integrated services digital networks (ISDN): services, history, subscriber access to ISDN.

Unit IV [16L]
Overview of Optical Fiber Communication - Introduction, historical development, general system, advantages, disadvantages, and applications of optical fiber communication, optical fiber

waveguides, fiber materials, Optical Sources And Detectors- Introduction, LED"s, LASER diodes, Photo detectors. Ray theory, cylindrical fiber, single mode fiber, cutoff wave length, mode field diameter.

Text Book:
1. Behrouz A. Forouzan, Data Communication and Networking, TMH
2. Mobile Communications – Jochen H. Schiller , Second Edition ,Pearson
3. Optical Fiber Communication – Gerd Keiser, 4th Ed., MGH, 2008.

Reference Books:
1. William Stallings: Data & Computer Communications, 6/e, Pearson Education.
2. William L. Schweber : Data Communication, McGraw Hill.
3. Electronic Communication Systems - Kennedy and Davis, TMH
4. Optical Fiber Communications– – John M. Senior, Pearson Education. 3rd Impression,2007.
5. Fiber optic communication – Joseph C Palais: 4th Edition, Pearson Education.

SEMESTER III
Core Course

GBCS3B04T: DATA STRUCTURES USING C

Course Number: 16
Contact Hours per Week: 7 (3L + 4P)
Number of Credits: 3
Number of Contact Hours: 112 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Knowledge in C Programming Language

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To introduce the concept of data structures
CO2	To make the students aware of various data structures
CO3	To equip the students implement fundamental data structures
CO4	Basic knowledge of computers, data structures and programming

Course Outline:

Unit I [8 L + 8P]
Introduction: Elementary data organization, Data Structure definition, Data type vs. data structure, Categories of data structures, Data structure operations, Applications of data structures, Algorithms complexity and time-space trade off, Big-O notation.

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

Unit II [10 L + 14 P]
Arrays: Introduction, Linear arrays, Representation of linear array in memory, Traversal, Insertions, Deletion in an array, Multidimensional arrays, Parallel arrays, sparse matrix.

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

Unit III [10 L + 14 P]
Stack: primitive operation on stack, algorithms for push and pop. Representation of Stack as Linked List and array, Stacks applications: polish notation, recursion.

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

Unit IV [10 L + 14 P]
Trees - Basic Terminology, representation, Binary Trees, Tree Representations using Array & Linked List, Basic operation on Binary tree: insertion, deletion and processing, Traversal of binary trees: In order, Pre-order & post-order, Algorithm of tree traversal with and without recursion, Binary Search Tree, Operation on Binary Search Tree, expression trees, implementation using pointers, applications.

Unit V [10 L + 14 P]
Introduction to graphs, Definition, Terminology, Directed, Undirected & Weighted graph, Representation of graphs, graph traversal- depth-first and breadth-first traversal of graphs, applications.

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

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

Reference books
1. Trembley, J.P. And Sorenson P.G., "An Introduction to Data Structures With Applications", Mcgraw- Hill International Student Edition, New York.
2. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Addison- Wesley, (An Imprint of Pearson Education), Mexico City.
3. A.K.Sharma, Data Structures Using C, Pearson, Second edition, 2011
4. Nair A.S., Makhalekshmi, Data Structures in C, PHI, Third edition 2011.
5. R. Kruse etal, "Data Structures and Program Design in C", Pearson Education Asia, Delhi- 2002
6. K Loudon, "Mastering Algorithms with C", Shroff Publisher & Distributors Pvt. Ltd.

SEMESTER IV
General Course
GBCS4A03T: MICROPROCESSORS ARCHITECTURE AND PROGRAMMING

Course Number: 19
Contact Hours per Week: 4L
Number of Credits: 4
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Prerequisites:

- Basic knowledge of Computer.

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To understand internals of Microprocessor.
CO2	To learn architecture of 8085 Microprocessor
CO3	To learn instruction set of 8085 Microprocessor
CO4	To learn how to program a Microprocessor

Course Outline:

Unit I [16 L]
General architecture of computer, Introduction to Microprocessor, Memory classification, Introduction to 8085,Microprocessor bus organizations ,data bus, address bus, control bus. Memory addressing, memory mapping. 8085 architecture in detail. General purpose registers and special purpose registers, flag register -8085 pins and signals.

Unit II [16 L]
Assembly language programming basics. Opcode, Mnemonics etc. 8085 instruction set ,Data transfer ,Arithmetic and Logic, Shifting and rotating, Branching/Jump, Program control. Addressing modes. Memory read and write cycle. Timing diagram. Instruction cycle , machine cycle and T-states. Types of I/O addressing .Simple programs.

Unit III [16 L]
Types of programming techniques looping, indexing (pointers),delay generation. Stack in 8085, call and return Instructions. Data transfer between stack and microprocessor. Subroutine and delay

programs. Interrupts in 8085. Interrupt driven programs. Interfacing - Programmable peripheral devices - 8255A, 8254, 8237.

Unit IV [16 L]
Introduction to 8086/88 microprocessors – overview, 8086 internal architecture. The execution unit, BIU, Registers, Flags, Segmentation, physical address calculation, addressing modes.

Text Book:
1. Ramesh S. Gaonkar, Microprocessor Architecture Programming and Application with 8085, Prentice Hall
2. Doughles V Hall, Microprocessors and Interfacing: Programming and Hardware, Tata McGraw Hill

Reference Book:
1. Microprocessor and Microcomputer - Based system Design - M. Rafiquzzman - CRC press
2. 2.A.P Mathur, Introduction to Microprocessors, Tata McGraw-Hill Education
3. The Intel Microprocessors: 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium Pro, Pentium II, III, IV and Core 2 with 64 bit Extensions, Barry B. Brey, Prentice Hall Pearson
4. Microprocessors PC Hardware and Interfacing –N.Mathivanan – PHI

SEMESTER IV
General Course
GBCS4A04T: SENSORS AND TRANSDUCERS

Course Number: 20
Contact Hours per Week: 4L
Number of Credits: 4
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Course Outcome:

COs	COURSE OUTCOMES
CO1	Explain resistance, inductance and capacitance transducers.
CO2	Perceive the concepts of temperature and pressure transducers.
CO3	Perceive the concepts level transducers such as and flow transducers
CO4	Explain Electromagnetic transducers and radiation sensors
CO5	Explain force and torque transducers and sound transducers

Course Outline:

Unit I [16 L]
Transducers: Definition, Principle of sensing & transduction, Classification, Characteristics of transducers. Basic requirement of transducers.

Resistance Transducer: Basic principle – Potentiometer –Loading effects, Resistance strain gauge– Types.
Inductance Transducer: - Basic principle – Linear variable differential transformer – RVDT-types. Capacitance Transducer: Basic principle- transducers using change in area of plates – distance between plates- variation of dielectric constants –Types

Unit II [16 L]
Thermal sensors: Resistance change type: RTD - materials, construction, types, working principle, Thermistor - materials, construction, types, working principle, Thermo emf sensors: Thermocouple – Principle and types, Radiation sensors: Principle and types.
Pressure Transducers: basic principle- different types of manometers-u tube manometer-well type manometers.

Unit III [16 L]
Level transducer-continuous level measurement-discrete level measurement-mass – capacitive level gauges

Flow Transducers: Bernoulli‟s principle and continuity, Orifice plate, nozzle plate, venture tube, Rotameter, anemometers, electromagnetic flow meter, impeller meter and turbid flow meter

Unit IV [16 L]

Hall effect transducers, Digital transducers, Piezo-electric sensors, eddy current transducers, tacho generators and stroboscope, Magnetostrictive transducers
Radiation sensors: LDR, Photovoltaic cells, photodiodes, photo emissive cell types

Force and Torque Transducers: Proving ring, hydraulic and pneumatic load cell, dynamometer and gyroscopes.
Sound Transducers: Sound level meter, Microphone.

Text Books
1. D Patranabis, Sensors and Transducers, PHI, 2nd Edition.
2. E. A. Doebelin, Measurement Systems: Application and Design McGraw Hill, New York
3. A.K. Sawhney,- A course in Electrical & Electronic Measurement and Instrumentation, DhanpatRai and Company Private Limited.
4. Murthy D.V.S., ―Transducers and Instrumentation, 2nd Edition, Prentice Hall of India Private Limited, New Delhi, 2010.
5. S.Renganathan, ―Transducer Engineering, Allied Publishers, 2005

SEMESTER IV
Core Course

GBCS4B05T: DATABASE MANAGEMENT SYSTEM AND RDBMS

Course Number: 21
Contact Hours per Week: 7 (3L + 4P)
Number of Credits: 3
Number of Contact Hours: 112 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Basic knowledge of computers, data structures and programming

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn the basics of RDBMS
CO2	To learn the concepts of database manipulation SQL
CO3	To study PL/SQL language
CO4	Basic knowledge of computers, data structures and programming
CO5	To learn the basic principles of database and database design

Course Outline:

Unit I [8 L+ 2P]
Database System concepts and applications Introduction to databases, File Systems vs. DBMS, Advantages and Disadvantages of using DBMS Approach, Database administrators and user, Data Models, Schemas, and Instances, Types of Data Models, Three Schema Architecture and Data Independence, Database Languages and Interfaces.

Unit II [10 L+ 6P]
Entity-Relationship Model - Conceptual Data Models for Database Design Entity Relationship Models, Concept of Entity, Entity Sets, Relationship Sets, Attributes, Domains, Constraints, Keys, Strong and Weak Entities, Concepts of EER.

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

Unit III [10 L+12P]

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

Unit IV [10 L +20P]

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

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

Unit V [10 L+24P]

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

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

Textbooks:

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

References:

1. Alex Krigel and Boris M.Trukhnov, SQL Bible, Wiley pubs
2. Paul Nielsen, Microsoft SQL Server 2000 Bible, Wiley Dreamtech India Pubs.
3. CJ Date, Introduction to Database Systems, Addison Wesley.
4. Ramkrishnan, Database Management Systems, McGraw Hill

GBCS4B06P: PROGRAMMING LABORATORY II: DATA STRUCTURES AND RDBMS
(Lab Exam of GBCS3B04T & GBCS4B05T)

Course Number: 22
Number of Credits: 4
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Prerequisites:

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

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To make the students equipped to solve mathematical or scientific problems using C
CO2	To learn how to implement various data structures.
CO3	To provide opportunity to students to use data structures to solve real life problems.
CO4	Knowledge in operating computer.
CO5	Theoretical knowledge in Data structures.
CO6	Knowledge in Database management.

Course Outline:

Part A: Data structure – Lab Questions

1. Sort a given list of strings
2. Reverse a string using pointers.
3. Implement Pattern matching algorithm.
4. Search an element in the 2-dimensional array
5. Append 2 arrays
6. Merge two sorted array into one sorted array.
7. Search an element in the array using iterative binary search.
8. Search an element in the array using recursive binary search.
9. Implement sparse matrix

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

Part B: DBMS– Lab Questions

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

Customer_Master

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

Order_Master

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

Order_Detail

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

ITEM

Item_No	-	Primary Key
UnitPrice	-	Not NULL

SHIPMENT

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

WAREHOUSE

Warehouse_No	-	Primary Key
City	-	Not NULL

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

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

Create the following tables with the specified constraints:

Department:

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

Employee:

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

Department location:

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

Department number & Department location are combined Primary Key

Project:

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

Works-on:

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

Employee ID & Project Number are combined primary key.

Dependent:

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

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

    A. List the department wise details of all the employees.

    B. Find out all those departments that are located in more than one location.

    C. Find the list of projects.

    D. Find out the list of employees working on a project.

    E. List the dependents of the employee whose employee id is 001

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

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

Book Records:

Accession Number
ISBN Number

Books:

ISBN Number
Author
Publisher
Price

Members:

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

Book Issue:

Member ID
Accession Number
Issue Date
Return Date

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

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

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

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

   A. Get the list of ISBN-Number, Book name, available copies of the books of which available copies are greater than zero.
   B. Get the list of ISBN-Number, Book name, Total copies, available copies of the book of which available copies are greater than zero. List should be displayed in alphabetical order of book name.
   C. Get the list of ISBN number, Book name, Author, total copies, cost (cost is price total copies). List should be displayed in descending order of cost.
   D. Get the list of books issued to each member.
   E. Write query to know the maximum and average price of the books.
   F. Get the list of all existing members and the number of days for which a member is allowed to keep the book. Also find out the members who have got the maximum number of books issued.

G. Get the list of member codes of those members who have more than two books issued.
H. Find the details of the books presently issued to a member.
I. Create the history of issue of a book having a typical accession number.
J. To set the width of the book name to 35.

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

1. Create the following table named customer

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

Insert the appropriate data into table and do the following.

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

7. Answer the following queries using Library system as created earlier. You must create a view to know member name and name of the book issued to them, use any inbuilt function and operators like IN, ANY, ALL,EXISTS.
   a. List the records of members who have not been issued any book using EXISTS operator.
   b. List the members who have got issued at least one book (use IN / ANY operator).
   c. List the books which have maximum Price using ALL operator.
   d. Display Book Name, Member Name, and Issue date of Book. Create a view of this query of the currently issued books.

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

a. Display all records from employee table where department is not found in department table.
b. Display records from employee table in a report format with proper headings. This report must also contain those records where department number does not match with any value of department table.
c. Display those employee records who have salary less than the salary of person whose emp_number= A100.
d. Create another table: SalesData (RegionCode, City, SalespersonCode, SalesQty).
e. Display records where salesperson has achieved sales more than average sales of all sales persons of all the regions.

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

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

11. Implement student information system

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

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

14. Create a table product with the fields(Product_code primary key, Product_Name, Category, Quantity, Price). Insert some records Write the queries to perform the following.
    a. Display the records in the descending order of Product_Name
    b. Display Product_Code, Product_Name with price between 20 and 50
    c. Display the Product_Name and price of categories bath soap, paste, washing powder
    d. Display the product details whose Quantity less than 100 and greater than 500
    e. Display product names starts with 's'
    f. Display the products which not belongs to the category 'paste'
    g. Display the product names whose second letter is 'a' and belongs to the Category 'washing powder'

15. Create a STUDENT table with following fields:

Field Name	Type	Width
RegNo	Character	10
Name	Character	20
Paper1	Numeric	3
Paper2	Numeric	3
Paper3	Numeric	3
Paper4	Numeric	3
Paper5	Numeric	3
Total	Numeric	3
Result	Character	6
Grade	Character	1

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

16. An examination has been conducted to a class of 10 students and 4 scores of each student have been provided in the data along with their reg_no, name, total and avg_score. Assign null values to the fields total and average. Write Postgresql block to do the following

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

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

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

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

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

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

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

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

18. Generate a Hospital information system that can generate the following reports:
    - Patients who belongs to in-patient category
    - Patients who belongs to out-patient category

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

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

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

19. Using the Hospital table created in Lab 16, generate a Hospital information system that can generate the following reports:
    - Patients undergone blood test.
    - Patients who have taken X-Rays

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

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

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

23. Programs involving multiple tables.

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

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

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

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

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

SEMESTER V
Core Course
GBCS5B07T: COMPUTER ORGANIZATION AND ARCHITECTURE

Course Number: 25
Contact Hours per Week: 4 L
Number of Credits: 3
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Boolean algebra

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn logic gates, combinational circuits and sequential circuits
CO2	To learn basics of computer organization and architecture
CO3	Learns to analyse and develop a digital logic and apply it to solve real life problems.
CO4	Understands the theory and architecture of central processing unit and learns to analyse the design issues in terms of speed, technology, cost and performance.

Course Outline:

Unit I [12 L]
Digital Logic - Positive and negative logic, logic gates ,NOT gate, OR gate, AND gate, XOR and X-NOR gates, Universal gates- NAND gate, NOR gate,. Combinational circuits- Half adder, half subtractor, full adder, full subtractor, ripple carry adders, look-ahead carry adders, decoders, BCD to 7-segment decoder, encoders, multiplexers and demultiplexers.

Unit II [13 L]
Sequential Logic Circuits: Edge triggering, Pulse triggering ,SR latch, SR flip flop, JK flip flop, Master Slave JK flip flop, D flip flop, T flip flop. Shift register: serial in - serial out, serial in - parallel out, parallel in - serial out, parallel in-parallel out configurations. counters (asynchronous & synchronous), up/down counter, decade counter, mod N counter, Ring counter, Johnson"s counter

Unit III [13 L]
Basic Computer Organization and Design: Instruction Codes , Computer Registers, Computer Instructions, Instruction types, Timing and Control, Instruction Cycle, Memory reference

Instructions, Register reference instructions, Input, Output and Interrupt Design of Basic Computer, Design of Accumulator logic

Unit IV [13 L]
Micro programmed Control: Control Memory, Address sequencing, Micro program Example, Design of control unit. Processor Organization: general register organization, stack organization, instruction formats, addressing modes, data transfer and manipulation, program control

Unit V [13 L]
Memory Organization: Memory mapping, Associative memory, Cache memory, Virtual Memory, Memory Management Hardware, hit/miss ratio, Input-Output Organization: Peripheral devices, I/O interface, Modes of Transfer-asynchronous and synchronous, Priority Interrupt, Strobe Control, Handshaking. Direct Memory Access, Input-Output Processor, Serial Communication. I/O Controllers

Textbooks:

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

References:

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

SEMESTER V
Core Course
GBCS5B08T: JAVA PROGRAMMING

Course Number: 26
Contact Hours per Week: 6 (3L + 3P)
Number of Credits: 3
Number of Contact Hours: 96 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Knowledge in OOP & Programming

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To review on concept of OOP.
CO2	To learn Java Programming Environments.
CO3	To practice programming in Java.
CO4	To learn GUI Application development in JAVA

Course Outline

Unit I [9 L + 5 P]
Introduction to OOPS, Characteristics of OOPS, Object oriented languages, comparison between procedural and object oriented programming, basic principles of Object Orientation-class, object, abstraction, encapsulation, inheritance, polymorphism, modularity, and message passing. Features of object orientation - attributes, state, identity, operation, behavior.

Unit II [9 L + 8 P]
Introduction to Java: History, Versioning, The Java Virtual Machine, Byte code, Writing simple java program, Language Components: Primitive Data Types, Comments, Keywords, literals, The break Statement, The continue Statement, Operators – Casts and Conversions, Arrays. Introduction to classes and methods, constructors, Passing Objects to Methods, Method Overloading, Static and final, The this Reference, finalize, inner and nested classes. Inheriting class, extends, member access and inheritance, super keyword, Object class. Dynamic method dispatch, method overriding, abstract class, interface, packages, import statement.

Unit III [10 L + 12 P]

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

Unit IV [10 L + 12 P]

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

Unit V [10 L + 11 P]

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

Text Books

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

References

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

SEMESTER V
Core Course
GBCS5B09T: WEB PROGRAMMING USING PHP

Course Number: 27
Contact Hours per Week: 6 (3L + 3P)
Number of Credits: 3
Number of Contact Hours: 96 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

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

Prerequisites:

- Knowledge in OOP & Programming

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To review on concept of java script.
CO2	To learn PHP Programming Environments.
CO3	To practice web development using html, java script and PHP
CO4	To learn GUI Application development in PHP

Course Outline:

Unit I [9 L + 9P]
Introduction web-documents: Static, Dynamic, Active - Web programming: client side and server side scripting. HTML 5: Document Structure, Elements, Attributes, Types of Elements and Attributes, Basic HTML Data types. Using HTML5 form elements: datalist, keygen, output, progress, meter. File uploading using forms - Frameset and frames. CSS: External CSS, CSS3 Syntax, Selector: Universal, Class, ID. Working with Lists and Tables, CSS ID and Class – Navigation Bar - Image Gallery – Image Opacity.

Unit II [9 L + 9P]
Javascript: Introduction, Client side programming, script tag, comments, variables. Including JavaScript in HTML: head, body, external. Data types. Operators: Arithmetic, Assignment, Relational, Logical. Conditional Statements, Loops, break and continue. Output functions: write, writeln, popup boxes: prompt, alert, confirm. Functions: Built-in Global Functions: alert(),

prompt(), confirm(), isNan(), Number(), parseInt(). User Defined Functions, Calling Functions with Timer, Events Familiarization: onLoad, onClick, onBlur, onSubmit, onChange, Document Object Model (Concept). Objects: String, Array, Date.

Unit III [10 L + 10 P]
PHP: Introduction, Server side programming, Role of Web Server software, Including PHP Script in HTML: head, body, external. Comments, Data types, variables and scope, echo and print. Operators: Arithmetic, Assignment, Relational, Logical. Conditional Statements, Loops, break and continue. User Defined Functions.

Unit IV [10 L + 10 P]
Working with PHP: Passing information between pages, HTTP GET and POST method, Cookie, Session. String functions: strlen, strops, strstr, strcmp, substr, str_replace, string case, Array constructs: array(),list() and foreach(). Header().

Unit V [10 L + 10 P]
PHP & PostgreSQL: Features of PostgreSQL, data types, PostgreSQL commands – CREATE DATABASE, CREATE TABLE, DESCRIBE TABLE (\d table_name or using usinginformation_schema), SELECT, SELECT INTO, CREATE AS, DELETE, UPDATE, INSERT. PHP - PostgreSQL Integration: Establishing Database Connection (pg_connect(), pg_connection_status(), pg_dbname()), Getting Error String (pg_last_error()), Closing database Connection (pg_close()), Executing SQL statements (pg_query(), pg_execute()), Retrieving Data (pg_fetch_row(), pg_fetch_array(), pg_fetch_all(), pg_fetch_assoc(), pg_fetch_object(), pg_num_rows(), pg_num_fields() pg_affected_rows(), pg_num_rows(), pg_free_result())), Insertion and Deletion of data using PHP, Displaying data from PostrgreSQL database in webpage. Introduction to AJAX - Implementation of AJAX in PHP - Simple example for partial page update.

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

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

SEMESTER V

Core Course

GBCS5B10T - PRINCIPLES OF SOFTWARE ENGINEERING

Course Number: 28
Contact Hours per Week: 4L
Number of Credits: 3
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Knowledge in Programming

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn engineering practices in Software development.
CO2	To learn various software development methodologies and practices.
CO3	To learn and study various Evaluation methods in Software Development.

Course Outline:

Unit I [13L]
Software and Software Engineering: Overview of Software Engineering, Practice & Myths; Software Process; Generic process model- Framework Activity, Task Set, Process Patterns, Process Improvement; SDLC , Prescriptive process model- Waterfall Model, Spiral Model, Incremental Process Model, Evolutionary Process Model; Specialized Process Models: Component Based Development, the Formal Methods Models.

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

Unit II [13L]
Requirements Engineering- Establishing the Groundwork- Eliciting Requirements - Developing use cases - Building the requirements model - Negotiating, validating Requirements - Requirements Analysis-Requirements Modeling Strategies.

Unit III [14L]
MODELING WITH UML: Concepts and Diagrams - Use Case Diagrams - Class Diagrams - Interaction Diagrams - State chart Diagrams – Activity Diagrams - Package Diagrams - Component Diagrams - Deployment Diagrams -Diagram Organization- Diagram Extensions. Design Process-

Design concepts: Abstraction, Architecture, patterns, Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Aspects, Refactoring, Object Oriented Design Concepts, Design Classes- Design Model: Data, Architectural, Interface, Component, Deployment Level Design Elements.

Unit IV [11L]
Structured coding Techniques-Coding Styles - Standards and Guidelines-Documentation Guidelines-Modern Programming Language Features: Type checking-User defined data types-Data Abstraction Exception Handling - Concurrency Mechanism.

Unit V [13L]
TESTING: Software Quality- Software Quality Dilemma- Achieving Software Quality- Testing: Strategic Approach to software Testing- Strategic Issues - Testing: Strategies for Conventional Software, Object oriented software, Web Apps-Validating Testing- System Testing- Art of Debugging.

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

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

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

<u>Open Courses</u> (GBCS5DXXT)
<u>(For students of other UG Programmes)</u>

GBCS5D01T: INTRODUCTION TO COMPUTERS AND OFFICE AUTOMATION

Contact Hours per Week: 3L
Number of Credits: 3
Number of Contact Hours: 48 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Basic knowledge in Computer & Internet.

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn Office Automation.
CO2	Basic knowledge in Computer & Internet
CO3	To learn about application software

Course Outline:

Unit I [12L]
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 [12L]
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 [12L]
Electronic Spread Sheet (Open Office Calc/MS-Excel) - Introduction to Spread Sheet, Creating & Editing Worksheet, Formatting and Essential Operations, Formulas and Functions, Charts, Advanced features – Pivot table & Pivot Chart, Linking and Consolidation.

Unit IV [12L]
Presentation using (OpenOffice Impress/MS-Power Point): Presentations, Creating, Manipulating & Enhancing Slides, Organizational Charts, Charts, Word Art, Layering art Objects, Animations and Sounds, Inserting Animated Pictures or Accessing through Object, Inserting Recorded Sound Effect or In-Built Sound Effect.

References:

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

SEMESTER V
Open Course
GBCS5D02T: WEB DESIGNING

Contact Hours per Week: 3L
Number of Credits: 3
Number of Contact Hours: 48 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Basic knowledge in Computer & Internet.

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn Web Designing.
CO2	Basic knowledge in Computer & Internet
CO3	To learn about web designing tools

Course Outline:

Unit I [12L]
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 [12L]
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 [12L]
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 [12L]

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

Reference:

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

SEMESTER V
<u>Open Course</u>
GBCS5D03T: INTRODUCTION TO PROBLEM SOLVING AND C PROGRAMMING

Contact Hours per Week: 3 L
Number of Credits: 3
Number of Contact Hours: 48 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Background of the basic science at +2 level

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To introduce fundamental principles of Problem Solving aspects.
CO2	To learn the concept of programming.
CO3	To learn C language.

Course Outline:

Unit I [12L]
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 [12L]
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 [12L]

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 [12L]

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

References:

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

SEMESTER V
Open Course
GBCS5D04T: INTRODUCTION TO DATA ANALYSIS USING SPREAD SHEET

Contact Hours per Week: **3L**
Number of Credits: **3**
Number of Contact Hours: **48 Hrs.**
Course Evaluation: **Internal: 15 Marks + External: 60 Marks**

Prerequisites:

- Background of the Basic science and statistics at +2 level

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To introduce fundamental principles of spread sheet
CO2	To introduce the importance of software tools.
CO3	To learn the Analysis using Spreadsheets.

Course Outline:

Unit I [12L]
Introduction to MS Excel and Understanding Basic Working with it: Quick review on MS Excel Options, Ribbon, Sheets , Difference between Excel 2003, 2007, 2010 and 2013- Saving Excel File as PDF, CSV and Older versions - Using Excel Shortcuts - Copy, Cut, Paste, Hide, Unhide, and Link the Data in Rows, Columns and Sheet Using Paste Special Options - Formatting Cells, Rows, Columns and Sheets - Protecting & Unprotecting Cells, Rows, Columns and Sheets with or without Password - Page Layout and Printer Properties - Inserting Pictures and other objects in Worksheets

Unit II [12L]
Introduction to Pivot table: Use multiple pivot tables and pivot charts to create dashboard, Connect multiple slicers to the pivot tables.

Unit III [12L]
Pivot table applications in analytics: filter the data shown in the pivot in different ways to achieve subsets of the data, Use calculated fields on top of the pivot table to calculate profitability and find anomalies.

Unit IV [12L]

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

Text Books:

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

References:

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

SEMESTER VI
Core Course
GBCS6B11T: ANDROID PROGRAMMING

Course Number: 30
Contact Hours per Week: 7 Hrs
Number of Credits: 3
Number of Contact Hours: 112 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Knowledge in OO & Java Programming.

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To have a review on concept of Android programming.
CO2	To learn Android Programming Environments.
CO3	To practice programming in Android.
CO4	To learn GUI Application development in Android platform with XML
CO5	To get Knowledge in OO & Java Programming.

Course Outline:

Unit I [19L+3P]
Introducing the android computing platform, History of android, an- droid software stack, Developing end user application using Android SDK, Android java packages, Setting up the development environment, Installing android development tools(ADT), Fundamental components, Android virtual devices, Running on real device, Structure of android application, Application life cycle.

Unit II [19L+3P]
Understanding android resources - String resources, Layout resources, Resource reference syntax, Defining own resource IDs - Enumerating key android resources, string arrays, plurals, Colour resources, dimension resources, image resources, Understanding content providers - Android built in providers, exploring databases on emulator, architecture of content providers, structure of android content URIs, reading data using URIs, using android cursor, working with where clause, inserting updates and deletes, implementing content, Understanding intents basics of intents,

available intents, exploring intent composition, Rules for Resolving Intents to Their Components, ACTION PICK, GET CONTENT, pending intents

Unit III [19L+3P]
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 [19L+4P]
Android menus - creating menus, working with menu groups, responding to menu items, icon menu, sub menu, context menu, dynamic menus, loading menu through XML, popup menus, Fragments in Android structure of fragment, fragment life cycle, fragment transaction and back stack, fragment manager, saving fragment state, persistence of fragments, communications with fragments, startActivity() and setTargetFragment(), using dialogs in android, dialogfragments, working with toast, Implementing action bar - tabbed navigation action bar activity, implementing base activity classes, tabbed action bar and tabbed listener, debug text view layout, action bar and menu interaction, list navigation action bar activity, spinner adapter, list listener, list action bar, standard navigation action bar activity, action bar and search view, action bar and fragments.

Unit V [19L+4P]
Persisting data - Files, saving state and preferences - saving application data, creating, saving and retrieving shared preferences, preference framework and preference activity, preference layout in XML, native preference controls, preference fragments, preference activity, persisting the application state, including static files as resources, Working with file system, SQL Lite – SQL Lite types, database manipulation using SQL Lite, SQL and database centric data model for Android, Android database classes.

References:
1. Satya Komatineni & Dave MacLean,*Pro Android 4*, Apress.
2. Retomeier, *Professional Android 4 Application Development*, Wrox.
3. Zigurd Mednieks, Laird Dornin, G. Blake Meike, and Masumi Nakamura, *Programming Android*, O'Reilly

SEMESTER VI
Core Course
GBCS6B12T: OPERATING SYSTEMS

Course Number: 31
Contact Hours per Week: 7 Hrs
Number of Credits: 3
Number of Contact Hours: 112 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Knowledge in Data structures.

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn objectives & functions of Operating Systems.
CO2	To understand processes and its lifecycle.
CO3	To learn and understand various Memory management techniques and Scheduling Algorithms.
CO4	To have an overall idea about the latest developments in Operating Systems
CO5	To get Knowledge in Data structures

Course Outline:

Unit I [19L + 4P]
Operating System - Objectives and functions - The Evolution of Operating Systems: Serial Processing, Simple batch Systems, Multi Programmed batch Systems, Time Sharing Systems, Parallel Systems, Distributed Systems, Real time systems. Definition of Process, Process States, Process Control Block, Operations on Process, Process Communication, Communication in Client server System, Basic concepts of threads, Concurrency, Principles of Concurrency, Mutual exclusion, Semaphores, Messages, Dead lock: Prevention, Detection, Avoidance.

Unit II [19 L + 3P]
Linux Shell Programming: Introduction – Shells available in Unix: Bourne shell (sh), C shell (csh), TC shell (tcsh), Korn shell (ksh), Bourne Again SHell (bash). Bash: special characters – getting help – man pages – Linux Directory Layout – Command for Navigating the Linux Filesystems: pwd, cd, ls, file, cat, cp, mv, mkdir, rmdir, whereis – Piping and Redirection - Informational Commands: ps, w, id, free – clear, echo, more. File permissions – Setting Permissions – Making a file executable. Creating shell programs: comments, variables, operators (arithmetic, relational,

logical) – single and double quotes - read – echo – test **-** conditional commands, iterative commands – break – continue - evaluating expressions using expr, bc – strings – grep – arrays.

Unit III [19L + 3P]
CPU Scheduling: Scheduling Criteria, Scheduling algorithms: FCFS, SJF, Priority, RR, Multilevel, Feedback Queue - Process synchronization, The Critical Section Problem, Synchronization Hardware, Classical Problems of Synchronization: Reader Writer, Dining Philosopher. File and Database System, File System, Functions of organization, Allocation and Free Space Management.

Unit IV [19L + 3P]
Memory Management, Address Binding, Logical Vs Physical Address Space, Dynamic Loading, Dynamic Linking and Shared Libraries, Overlays, Swapping, Contiguous Memory allocation, Paging, Segmentation, Virtual memory, Demand Paging, Page Replacement, Thrashing.

Unit V [19 L + 4 P]
Protection and security: policy and mechanism, authentication, authorization. Mobile OS: Concepts, history, features, architecture, future scope. Case studies: Android, UNIX kernel and Microsoft Windows NT (concepts only).

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

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

SEMESTER VI
Core Course
GBCS6B13T: COMPUTER NETWORKS

Course Number: 32
Contact Hours per Week: 5L
Number of Credits: 3
Number of Contact Hours: 80 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

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

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn about transmissions in Computer Networks.
CO2	To learn various Protocols used in Communication.
CO3	To have a general idea on Network Administration

Course Outline:

Unit I [16 L]
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, modems.

Unit II [16L]
Data link layer, Error detection and correction, Types of errors, Single CSC error and Burst error, Vertical redundancy check (VRC), longitudinal redundancy Check (LRC), Cyclic Redundancy Check(CRC), Error correction - Single CSC error correction, Hamming code Data compression - Huffman code, data link control, Line discipline, Flow control, Error control, Multiple Access, Random Access, ALOHA, pure ALOHA and slotted ALOHA, CSMA/CD and SCMA/CA, Polling, Wired LANs, Ethernet - IEEE standards, Wireless LANs.

Unit III [16L]
Network layer, Networking and Internetworking devices - Repeaters, Bridges, Routers, Gateways, Logical addressing - IPv4 & IPv6 addresses, Network Address Translation(NAT), Internet

protocols, internetworking, Datagram, Transition from IPv4 to IPv6, Address Mapping-Error reporting and multicasting - Delivery, Forwarding and Routing algorithms, Distance Vector Routing, Link State Routing.

Unit IV [16 L]
Transport layer, Process-to-process Delivery: UDP, TCP and SCTP, Congestion control and Quality of Service, Application Layer, Domain Name Systems-Remote Login-Email FTP, WWW, HTTP, Introductory concepts on Network management: SNMP.

Unit V [16L]
Cryptography and Network Security: Introduction – Goals of Security – Attacks - Services and Techniques. Basics of Cryptography: Plain Text - Cipher Text – Encryption – Decryption. Confidentiality: Basics of Symmetric Key Ciphers - Traditional Symmetric Key Ciphers: Substitution, Transposition, Stream & Lock, Modern – Components of Modern Block Cipher – DES - Modern Stream Cipher. Basics of Asymmetric Key Ciphers – RSA Cryptosystem. Integrity: Message – Message Digest – Hash Function. Authentication: MAC. Digital Signature : Analogy with Manual Signature – Process – Signing the Digest – Services – RSA Digital Signature Scheme.

Textbook:

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

References:

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

GBCS6B14P: PROGRAMMING LABORATORY III: JAVA AND PHP PROGRAMMING
(Lab Exam of GBCS5B08T & GBCS5B09T)

Course Number: 33

Number of Credits: 4

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

Prerequisites:

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

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To practice Java programming.
CO2	To practice client side and server side scripting.
CO3	To practice PHP Programming.
CO4	To practice developing dynamic websites.
CO5	To practice how to interact with databases through PHP.

Course Outline:

Part A: Java Programming:

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

c) Integer to Hexadecimal

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

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

Part B: PHP Programming

HTML and CSS

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

Top Frame	
Menu frame	**Details frame**

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

JavaScript

11. Write a javascript program to perform find the area and circumference of a circle.

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

PHP

25. Write a PHP program to check whether the given number is Armstrong or not.
26. Write a PHP program to check whether a given number is perfect, abundant or deficient.
27. Display the Fibonacci series up to a given number.
28. Create a php program to display the bio data of a person by reading the personal details using an HTML page.
29. Create a login page using database.
30. Create a mysql table student with fields roll no, name, mark, grade. Insert records in the table. Write a PHP program to display the mark list of a student by accepting the register no of the student.
31. Write a php application to generate the pay slip of an employee by accepting name, basic salary and designation. The net salary will be calculated based on the following conditions.

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

HRA – 25 %

Income tax

Gross <=200   0

2500 < gross <=4000   3%

4000< gross <=5000   5%

Gross >5000   8%

Gross= basic + HRA + conveyance + extra

Net = gross – income tax

32. Create a table "product" with fields itemcode, itemname, unitprice. Write php program to insert 5 records into the table and display it in a table format.

33. Write a php program for delete and update operation on account table. The account table contain fields such as accountno, name and amount.

34. Write an HTML page to display a list of fruits in a list box. Write php program to display the names of the fruits which are selected by the user.

35. Write php program to store current date/time in a cookie and display the „last visited on„ date time on the web page upon reopening of the same page.

36. Design a PHP page to implement a login screen using sessions. Login details are to be verified from the server side with values stored in a database.

37. Write a php program to create an array and store 10 names in the array. Do the following operations.

    a. Display the contents using for each statement.
    b. Display the array in a sorted order.
    c. Display the array without the duplicate elements
    d. Remove the last element and display
    e. Display the array in reverse order
    f. Search an element in the given array.

38. Design a PHP page to illustrate the use of **keygen** HTML element.

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

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

Android versions eclipse, studio etc. may be used for doing the lab works.

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

SEMESTER VI
Core Course
GBCS6B15P: PROGRAMMING LABORATORY IV: ANDROID AND LINUX SHELL PROGRAMMING
(Lab Exam of GBCS6B11T & GBCS6B12T)

Course Number: 34
Contact Hours per Week: 6 P
Number of Credits: 4
Number of Contact Hours: 96 Hrs.
Course Evaluation: Internal: 20 Marks + External: 80 Marks

Prerequisites:

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

Course Outcome:

COs	COURSE OUTCOMES
CO1	To practice Android programming.
CO2	To practice user interface applications.
CO3	To develop mobile application.
CO4	To practice shell programming
CO5	Theoretical knowledge in Android programming.
CO6	Theoretical knowledge of Shell Programming.

Course Outline:

Part A: Android Programming

1. Programs to understand basic arithmetic operations
2. Programs to understand basic logic operations
3. Programs to understand loops and control statements
4. Programs to understand GUI in android
5. Android application for adding two numbers
6. Develop simple user interface to display message
7. Create two menu items-opening a file-saving a file
8. Inserting values into Spinner control using Text view and Button.

9. Implementation of background image
10. Starting another activity from your own activity using intent
11. Create a new activity that services ACTION-PICK for contact data which display each of the contact in the contact database and lets the user to select one before closing and returning the selected contacts URL to the calling activities
12. Create Android application to linkify a text view to display web and E-mail address as hyperlinks. When clicked they will open the browser and E-mail address respectively
13. Implementation of array adapter
14. Create an alert dialogs used to display a message and offer two button options to continue. Clicking either button will close the dialog after executing the attached click listener
15. Get data from Text view control and insert into database using SQLite. Another activity shows inserted data in a List View control
16. Load menu item by parsing XML data.
17. Program to implement simple calculator
18. Program to Get IP Address
19. Program to Home and Lock Screen Widget (Temperature Widget)
20. Create a new contact using intent
21. A Button control shows Date picker and Text view control displays selected date.
22. Insert data into Spinner and delete selected item using SQLite.
23. Program to create simple login screen.
24. Create an Android application to display the map of your locality. Use ACTION_VIEW intent by passing latitude and longitude as parameters.
25. Create an Android application to convert a voice into text (using Google Speech to Text service)
26. Create an Android application to populate a list view by getting names & numbers from a SQLite database table.
27. Display the saved contacts available in the android phone in a listview using content providers
28. Create an image grid. Images should be placed under the resources section.
29. Create an android app with three tabs. First tab should contain two Edit text and that should accept age and name. In the second tab you need another 3 edit texts that accept education address and phone number. After these information is proved, when the third tab is open it should display all the provided information neatly.
30. Create a custom toast with an Image and a TextView.
31. Apply a Custom List style to a ListView. ListView should have at least 10 Items.
32. Determine the acceleration of your android device along all three axes using accelerometer? (i.e. x,y,z).
33. Capture an Image from the primary camera of an android device and save that picture into the internal storage.
34. Create an app to list files under a given folder name in an EditText
35. Fetch data from an arbitrary URL given in an EditText and display it in a TextView

36. Create an SQLite database named student. Accept student details from the MainActivity and save it in the table called student. Display the calculated result in the second activity when a button on the MainActivity is clicked.
37. Create an android app to switch the wifi on and off also illustrate the use of permission in android?
38. Create a spinner that takes data from the String.xml file.
39. Create a simple android application that opens the default messaging application available in the android device?
40. Create an app to display message in the notification bar?

Part B: Shell Programming

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

25. Write a shell script to compare two files and remove one of them if they are same

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

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

GBCS6B16D: INDUSTRIAL VISIT & PROJECT WORK

Course Number: **36**
Contact Hours per Week: **2P in V Sem & 2P in VI Sem**
Number of Credits: **3**
Number of Contact Hours: **64 Hrs.**
Course Evaluation: Internal: 15 Marks + External: 60Marks (Project: 50 + Ind. Visit: 10)

Prerequisites:

- Basic programming and system development knowledge

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To practice programming skills.
CO2	To practice user interface applications.
CO3	To provide practical knowledge on software development process
CO4	Basic programming and system development knowledge

Course Outline:

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

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

Students are encouraged to work on a project preferably on a live software project sponsored by industry or any research organization. Topics selected should be complex and large enough to

justify as a B.Sc. Computer Science final semester project. The courses studied by the students during the B.Sc. Computer Science

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

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

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

Industrial Visit:

Guide Lines:

• Minimum one day visit to National research Institutes, Laboratories, places of scientific Importance or Software Companies.

OR

• One week Industrial Training / internship at any software firms/ Research Labs

• The Industrial visit should be done in fifth or sixth semester.

• A 10 – 20 page Industrial visit / Training report have to be submitted with certificate from industry / institute, sufficient photos and analysis along with Project for evaluation in the sixth semester.

SEMESTER VI
Elective Course
GBCS6E01T: SYSTEM SOFTWARE

Course Number: 35

Contact Hours per Week: 4L

Number of Credits: 3

Number of Contact Hours: 64 Hrs.

Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites

- Basic knowledge in Programming

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To build fundamental knowledge in system software.
CO2	To learn functions of various system software.
CO3	To learn specifically learn compilation process of a program.

Course Outline:

Unit I [12L]
System software: General concept, Assemblers, loaders, linkers, macros, compilers, interpreters, operating system, Design of assemblers.

Unit II[13L]
Macros and macro processors, Macro definitions and instructions, Macro calls, Features of Macros, Design of Macro processors.

Unit III [13L]
Loading, linking and relocating Loader schemes- Binders, linking loaders, overlays, dynamic binders- Dynamic loading and dynamic linking – Relocatability of programs.

Unit IV [13L]
Compilers - Phases of a compiler - Lexical, Syntax, Intermediate code generation, Optimization, Code generation, Symbol table and error correcting routines – Passes of a compiler.

Unit V [13L]
Case studies of lexical and syntax analyzers: LEX and YAAC.

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

SEMESTER VI
Elective Course
GBCS6E02T: MACHINE LEARNING

Course Number: 35
Contact Hours per Week: 4L
Number of Credits: 3
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To familiarize with the basic concepts of Linear Algebra, Probability Theory for Machine Learning.
CO2	Introduce Machine Learning to the graduates and enable them think more scientifically
CO3	The students will be able to understand machine learning algorithms
CO4	They also get the essential mathematical and statistical foundations of machine learning

Course Outline:

Unit I [12 L]
Basics of Linear Algebra for ML : Classes of spaces (vector , metric , normed , inner product), Pythagorean Theorem, Type of matrices, Matrix operations, Eigenvector, Fundamental Theorem of Linear Algebra.

Unit II [13 L]
Foundations of Probability for ML : Probability Theory (Random Variables, Distributions, Mean and Variance, Bayes Rule), Basic Techniques (Naive Bayes, Nearest Neighbor Estimators, K-Means), Density Estimation (Limit Theorems, Parzen Windows, Exponential Families, Estimation, Sampling).

Unit III [13 L]
Introduction to Machine Learning : Applications, Issues in Machine Learning, Designing a Learning system - Supervised Learning - Unsupervised Learning, Vapnik-Chervonenkis Dimension – Probably Approximately Correct (PAC) Learning- Learning Multiple Classes, Bayesian Decision theory- Classification, Discriminant Functions, Association rules.

Unit IV [13 L]
Dimensionality Reduction : Subset Selection-Principal Component Analysis-Linear Discriminant Analysis, Clustering- K-means Clustering- Expectation maximization Algorithm- Hierarchical Clustering,

Unit V [13 L]

Parametric and Non-Parametric methods for classification and regression : Parametric methods-Maximum Likelihood Estimation- Baye"s Estimator-Parametric Classification, Regression ,Non-Parametric methods- Kernel Estimator- K-nearest neighbour estimator, Decision Trees-Univariate Trees-Classification trees, Regression trees, Rules extraction from Trees

References

1. Ethem Alpaydın, Introduction to Machine Learning, Second edition, MIT Press, 2010.
2. Alex Smola and S.V.N. Vishwanathan ,Introduction to Machine Learning, Second Edition, Cambridge University Press
3. Jason Brownlee, Basics of Linear Algebra for Machine Learning, First Edition, Machine Learning Mastery, 2018
4. Bishop. C M, Pattern Recognition and Machine Learning. Springer, 2006.
5. Duda, R O, Hart P E and Stork D G. Pattern Classification. Wiley-Interscience, 2nd Edition, 2000.
6. Hastie T, Tibshirani R and Friedman J, The Elements of Statistical Learning: Data Mining, Inference and Prediction. Springer, 2nd Edition, 2009.
7. Mitchell T, Machine Learning. McGraw Hill, 1997.
8. Mohssen Mohammed, Muhammad Badruddin Khan ,Eihab Bashier Mohammed Bashier, Machine Learning Algorithms and applications, CRC Press, First Edition,2017

Online References

MIT Open Courseware, Stanford Machine Learning Courses, IISc Machine Learning, IIT Kharaghpur, Bombay and Delhi ML Courses,

GBCS6E03T: DISCRETE STRUCTURES

Course Number: 35
Contact Hours per Week: 4L
Number of Credits: 3
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn mathematical logic and Boolean algebra.
CO2	Ability to apply mathematical logic to solve problems.
CO3	Understand sets, relations, functions and discrete structures
CO4	Able to use logical notations to define and reason about fundamental mathematical concepts such as sets relations and functions
CO5	Able to model and solve real world problems using graphs and trees

Course Outline:

Unit I [13 L]
Prepositional Calculus - Propositions, Logical Connectives, Disjunction, Conjunction, Negation, Conditional Connectives, Precedence Rule, Logical Equivalence, Logical Quantifiers. Methods of Proof - What is a Proof? Different Methods of Proof, Direct Proof, Indirect Proofs, Counter Examples, Principle of Induction. Boolean Algebra and Circuits – Boolean Algebras, Logic Circuits, Boolean Functions.

Unit II [12 L]
Sets, Relations and Functions: Introducing Sets, Operations on Sets, Basic Operations, Properties Common to Logic and Sets Relations, Cartesian Product, Relations and their types, Properties of Relations, Functions, Functions, Operations on Functions.

Unit III [13 L]
Combinatorics - An Introduction, Multiplication and Addition Principles, Permutations, Permutations of Objects not Necessarily Distinct, Circular Permutations, Combinations, Binomial Coefficients, Combinatorial Probability

Unit IV [13 L]

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

Unit V [13 L]

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

Text Book:
1. Elements of Discrete Mathematics By C.L.Liu - Mograw-Hills.

References:
1. Applied Discrete Structure for Computer Science By Alan Doerr And Kenneth Levassur- Galgotia Publication.
2. Discrete Mathematics by J.K. Sharma, Macmillan India Ltd.

SEMESTER VI
Elective Course

GBCS6E04T: COMPUTER GRAPHICS

Course Number: 35
Contact Hours per Week: 4L
Number of Credits: 3
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To learn the basics of Computer Graphics
CO2	Basic knowledge in Mathematics and Computer fundamentals

Course Outline:

Unit I [13 L]
Introduction to Computer Graphics Definition, Application, Pixel, Frame Buer, Raster and Random Scan display, Display devices CRT, Color CRT Monitors, basics of LCD & LED Monitors.

Unit II [12 L]
Scan Conversion of line DDA algorithm of line drawing, Scan conversion of circles Bresenham's circle generating algorithm, Polygon FillingScan line polygon filling algorithm.

Unit III [13 L]
Two Dimensional transformation, Translation, Rotation, Scaling, Homogeneous Coordinates, Reflection, Shear.

Unit IV [13 L]
Window to view port transformation, clipping, line clipping, Cohen Sutherland line clipping, Polygon clipping, Sutherland and Gary Hodgman polygon clipping algorithm.

Module V [13 L]
Color Models &Color Applications Light and Color, Different color models, RGB, CMY, YIQ. Introduction to GIMP Image Manipulation using GIMP.

Textbook:

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

References:

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

SEMESTER VI
Elective Course
GBCS6E05T: TECHNICAL WRITING

Course Number: 35
Contact Hours per Week: 4L
Number of Credits: 3
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Prerequisites:

- Basic communication skills in English.

Course Outcomes:

COs	COURSE OUTCOMES
CO1	To familiarize with the basic concepts of technical writing
CO2	Basic communication skills in English
CO3	The students will be able to Acquire the skills and knowledge for professional technical communication, web content writing
CO4	Develop soft skill development and search engine optimization

Course Outline:

Unit I [13 L]
Basics of Technical Communication: Technical Communication - features; Distinction between General and Technical communication; Language as a tool of communication; Levels of communication: Interpersonal, Organizational, Mass communication; the flow of Communication: Downward, Upward, Lateral or Horizontal (Peer group); Barriers to Communication.

Unit II [13 L]
Constituents of Technical Written Communication: Word formation, Prefix and Suffix; Synonyms and Antonyms; Homophones; One Word Substitution; Technical Terms; Paragraph Development: Techniques and Methods -Inductive, Deductive, Spatial, Linear, Chronological etc; The Art of Condensation- various steps.

Unit III [12 L]
Forms of Technical Communication - Business Letters: Sales and Credit letters; Letter of Enquiry; Letter of Quotation, Order, Claim and Adjustment Letters; Memos, Notices, Circulars; Job application and Resumes.

Unit IV [13 L]

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

Unit V [13 L]

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

Text Books

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

References

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

SEMESTER VI
Elective Course
GBCS6E06T: FUNDAMENTALS OF LIFE SKILL EDUCATION

Course Number: 35
Contact Hours per Week: 4L
Number of Credits: 3
Number of Contact Hours: 64 Hrs.
Course Evaluation: Internal: 15 Marks + External: 60 Marks

Course Outcomes:

COs	COURSE OUTCOMES
CO1	Develop intra-personal, inter-personal, critical thinking, decision making and communication skills
CO2	Get an insight to career planning and development
CO3	Establish self-management and help to maintain work life balance

Course Outline:

Unit I [13 L]
Introduction to life skill education, definition, components, pillars of learning, need for life skill training, approaches - critical thinking skills/decision making skills, interpersonal/communication skills, criteria for using life skills.

Unit II [13 L]
Communication skills, communication, definition , components- sender, message, channel , receiver, feedback, types of communication, effective interpersonal communication, barriers, communication noise, listening, ways to improve interpersonal communication, effective public speaking interview, group discussion

Unit III [13 L]
Career planning, career planning steps, choosing a career, career development, career guidance and career guidance centre, need and importance of career guidance, career guidance centre and sources, making a career decision, preparing a resume and tips

Unit IV [12 L]
Self management, self esteem, definitions, practice self acceptance, practice self acceptance characteristics of people with high self-esteem, low self esteem, characteristics and causes, self-esteem building, self awareness importance, develop self awareness, self control, developing self

control, emotional intelligence or emotional quotient, emotional quotient , two aspects of emotional intelligence, five domains of emotional eq or ei, social intelligence, coping with emotions, emotional intelligence,

Unit V [13 L]
Stress and strain: concept of stress, meaning and definition of stress, types of stress, major symptoms of stress, manage everyday stress. strain-mental strain, causes of strain, conflict, conflict resolution, understanding conflict in relationships, emotional awareness, managing and resolving conflict, stages of healthy conflict resolution, styles of conflict resolution, styles of dealing with conflict, developing positive thinking, positive and negative self-talk, better self-talk, impacts , assertiveness, behaviour , importance of assertive behaviour.

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

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

SECOND SEMESTER B.Sc. DEGREE EXAMINATION APRIL2020
(CBCSS-UG)

Core Course – COMPUTER SCIENCE: GBCS2B02T – PROBLEM SOLVING USING C

Time: 2 Hours   Maximum: 60 Marks

Section A – Short Answer type questions

(Answer all questions, each correct answer carries a maximum of 2Marks. Ceiling 20 marks)

1. How will you define a symbolic constant?

2. Differentiate between the constants 'A' and "A"

3. What is an idenStifier? Give examples

4. What is a register variable?

5. What is the purpose of return () statement?

6. How will you read a string using scanf() function?

7. What is a pointer? How will you access a variable through its pointer?

8. What is a conditional operator? Explain with example.

9. What is a structure?

10. Differentiate between actual and formal parameters with example

11. What is meant by type casting?

12. What is meant by dynamic memory allocation?

Section B – Short Essay type questions

(Answer all questions, each correct answer carries a maximum of 5 marks. Ceiling 30 marks)

13. Write a program to find the length of a string using pointer

14. What are arrays? How are arrays defined? Illustrate with example.

15. Describe the switch statement with example

16. What is a function? What are its advantages?

17. Explain the jumping statements in C with examples

18. Which are the file handling functions?

19. Differentiate between call by value and call by reference with suitable examples

SECTION C – Essay type questions

(Answer any one question, correct answer carries l0 marks)

20. Explain the different looping statements in C with examples

21. Write a program to print prime numbers in a group of n numbers using function.