ORDINANCE AND REGULATIONS FOR

B.A. / B. Sc. (VOCATIONAL) HONOURS

IN

COMPUTER APPLICATION

UNDER

CHOICE BASED CREDIT SYSTEM (CBCS)

(To be effective from 2019 -20)

[Estd.: 1917]

PATNA UNIVERSITY

ORDIANACE FOR ADMISSION OF STUDENTS TO B.A. / B. SC. (VOCATIONAL) HONOURS IN COMPUTER APPLICATION AT PATNA UNIVERSITY.

1. The University shall offer above course as per guidelines of the University Grants Commission (UGC) / State / Central Govt. issued from time to time.
2. The medium of instructions and examinations shall be English throughout the course of studies, except for Certificate and Diploma courses where the student may be offered the medium as Hindi.
3. The minimum duration to complete the course shall be that of Three (3) years and maximum up to Six (6) year.
4. **Admission:**

   Eligibility:

   I. Candidates who have passed the intermediate Examination conducted by a Board or University established or incorporated by Law / 12 year products of the Central or State Board 10+2 School system or equivalent thereof are eligible to apply for admission.

   II. Candidates must have passed the Intermediate / +2 level examination with the following combination of subject / subjects for admission to the relevant vocational course:

   a. **B.Sc. in Computer Application:** 10+2 in science with math as one of the core subject and 55 % in aggregate in Physics, Mathematics and Chemistry, 45 % aggregate Marks in Physics, math and Chemistry for SC/ST/BCI

   b. **B. A. in Computer Application:** I.A/I.Sc./I.Com or 10+2 with minimum 55% marks and 45% marks for SC/ST/BCI

   Procedure:

   Admissionwillbe made on thebasis of marks obtainedat theaptitude testfollowedby viva-voce conducted by the college.Aptitude test will be of multiple choices consisting of 20questions each from Physics and Mathematics and 10 questions from fundamentals of computer. Aptitude testwill be of100marks and viva-voce will be of25 marks.

   Candidates sponsored by NRI / Employing Establishments shall be exempted from taking the written aptitude test, but they must appear at the viva-voce examination.

   The number of seats in different vocational courses shall be the following:

a. B.Sc. in Computer in Application	:	40
b. B.A. in Computer Application	:	40

   The reservation of seats for different course shall be as per rules the Patna University for the candidates of S.C., S.T. & B.C.
5. The fee structure for B.Sc. in Computer Application/ B. A. in Computer Application course shall be the following:-

Sl. No.	Fee Structure	BCA-I Year	BCA-II Year	BCA-III Year
1.	Admission Fee	200	200	200
2.	Registration Fee (one time)	230	--	--
3.	Tuition Fee	18000	18000	18000
4.	Library	1000	1000	1000
5.	Laboratory Development charges *	1000	1000	1000
6.	Development Charges	2000	--	--
7.	Maintenance	1000	1000	1000
8.	Common Room	400	--	--
9.	Identify card (one time)	50	--	--
10.	Extra- Curricular Activities	1000	--	--
11.	Miscellaneous Charges	1000	--	--
12.	Electricity Charges (per annum)*	1000	1500	1500
13.	Internet Fee (per annum)	600	500	500
14.	Syllabus, Handbook & College Magazine	250	--	--
15.	Environment Protection	200	--	--
16.	Social work	500	--	--
17.	NSS (per annum)	20	20	20
18.	Eklaya, Tarang (per annum)	20	30	30
19.	Soft skill (per annum)	800	--	--
20.	Caution Money (Refundable)	2000	--	--
	Total	**31270.00**	**23250.00**	**23250.00**

Each candidate shall have to pay examination fee for each semester of examination in addition to the other fees as per Patna University norms.

The fee structure may be reviewed every three years by the vice-chancellor in consultation with the principals of Colleges.

For candidates sponsored by Non-Resident Indians (NRI) / Employing Establishments the fee structure shall be as follows:

CourseFees	NRI	BUSINESS ESTABLISHMENT
1) B.A / B.Sc. (vocational) Honours in computer Applications	US $ 2000 Per annum	Rs. 1,30,000/- Per annum

6. The Teaching Scheme: CHOICE BASED CREDIT SYSTEM (CBCS):

The CBCS provides an opportunity for the students to choose courses from the prescribed courses comprising core, elective/minor or skill based courses. The courses can be evaluated following the grading system, which is considered to be better than the conventional marks system. Therefore, it is necessary to introduce uniform grading system in the entire higher education in India. This will benefit the students to move across institutions within India to begin with and across countries. The uniform grading system will also enable potential employers in assessing the performance of the candidates. In order to bring uniformity in evaluation system and computation of the Cumulative Grade Point Average (CGPA) based on student's performance in examinations, the UGC has formulated the guidelines to be followed.

Outline of Choice Based Credit System:

1. **Core Course:** A course, which should compulsorily be studied by a candidate as a core requirement is termed as a Core course.
2. **Elective Course:** Generally a course which can be chosen from a pool of courses and which may be very specific or specialized or advanced or supportive to the discipline/ subject of study or which provides an extended scope or which enables an exposure to some other discipline/subject/domain or nurtures the candidate's proficiency/skill is called an Elective Course.

2.1. **Discipline Specific Elective (DSE) Course:** Elective courses may be offered by the main discipline/subject of study is referred to as Discipline Specific Elective. The University/Institute may also offer discipline related Elective courses of interdisciplinary nature (to be offered by main discipline/subject of study).

2.2. **Project/Dissertation:** An elective course designed to acquire special/advanced knowledge, as supplement study/support study to a project work. A candidate studies such a course on his own with an advisory support by a teacher/faculty member is called dissertation/project. Project work/Dissertation is considered as a special course involving application of knowledge in solving / analyzing /exploring a real life situation / difficult problem. A Project/Dissertation work would be of 6 credits. A Project/Dissertation work may be given in lieu of a discipline specific elective paper.

2.3. **Generic Elective (GE) Course:** An elective course chosen generally from an unrelated discipline/subject, with an intention to seek exposure is called a Generic Elective.

P.S.: A core course offered in a discipline/subject may be treated as an elective by otherdiscipline/subject and vice versa and such electives may also be referred to asGeneric Elective.

3. **AbilityEnhancement Courses (AE)/ Ability Enhancement Courses Compulsory (AEC)/ Ability Enhancement Courses Elective- Skill Development Courses (SEC):** The Ability Enhancement (AE) Courses may be of two kinds: AE Compulsory Course (AEC) and AE Elective-Skill Based Course (SEC). "AEC" courses are the courses based upon the content that leads to Knowledge enhancement.
   They (i) Environmental Science,
   (ii) English/MIL Communication) are mandatory for all disciplines. AEC courses are
   Value-based and/or skill-based and are aimed at providing hands-on-training, competencies, skills, etc.

3.1. **AECompulsory Course (AEC):** Environmental Science, English Communication/MIL Communication.

3.2. **AE Elective Course- Skill Development (SEC):** These courses may be chosen from a pool of courses designed to provide value-based and/or skill-based instruction.

Details of courses under B.A./B.Sc. (Vocational) (Honors) in Computer Application

Course Type	Course	Theory/Practical/Tutorial	*Credits	
	Category		Theory+ Practical	Theory + Tutorial
1.Core Course (CC)	1.1 Core Course (CC)	CC Theory (14 Papers)	14X4= 56	14X5=70
		CC Practical / Tutorial* (14 Papers)	14X2=28	14X1=14
2. Elective Course	2.1 Discipline Specific Elective (DSE)	DSE Theory (4 Papers)	4X4=16	4X5=20
		DSE Practical / Tutorial* (4 Papers)(Optional Dissertation or project work in place of one Discipline Specific Elective paper (6 credits) in 6th Semester)	4 X 2=8	4X1=4
	2.2 Generic Elective/ Interdisciplinary (GE)	GE Theory (4 Papers)	4X4=16	4X5=20
		GE Practical/Tutorial* (4 Papers)	4 X 2=8	4X1=4
3. Ability Enhancement Courses (AECC)	3.1 Ability Enhancement Compulsory (AEC)	AEC (2 Papers)	2 X 4=8	2X4=8
	3.2 Ability Enhancement Elective- Skill Based (SEC)	SEC (Minimum 2 Papers)	2 X 4=8	2X4=8
			148	**148**

- Institute should evolve a system/policy about ECA/ General Interest/Hobby/Sports/NCC/NSS/ related courses on its own.
- *Wherever there is a practical there will be no tutorial and vice-versa

PROPOSED SCHEME FOR CHOICE BASED CREDIT SYSTEM IN B.A./B.Sc (Vocational)Honours in Computer Application

Semester	Core Course (CC) (14 Pairs: Theory and Lab/Tutorial)	Discipline Specific Elective (DSE) (4 Pairs)	Generic Elective/ Interdisciplinary (GE) (4 Pairs)	Ability Enhancement Elective Skill Based (SEC) (1 Pair)	Ability Enhancement Compulsory (AEC) (1 Pair)	Credits
I	Programming in C		GE-1		English /MIL Communication	20
	Digital Electronics					
II	Programming in JAVA		GE-2		Environmental Science	20
	Computer System Architecture					
III	Data Structures		GE-3	SEC – 1		30
	Operating Systems					
	Discrete Structures					
IV	Computer Networks		GE-4	SEC - 2		30
	Software Engineering					
	Database Management Systems					
V	Internet Technologies	DSE-1				24
	Theory of Computation	DSE-2				
VI	Mobile Computing	DSE-3				24
	Design and Analysis of Algorithms	DSE-4				

Total = 148 Credits

Table 1. Course Structure forB.A./B.Sc (Vocational)Honours in Computer ApplicationUnder CBCS Session(2019 onwards)

Semester	Course/ Paper Code	Nature of course/Paper	Credit	Theory	Practical	Tutorial	Marks	Marks of CIA	Mark s of ESE	Passing Criterion	Qualifyingng Criterion
SEMESTER I	CC-1	Programming in C and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	CC-2	Digital Electronics	6	5	-	1	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	GE-1	Refer Table 2 for the list	6	5	-	1	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	AECC-1	English/MIL Communication	4	3	-	1	100	30	70	45% in CIA 45% in ESE	Qualifyingng
SEMESTER II	CC-3	Programming in JAVA and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	CC-4	Computer System Architecture and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	GE-2	select another paper from Table 2	6	5	-	1	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	AECC-2	Environmental Science	4	3	-	1	100	30	70	45% in CIA 45% in ESE	Qualifyingng
SEMESTER III	CC-5	Data Structures and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	CC-6	Operating Systems and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	CC-7	Discrete Structures	6	5	-	1	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	GE-3	select another paper from Table 2	6	5	-	1	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	SEC-1	Select from Table 2	4	3	-	1	100	30	70	45% in CIA 45% in ESE	Qualifyingng
SEMESTER IV	CC-8	Computer Networks	6	5	-	1	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	CC-9	Software engineering and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	CC-10	Database management systems and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	GE-4	select another paper from Table 2	6	5	-	1	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	SEC-2	select another paper from Table 2	4	3	-	1	100	30	70	45% in CIA 45% in ESE	Qualifyingng
SEMESTER V	CC-11	Internet Technologies and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	CC-12	Theory of Computation	6	5	-	1	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	DSE-1	Select Paper and Lab from Table 2	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	DSE-2	Dissertation/ OJT/ Project/ Internship	6	6	-	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
SEMESTER VI	CC-13	Mobile Computing and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	CC-14	Design and Analysis of Algorithms and Lab	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	DSE-3	Select Paper and Lab from Table 2	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA
	DSE-4	Select Paper and Lab from Table 2	6	4	2	-	100	30	70	45% in CIA 45% in ESE	Marks decide Class CGPA

Table 2: Proposed elective paper B.A./B.Sc (Vocational)Honours in Computer ApplicationUnder CBCS Session(2019 onwards)

Semester	Paper code	Paper Title	Credits
I	GE-1	Step1: Choose only ONE distinct discipline of interest from the following in consultation with the concerned college authority and faculty members for different semesters: 1. Mathematics 2. Statistics 3. Economics Step 2: Then, for GE -1, select one paper from the chosen discipline and follow the respective syllabus.	6 (5 Theory +1 Tutorial)
II	GE-2	For GE-2, select another paper from the same discipline given in the list of Step 1 of GE-1 which is not chosen for GE-1	6 (5 Theory +1 Tutorial)
III	GE-3	For GE-3, select another paper from a separate given in the list of Step 1 of GE -1 not the respective for GE-1 and GE-2	6 (5 Theory +1 Tutorial)
	SEC-1	Any ONE from the following list: • E1: Arduino Sensors • E2: Website Design with HTML and PHP • E3: Python Programming • Elective course available through SWAYAM or NPTEL portal (after approval of Board of Studies)	4 (3 Theory +1 Tutorial)
IV	GE-4	For GE-4, select another paper from the same discipline given in the list of Step 1 of GE-1 not chosen previously for GE-2, and GE-3	6 (5 Theory +1 Tutorial)
	SEC -2	Any ONE from the following list: • E1: Android Programming • E2: Programming in MAT Lab • E3: VB.NET Programming • Elective course available through SWAYAM or NPTEL portal (after approval of Board of Studies)	4 (3 Theory +1 Tutorial)
V	DSE -1	For DSE-1, choose any ONE from the following: • E1: Microprocessor • E2: Information Security • E3: Modelling and Simulation	6 (4 Theory +2 Practical)
VI	DSE-3	For DSE-3, choose any ONE from the following list:	6 (4 Theory +2 Practical)

		• E1: Operational Research • E2: Combinatorial Optimization • E3: Numerical Methods	
	DSE-4	For DSE 4, choose any one from the list • E1: Design Image Processing • E2: Introduction of Data Sciences • E2:Data Mining	6 (4 Theory +2 Practical)

Abbreviations	**T**: Theory, **L:** Laboratory**, CC:** Core Course, **DSE:** Discipline Specific Elective, **GE:** Generic Elective, **TL:** Theory and Lab, **AEC:** Ability Enhancement Compulsory, **SEC:** Skill Enhancement Compulsory, **PJT:** Project **CE:** Continuous Evaluation, **AT:** Attendance
Note	Each Lecture/Lab to be of 1 hour duration

REGULATIONS

1. There shall be an examination at the end of the each semester of course conducted by the college to be known respectively as B. A./B.Sc. (Vocational) Honoursin Computer Application end semester examinations

1.1.1 A student will be eligible for promotion from Semester-I to II, Semester-III to IV if he/she has:

- Secured pass marks in Internal Assessment of all the subjects/papers of the Semester I/III as the case may be.
- Has appeared in atleast one of the papers of Semester End examination of Semester I/III as the case may be.

1.1.2. A student will be eligible for promotion from Semester-II to III and Semester- IV to V provided
he/she has earned atleast 50% of the total credits for the subects/papers of theory/practicals of Semester-I/III as the case may be and passed in Internal Assessment of all subjects/papers of Semester–II/IV as the case may be.

1.1.3 A student will be eligible for promotion from Semester-V to VI provided he/she has passed in all subjects/papers of Semester-I and Semester-IIII. Provided that a student who does not fulfill the promotion criteria (1.1.1), (1.1.2) & (1.1.3) above shall be declared fail in the semester concerned. However, he/she shall have the option to retain the marks in the papers in which he/she has secured Pass marks.

1.2.A student who has to reappear in a subject / paper prescribed for Semester-I to IV shall appear in the subsequent Semester end examination to be held as per the dates prescribed by the University.

1.3 In order to be declared successful at any of these examinations the candidate shall have to pass both in Theory and Internal. Assessment/ Practical/ Project Work separately.

1.4 The College shall hold all examinations related to the vocational course and the result is to be published in accordance with the provisions of section 29(2) of the Patna University Act, 1976.

1.5 A student shall be awarded the (Vocational) Honours Degree after completion of the course according to the following examination structure :

(a) Name of the Examination - B. A. (Vocational) Honours
(b) Duration of Course in year – Three
(c) Minimum Percentage of Marks required for passing in Honours

I Class – 60 percent and above in aggregate of Honours
II Class – 45 percent and above but below 60% in aggregate of Honours.

Provided that the candidate in order to pass must have obtained 45 percent marks each in Theory and Internal Assessment / Practical/ Project Work separately.

1.6 A candidate obtaining 75 percent marks or above in Honours papers shall be declared to have passed the B. A./B.Sc. (Vocational) Honours in Computer Application degree in first class with distinction in Honours.

MODE OF EXAMINATION

2.1 There shall be written examination for each of the Theory papers and shall be of three hours duration.
2.2 Internal Assessment / Practical/ Project Work examination shall be of four hours duration.

AWARDED OF DEGREE

Each successful candidate at the B. A./B.Sc. (Vocational) Honours in Computer Application examination hall receive, in token of her degree, a diploma in prescribed form pacifying the name of the college and class in which she was placed.

OTHER

Other provisions not covered under present regulation shall be governed by the Regulation for B.A./B.Sc. Honours(Three years degree course) of PatnaUniversity

SCHEME OF EXAMINATION / ASSESSMENT UNDER CBCS

The evaluation of each course shall contain two parts :Internal or In Semester Assessment (IA) and External or End-Semester Assessment (EA).The internal grade awarded to the students in each course in a semester shall be published on the notice board at least one week before the commencement of end semester examination. The responsibility of evaluating the internal assessment is vested on the teacher(s) who teaches the course. There will be University Examinations at the end of each semester for both Theory and Practical. The evaluation of a candidate shall be awarded and record thereof maintained in accordance with the Regulations prescribed for the purpose under the CBCS as per the following:

THEORY	**Syllabus to be covered in the examination**	**Time allotted**	**% Weightage (Marks)**
Internal Assessment Test (Pattern:One Long	Upto 50% (after 45 days)	1 Hour	20

answer type question of 10 marks and Five Short answer type questions of 2 marks each)			
External End Semester University Exam (Pattern: approved by Academic Council)	Upto 100% (after 90 days)	**3 Hours**	**80**
Total			**100**
Daily Evaluation of Practical records/Viva voce/attendance etc.			50 (including 20% for attendance, 20% for Viva-voce and 60% for internal test and day to day performance)
Final Practical Performance + Viva voce (External Examination)	100% Syllabus		50 [40(Paper) +10(Viva-voce)]
Total			**100**

In case of failure/re-appear category the Internal Assessment earned by the candidate as a regular student shall be carried forward to the subsequent examination.

Examination Scheme

Scheme	CIA (Marks)	ESE (Marks)	Credit
Theory + Practical	Theory(20) + Practical (10) = 30	Theory(50) + Practical (20) = 70	Theory(4) + Practical (2) = 6
Theory + Tutorial	30	70	6

Question Paper Scheme

Continuous Internal Assessment (CIA) :

There will be **three**group of questions. **Group A is compulsory** and will contain **five** questions of **multiple choice type** consisting of **1 mark** each. **Group B will contain short answer type** consisting of **one** question of **5 marks** and **Group C will contain LONG answer type** consisting of **one** question of **10 marks**.

End Semester Examination (ESE):

There will be **three**group of questions. **Group A is compulsory** and will contain **ten** questions of **multiple choice type** consisting of 1 mark each. **Group B will contain short answer type** consisting of two questions of 5 marks each and **Group C will contain LONG answer type** consisting of **three** question of **10 marks**.

Note: There may be subdivisions in each question asked in Theory Examinations.

MINIMUM ELIGIBILITY TO APPEAR IN EXAMINATION

A semester end external examination for B. A./B.Sc. (Vocational) Honours in Computer Applicationcourse, under CBCS, shall be open to the following categories of students :

1. A Regular student i.e. a student who has undergone a regular course of study in a college for the period specified for that course of study by having been on the rolls of the college immediately preceding the examination and has his/her name submitted to the Controller of Examinations by the college Principal where he/she has pursued the course for the examination and has fulfilled the following conditions to be certified by the college Principal concerned:

   i. He/she has been a student of good conduct.

   ii. He/she has attended not less than 75% of the lecture delivered including seminars, tutorials etc. in each course opted by him/her in that semester.

   iii. He/she has passed in internal assessment in the case of laboratory course/practicals, he/she has attended not less than 75% of the practical classes conducted (practicals include field studies, workshop practice, map work, surveying etc.).

   iv. He/she has paid the prescribed fee.

   v. Ex-students i.e. students who after having undergone a regular course of study and having completed all conditions of eligibility for appearance in a course(s) in a semester examination including minimum attendance requirement and having passed in Internal Assessment and having either failed to pass the semester examination in that subject/paper(s) or been unable to appear in the examination in that subject/paper(s) will be eligible to appear as a private candidate in the examination by submitting his/her 15 application on the prescribed form along with prescribed fees to reach the Controller of Examinations within the dates fixed for this purpose.

SCRUTINY OF ANSWER SCRIPTS

i. Scrutiny of answer scripts of semester end external examination shall be permissible in each course of a Semester examination of B. A./B.Sc. (Vocational) Honours in Computer Application as per the Patna University statutes.

ii. Scrutiny shall not be permitted in the practical, internal assessment, project report, dissertation, thesis & viva-voce etc.

1st Year: Semester I

CC-1: Programming in C **[Credits: 4, Lectures: 60]**

Unit 1: Introduction to C (6 Lectures)
History of C, Overview of Procedural Programming, Introduction to Algorithm & Flowcharts.

Unit 2: Understanding Compilation and Execution in C (6 Lectures)
Using main() function Compiling and Executing Simple Programs in C.

Unit 3: Data Types, Variables, Constants, Operators and Basic I/O (7 Lectures)
Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putcharetc), Formatted and Console I/O (printf(), scanf(), cin, cout), Using Basic Header Files (stdio.h, iostream.h, conio.hetc).

Unit 4: Expressions, Conditional Statements and Iterative Statements (7 Lectures)
Simple Expressions in C(including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

Unit 5: Understanding Functions (6 Lectures)
Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments.

Unit 6: Implementation of Arrays and Strings (6 Lectures)
Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two-dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays

Unit 7: User-defined Data Types (Structures and Unions) (4 Lectures)
Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.

Unit 8: Pointers and References in C (8 Lectures)
Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with Pointers, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions. Pointers vs. References, Declaring and initializing references, Using references as function arguments and function return values

Unit 9: Memory Allocation in C (3 Lectures)

Differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions, use of new and delete operators, storage of variables in static and dynamic memory allocation

Unit 10: File I/O (4 Lectures)
Opening and closing a file, Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files,

Unit 11: Preprocessor Directives (3 Lectures)
Understanding the Preprocessor Directives (#include, #define, #error, #if, #else, #elif, #endif, #ifdef, #ifndef and #undef), Macros

Suggested Readings

1."The C Programming Language ANSI C Version", Kernighan & Ritchie, Prentice Hall SoftwareSeries
2. "ANSI C - Made Easy", Herbert Schildt, Osborne McGraw-Hill
3. "Learning to Program in C", N. Kantaris, Babani
4. "C - The Complete Reference", Herbert Schildt, Osborne McGraw-Hill
5. "Programming in C", ReemaThareja, Oxford University Press
6. "A First Course in Programming With C", T. Jeyapoovan, Vikas Publishing House
7. "Let Us C", Yashavant P. Kanetkar, BPB Publications

Programming in C Lab **[Credit: 2, Lab Hours: 60]**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. WAP to perform input/output of all basic data types.
2. WAP to enter two numbers and find their sum.
3. WAP to reverse a number.
4. WAP to Swap Two Numbers.
5. WAP to Check Whether a Number is Even or Odd
6. WAP to compute the factors of a given number.
7. WAP to enter marks of five subjects and calculate total, average and percentage.
8. WAP to print the sum and product of digits of an integer.
9. WAP to Check Whether a Character is Vowel or Consonant
10. WAP to Find the Largest Number Among Three Numbers
11. WAP to compute the sum of the first n terms of the following series S =1-2+3-4+5…………….
12. WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+……
13. WAP to print a triangle of stars as follows (take number of lines from user):

```
    *          1           1                1   A                  A
   ***        11          2 3             1 2   A B                BA
  *****       121        4 5 6          1 2 3   A B C              CBA
 *******      1331      7 8 9 10      1 2 3 4   A B C D            DCBA
*********     14641  11 12 13 14 15   1 2 3 4 5 A B C D E         EDCBA
```

14. WAP to perform following actions on an array entered by the user:
    a) Print the even-valued elements
    b) Print the odd-valued elements
    c) Calculate and print the sum and average of the elements of array
    d) Print the maximum and minimum element of array
    e) Remove the duplicates from the array
    f) Print the array in reverse order

The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

15. WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
16. Write a program that swaps two numbers using pointers.
17. Write a program in which a function is passed address of two variables and then alter its contents.
18. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.
19. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc() functions or new operator.
20. Write a menu driven program to perform following operations on strings:
21. Show address of each character in string
22. Concatenate two strings without using strcat function.
23. Concatenate two strings using strcat function.
24. Compare two strings
25. Calculate length of the string (use pointers)
26. Convert all lowercase characters to uppercase
27. Convert all uppercase characters to lowercase
28. Calculate number of vowels
29. Reverse the string
30. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
31. WAP to display Fibonacci series (i)using recursion, (ii) using iteration
32. WAP to calculate Factorial of a number (i)using recursion, (ii) using iteration
33. WAP to calculate GCD of two numbers (i) with recursion (ii) without recursion.
34. Write a menu-driven program to perform following Matrix operations (2-D array implementation):

    a) Sum b) Difference c) Product d) Transpose

35. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.
36. Write a program to retrieve the student information from file created in previous question and print it in following format:

    Roll No. Name Marks

37. Copy the contents of one text file to another file, after removing all whitespaces.
38. WAP to Write a Sentence to a File.
39. WAP to Read a Line From a File and Display it.
40. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.
41. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

CC -2: Digital Electronics [Credits: 5+1, Lectures: 60]

Unit-1: Fundamentals of Computers (15 Lectures)

Generation of Computers and Computer Languages, Computer Systems, Basic block Diagram, Von-Neumann Architecture, Types of Computers, Hardware, Firmware, I/O Devices, Storage classifications, Language translators.

Unit-2: Logic gates (15 Lectures)

AND, OR, NOT Gates and their Truth Tables, NOR, NAND & XOR gates, Boolean algebra, Basic Boolean Laws, De-morgan‟s theorem, Boolean function and their truth tables, MAP simplification, Minimization techniques, K-Map, Sum of Product & Product of Sum, Venn diagram.

Unit-3: Circuits, Adders, Flip Flops, Registers etc. (15 Lectures)

Combinational & Sequential circuits, Half adder & Full adder, BCD adder, Full Subtractor, Flip-flops-RS, D, JK, T & Master-Slave flip-flops, Shift registers, Multiplexer, Encoder, Decoder.

Unit-4: Number Systems & Arithmetic (15 Lectures)
Number System: Positional, Binary, Octal, Decimal, Hexa-Decimal and their Representations. Methods of conversion
from one Base to another. Binary Addition, Subtraction, Multiplication. Negative number representation: Sign magnitude, 1's, 2's Complement. Different coding schemes: BCD, EBCDIC, UNICODE, ASCII, GRAY, Excess-3 Codes. Fixed and floating point representation

Suggested Readings

1. Rajaraman V. &Radhakrishnan, An Introduction To Digital Computer Design, PHI.
2. Malvino& Leach, Digital Principles & Applications, TMH
3. S. Salivahanan, S. Arivazhagan, Digital Circuits and Design, Oxford University Press

GE 14: General Elective – 1 **[Theory: 60 Lectures]**

Step1: Choose only ONE discipline of interest from the following in consultation with the concerned college authority and faculty members:

1. Mathematics
2. Statistics
3. Economics

AECC-1: English/MIL Communication **[Credit: 4, Lecture Hours: 50]**

Preamble: The purpose of this course is to introduce students to the theory, fundamentals and tools of communication and to develop in them vital communication skills which should be integral to personal, social and professional interactions. One of the critical links among human beings and an important thread that binds society together is the ability to share thoughts, emotions and ideas through various means of communication: both verbal and non-verbal. In the context of rapid globalization and increasing recognition of social and cultural pluralities, the significance of clear and effective communication has substantially enhanced.

The present course hopes to address some of these aspects through an interactive mode of teaching-learning process and by focusing on various dimensions of communication skills. Some of these are:

Language of communication, various speaking skills such as personal communication, social interactions and communication in professional situations such as interviews, group discussions and office environments, important reading skills as well as writing skills such as report writing, note taking etc.

While, to an extent, the art of communication is natural to all living beings, in today's world of complexities, it has also acquired some elements of science. It is hoped that after studying this course, students will find a difference in their personal and professional interactions.

The recommended readings given at the end are only suggestive; the students and teachers have the freedom to consult other materials on various units/topics given below. Similarly, the questions in the examination will be aimed towards assessing the skills learnt by the students rather than the textual content of the recommended books.

1. **Introduction**:
   i. Theory of Communication,
   ii. Types and modes of Communication

2. **Language of Communication**:
   i. Verbal and Non-verbal (Spoken and Written)
   ii. Personal, Social and Business
   iii. Barriers and Strategies
   iv. Intra-personal, Inter-personal and Group communication
3. **Speaking Skills**:

i. Monologue
ii. Dialogue
iii. Group Discussion
iv. Effective Communication/ Ms- Communication
v. Interview
vi. Public Speech

4. **Reading and Understanding**
   i. Close Reading
   ii. Comprehension
   iii. Summary Paraphrasing
   iv. Analysis and Interpretation
   v. Translation(from Indian language to English and vice-versa)
   vi. Literary/Knowledge Texts
5. **Writing Skills**
   i. Documenting
   ii. Report Writing
   iii. Making notes
   iv. Letter writing

Suggested Readings:

1. *Fluency in English* - Part II, Oxford University Press, 2006.
2. *Business English*, Pearson, 2008.
3. *Language, Literature and Creativity*, OrientBlackswan, 2013.
4. *Language through Literature* (forthcoming) ed. Dr. Gauri Mishra, Dr RanjanaKaul, Dr Brati Biswas

----- End of Semester I-----

1st Year: Semester II

CC-3: Programming in JAVA [Credit: 4, Lecture Hours: 60]

Unit 1: Introduction to Java (4 Lectures)
Java Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods),

Unit 2: Arrays, Strings and I/O (8 Lectures)
Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System.out and the Scanner class, Byte and Character streams, Reading/Writing from console and files.

Unit 3: Object-Oriented Programming Overview (4 Lectures)
Principles of Object-Oriented Programming, Defining & Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

Unit 4: Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata (14 lectures)
Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

Unit 5: Exception Handling, Threading, Networking and Database Connectivity (15 Lectures)
Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

Unit 6: Applets and Event Handling (15 Lectures)
Java Applets: Introduction to Applets, Writing Java Applets, Working with Graphics,
Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes. The design and Implementation of GUIs using the AWT controls, Swing components of Java Foundation Classes such as labels, buttons, textfields, layout managers, menus, events and listeners; Graphic objects for drawing figures such as lines, rectangles, ovals, using different fonts. Overview of servlets.

Suggested Readings

1. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th Edition, 2005.
2. James Gosling, Bill Joy, Guy L Steele Jr, GiladBracha, Alex Buckley"The Java Language Specification, Java SE 8 Edition (Java Series)", Published by Addison Wesley, 2014.
3. Joshua Bloch, "Effective Java" 2nd Edition,Publisher: Addison-Wesley, 2008.
4. Cay S. Horstmann, GaryCornell, "Core Java 2 Volume 1 ,9th Edition,Printice Hall.2012
5. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 2 - Advanced Features)", 9th Edition, Printice Hall.2013
6. Bruce Eckel, "Thinking in Java", 3rd Edition, PHI, 2002.
7. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill.2009.
8. Paul Deitel, Harvey Deitel, "Java: How to Program", 10th Edition, Prentice Hall, 2011.
9. Bert Bates, Kathy Sierra, "Head First Java", Orielly Media Inc. 2nd Edition, 2005.
10. Object Oriented Programming through JAVA, P Radha Krishna, University Press
11. David J. Eck, "Introduction to Programming Using Java", Published by CreateSpace Independent Publishing Platform, 2009.
12. John R. Hubbard, "Programming with JAVA", Schaum's Series, 2nd Edition, 2004.
13. R. Nageswara, Core Java: An Integrated Approach, Dreamtech Press

Programming in JAVA Lab[Credit: 2, Lab Hours: 60]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. To find the sum of any number of integers entered as command line arguments
2. To find the factorial of a given number
3. To learn use of single dimensional array by defining the array dynamically.
4. To learn use of .lenth in case of a two dimensional array
5. To convert a decimal to binary number
6. To check if a number is prime or not, by taking the number as input from the keyboard
7. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument
8. Write a program that show working of different functions of String and StringBufferclasss like setCharAt(, setLength(), append(), insert(), concat()and equals().
9. Write a program to create a ―distance‖ class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer
10. Modify the ―distance‖ class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
11. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions(from lower to higher data type)
12. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword
13. Write a program to show the use of static functions and to pass variable length arguments in a function.

14. Write a program to demonstrate the concept of boxing and unboxing.
15. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
16. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate fibonacii series is given in a different file belonging to the same package.
17. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
18. Write a program ―DivideByZero that takes two numbers a and b as input, computes a/b, and invokesArithmetic Exception to generate a message when the denominator is zero.
19. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
20. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Throwable).
21. Write a program to demonstrate priorities among multiple threads.
22. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).
23. Write a program to create URL object, create a URLConnection using the openConnection() method and then use it examine the different components of the URLand content.
24. Write a program to implement a simple datagram client and server in which a message that is typed into the server window is sent to the client side where it is displayed.
25. Write a program that creates a Banner and then creates a thread to scrolls the message in the banner from left to right across the applet„s window.
26. Write a program to get the URL/location of code (i.e. java code) and document(i.e. html file).
27. Write a program to demonstrate different mouse handling events like :- mouseClicked(),mouseEntered(), mouseExited(), mousePressed, mouseReleased() and mouseDragged().
28. Write a program to demonstrate different keyboard handling events.
29. Write a program to generate a window without an applet window using main() function.
30. Write a program to demonstrate the use of push buttons.

CC-4: Computer System Architecture [Credits: 4, Lectures: 60]

Unit 1: Introduction (8 lectures)
Logic gates, boolean algebra, combinational circuits, circuit simplification, flip-flops and sequential circuits, decoders, multiplexers, registers, counters and memory units.

Unit 2: Data Representation and Basic Computer Arithmetic (10 lectures)
Number systems, complements, fixed and floating point representation, character representation, addition, subtraction, magnitude comparison, multiplication and division algorithms for integers

Unit 3: Basic Computer Organization and Design (13 lectures)
Computer registers, bus system, instruction set, timing and control, instruction cycle, memory reference, input-output and interrupt, Interconnection Structures, Bus Interconnection design of basic computer.

Unit 4: Central Processing Unit (15 lectures)
Register organization, arithmetic and logical micro-operations, stack organization, micro programmed control. Instruction formats, addressing modes, instruction codes, machine language, assembly language, input output programming, RISC, CISC architectures, pipelining and parallel architecture.

Unit 5: Memory Organization (6 lectures)
Cache memory, Associative memory, mapping.

Unit 6: Input-Output Organization (8 lectures)

Input / Output: External Devices, I/O Modules, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access, I/O Channels.

Suggested Readings

1. M. Mano, Computer System Architecture, Pearson Education 1992
2. A. J. Dos Reis, Assembly Language and Computer Architecture using C++ and JAVA, Course Technology, 2004
3. W. Stallings, Computer Organization and Architecture Designing for Performance, 8th Edition, Prentice Hall of India,2009
4. M.M. Mano , Digital Design, Pearson Education Asia,2013
5. Carl Hamacher, Computer Organization, Fifth edition, McGrawHill, 2012.

Computer System Architecture Lab[Credit: 2, Lab Hours: 60]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. General study of Basic & Universal gates
2. Simple Boolean Expression using Basic gates and Universalgates: A . (B+A) + B.A
3. Design Half-Adder, Full-Adder, Half-Subtractor, Full-Subtractor Circuit.
4. Parallel Adder (2-bit, 3-bit) Circuit.
5. Implement logic functions in SOP form using Multiplexer.
6. Implement De-multiplexer.
7. Implement 7- Segment Display with Decoder.
8. Implement Parity Generator (Odd & Even)
9. Implement Magnitude Comparator (1-bit, 2-bit, 3-bit)
10. Circuit Design and implementation of Flip-Flops (SR, JK, D) using ICs
11. Circuit design and implementation of Decoder (2x4) and Encoder (4x2) using ICs

GE-2:Generic Elective-2 [Theory: 60 Lectures]

For GE -2, select another paper from separate discipline given in the list of Step 1 of GE -1 which is not chosen for GE -1 and follow the respective syllabus.

AEC-2: Environmental Science [Credits: 4, Lectures: 50]

Unit 1: Introduction to environmental studies (2 lectures)

- Multidisciplinary nature of environmental studies;
- Scope and importance; Concept of sustainability and sustainable development.

Unit 2: Ecosystems (6 Lectures)

i. What is an ecosystem? Structure and function of ecosystem; Energy flow in an ecosystem: food chains, foodwebs and ecological succession. Case studies of the following ecosystems :

a) Forest ecosystem
b) Grassland ecosystem
c) Desert ecosystem
d) Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

ii. Land resources and land use change; Land degradation, soil erosion and desertification.Deforestation: Causes and impacts due to mining, dam building on environment, forests, biodiversity andtribal populations.

iii. Water: Use and over--exploitation of surface and ground water, floods, droughts, conflicts over water (international & inter--state).

iv. Energy resources: Renewable and non-renewable energy sources, use of alternate energy sources, growing energy needs, case studies.

Unit 4: Biodiversity and Conservation (8 Lectures)

i. Levels of biological diversity: genetic, species and ecosystem diversity; Biogeographic zones of India; Biodiversity patterns and global biodiversity hot spots.

ii. India as a mega--biodiversity nation; Endangered and endemic species of India.

iii. Threats to biodiversity: Habitat loss, poaching of wildlife, man--wildlife conflicts, biological invasions;Conservation of biodiversity: In--situ and Ex--situ conservation of biodiversity.

iv. Ecosystem and biodiversity services: Ecological, economic, social, ethical, aesthetic and Informational value.

Unit 5: Environmental Pollution (8 Lectures)

i. Environmental pollution : types, causes, effects and controls; Air, water, soil and noise pollution

ii. Nuclear hazards and human health risks

iii. Solid waste management: Control measures of urban and industrial waste.

iv. Pollution case studies.

Unit 6: Environmental Policies & Practices (7 Lectures)

i. Climate change, global warming, ozone layer depletion, acid rain and impacts on human communities and agriculture

ii. Environment Laws: Environment Protection Act; Air (Prevention & Control of Pollution) Act; Water (Prevention and control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act. Internationalagreements: Montreal and Kyoto protocols and Convention on Biological Diversity (CBD).Nature reserves, tribal populations and rights, and human wildlife conflicts in Indian context.

Unit 7: Human Communities and the Environment (6 Lectures)

i. Human population growth: Impacts on environment, human health and welfare.

ii. Resettlement and rehabilitation of project affected persons; case studies.

iii. Disaster management: floods, earthquake, cyclones and landslides.

iv. Environmental movements: Chipko, Silent valley, Bishnois of Rajasthan.

v. Environmental ethics: Role of Indian and other religions and cultures in environmental conservation.

vi. Environmental communication and public awareness, case studies (e.g., CNG vehicles in Delhi).

Unit 8: Field work (Equivalent to 5 Lectures)

i. Visit to an area to document environmental assets: river/ forest/ flora/fauna, etc.

ii. Visit to a local polluted site--Urban/Rural/Industrial/Agricultural.

iii. Study of common plants, insects, birds and basic principles of identification.

iv. Study of simple ecosystems--pond, river, Delhi Ridge, etc.

Suggested Readings:

1. Carson, R. 2002. Silent Spring. Houghton Mifflin Harcourt.
2. Gadgil, M., &Guha, R. 1993. This Fissured Land: An Ecological History of India. Univ. of California Press.
3. Gleeson, B. and Low, N. (eds.) 1999. Global Ethics and Environment, London, Routledge.
4. Gleick, P. H. 1993. Water in Crisis. Pacific Institute for Studies in Dev., Environment & Security. Stockholm Env. Institute, Oxford Univ. Press.
5. Groom, Martha J., Gary K. Meffe, and Carl Ronald Carroll. Principles of Conservation Biology. Sunderland:Sinauer Associates, 2006.
6. Grumbine, R. Edward, and Pandit, M.K. 2013. Threats from India‟s Himalaya dams. Science, 339:
   36--37.

7. McCully, P. 1996. Rivers no more: the environmental effects of dams (pp. 29--64). Zed Books.
8. McNeill, John R. 2000. Something New Under the Sun: An Environmental History of the Twentieth Century.
9. Odum, E.P., Odum, H.T. & Andrews, J. 1971. Fundamentals of Ecology. Philadelphia: Saunders.
10. Pepper, I.L., Gerba, C.P. &Brusseau, M.L. 2011. Environmental and Pollution Science. Academic Press.
11. Rao, M.N. &Datta, A.K. 1987. Waste Water Treatment. Oxford and IBH Publishing Co. Pvt. Ltd.
12. Raven, P.H., Hassenzahl, D.M. & Berg, L.R. 2012. Environment. 8th edition. John Wiley & Sons.
13. Rosencranz, A., Divan, S., & Noble, M. L. 2001. Environmental law and policy in India. Tripathi 1992.
14. Sengupta, R. 2003. Ecology and economics: An approach to sustainable development. OUP.
15. Singh, J.S., Singh, S.P. and Gupta, S.R. 2014. Ecology, Environmental Science and Conservation. S. Chand Publishing, New Delhi.
16. Sodhi, N.S., Gibson, L. & Raven, P.H. (eds). 2013. Conservation Biology: Voices from the Tropics. John Wiley & Sons.
17. Thapar, V. 1998. Land of the Tiger: A Natural History of the Indian Subcontinent.
18. Warren, C. E. 1971. Biology and Water Pollution Control. WB Saunders.
19. Wilson, E. O. 2006. The Creation: An appeal to save life on earth. New York: Norton.
20. World Commission on Environment and Development. 1987. Our Common Future. Oxford University Press.

^

----- End of Semester II-----

2nd Year: Semester III

CC -5 : Data Structures **[Credit: 4, Lecture Hours: 60]**

Unit 1: Arrays (5 Lectures)
Single and Multi-dimensional Arrays, Sparse Matrices (Array and Linked Representation)

Unit 2: Stacks (5 Lectures)
Implementing single / multiple stack/s in an Array; Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another; Applications of stack; Limitations of Array representation of stack

Unit 3: Linked Lists (10 Lectures)
Singly, Doubly and Circular Lists (Array and Linked representation); Normal and Circular representation of Stack in Lists; Self Organizing Lists; Skip Lists

Unit 4: Queues (5 Lectures)
Array and Linked representation of Queue, De-queue, Priority Queues

Unit 5: Recursion (5 lectures)
Developing Recursive Definition of Simple Problems and their implementation; Advantages and Limitations of Recursion; Understanding what goes behind Recursion (Internal Stack Implementation)

Unit 6: Trees (20 Lectures)

Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion , Recursive and Iterative Traversals on Binary Search Trees); Threaded Binary Trees (Insertion, Deletion, Traversals); Height-Balanced Trees (Various operations on AVL Trees).

Unit 7: Searching and Sorting (5 Lectures)

Linear Search, Binary Search, Comparison of Linear and Binary Search, Selection Sort, Insertion Sort, Insertion Sort, Shell Sort, Comparison of Sorting Techniques

Unit 8: Hashing (5 Lectures)

Introduction to Hashing, Deleting from Hash Table, Efficiency of Rehash Methods, Hash Table Reordering, Resolving collusion by Open Addressing, Coalesced Hashing, Separate Chaining, Dynamic and Extendible Hashing, Choosing a Hash Function, Perfect Hashing Function

Suggested Readings

1. Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.
2. SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.
3. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using C and C++:, Second edition, PHI, 2009.
4. Robert L. Kruse, "Data Structures and Program Design in C++", Pearson, 1999.
5. D.S Malik, Data Structure using C++,Second edition, Cengage Learning, 2010
6. Mark Allen Weiss, "Data Structures and Algorithms Analysis in Java", Pearson Education, 3rd edition, 2011.
7. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using Java, 2003.
8. Robert Lafore, "Data Structures and Algorithms in Java, 2/E", Pearson/ Macmillan Computer Pub,2003.
9. John Hubbard, "Data Structures with JAVA", McGraw Hill Education (India) Private Limited; 2 edition, 2009.
10. Goodrich, M. and Tamassia, R. "Data Structures and Algorithms Analysis in Java", 4th Edition, Wiley,2013
11. Herbert Schildt, "Java The Complete Reference (English) 9th Edition Paperback", Tata McGraw Hill, 2014.
12. D. S. Malik, P.S. Nair, "Data Structures Using Java", Course Technology, 2003.

Data Structures Lab[Credit: 2, Lab Hours: 60]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.
2. WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.
3. Implement Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).
4. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
5. Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
6. Perform Stack operations using Linked List implementation.
7. Perform Stack operations using Array implementation. Use Templates.
8. Perform Queues operations using Circular Array implementation. Use Templates.

9. Create and perform different operations on Double-ended Queues using Linked List implementation.
10. WAP to scan a polynomial using linked list and add two polynomial.
11. WAP to calculate factorial and to compute the factors of a given no. (i)using recursion, (ii) using iteration
12. WAP to display fibonacci series (i)using recursion, (ii) using iteration
13. WAP to calculate GCD of 2 number (i) with recursion (ii) without recursion
14. WAP to create a Binary Search Tree and include following operations in tree:
    - (i) Insertion (Recursive and Iterative Implementation)
    - (ii) Deletion by copying
    - (iii) Deletion by Merging
    - (iv) Search a no. in BST
    - (v) Display its preorder, postorder and inorder traversals Recursively
    - (vi) Display its preorder, postorder and inorder traversals Iteratively
    - (vii) Display its level-by-level traversals
    - (viii) Count the non-leaf nodes and leaf nodes
    - (ix) Display height of tree
    - (x) Create a mirror image of tree
    - (xi) Check whether two BSTs are equal or not
15. WAP to convert the Sparse Matrix into non-zero form and vice-versa.
16. WAP to reverse the order of the elements in the stack using additional stack.
17. WAP to reverse the order of the elements in the stack using additional Queue.
18. WAP to implement Diagonal Matrix using one-dimensional array.
19. WAP to implement Lower Triangular Matrix using one-dimensional array.
20. WAP to implement Upper Triangular Matrix using one-dimensional array.
21. WAP to implement Symmetric Matrix using one-dimensional array.
22. WAP to create a Threaded Binary Tree as per inorder traversal, and implement operations like finding the successor / predecessor of an element, insert an element, inorder traversal.
23. WAP to implement various operations on AVL Tree.

CC -6: Operating Systems **[Credit: 4, Lecture Hours: 60]**

Unit 1: Introduction (10 Lectures)
Basic OS functions, resource abstraction, types of operating systems–multiprogramming systems, batch systems , time sharing systems; operating systems for personal computers & workstations, process control & real time systems.

Unit 2: Operating System Organization (6 Lectures)
Processor and user modes, kernels, system calls and system programs.

Unit 3: Process Management (20 Lectures)
System view of the process and resources, process abstraction, process hierarchy, threads, threading issues, thread libraries; Process Scheduling, non-preemptive and preemptive scheduling algorithms; concurrent and processes, critical section, semaphores, methods for inter-process communication; deadlocks.

Unit 4: Memory Management (10 Lectures)
Physical and virtual address space; memory allocation strategies -fixed and variable partitions, paging, segmentation, virtual memory

Unit 5: File and I/O Management (10 Lectures)
Directory structure, file operations, file allocation methods, device management.

Unit 6: Protection and Security (4 Lectures)
Policy mechanism, Authentication, Internal access Authorization.

Suggested Readings

1. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John Wiley Publications 2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
3. G. Nutt, Operating Systems: A Modern Perspective, 2nd Edition Pearson Education 1997.
4. W. Stallings, Operating Systems, Internals & Design Principles, 5th Edition, Prentice Hall of India. 2008.
5. M. Milenkovic, Operating Systems- Concepts and design, Tata McGraw Hill 1992.
6. Operating Systems, A K Sharma, University Press

Operating Systems Lab[Credit: 2, Lab Hours: 60]

Students are advised to do laboratory/practical practice not limited to, but including the following types of

Problems: **[Implement the followings using Bash Shell Script]**

1. Evaluate : 1 * 2^3 + 2 * 4^3 + 3 * 6^3 +.... up to n terms
2. Evaluate : 1^2 + 3^2 + 6^2 + 10^2.... up to n terms
3. Evaluate : 1 * 3^2 + 2 * 5^2 + 3 * 7^2 +.... up to n terms
4. Evaluate : 1 - 2 + 3 - 4 +.... up to n terms
5. Print the following:

```
1                  *                *****                      54321
12                 **               ****                       4321
123                ***              ***                        321
1234               ****             **                         21
12345              *****             *                         1
```

6. Find the factorial of a number.
7. Find HCF an LCM of 2/3 numbers.
8. Check whether a given year is leap year or not.
9. Find the sum of squares of the digits of a numbers.
10. Find the Armstrong numbers between 100 & 999.
11. Check whether a given number is prime or not.
12. Print all the prime numbers between 1 & 100.
13. Print the multiplication table of a number.
14. Check whether a number is perfect square or not.
15. Find the first and last digit of any number and find their product.
16. Find the Area of Circle, Rectangle, or Triangle using user-input.
17. Find the prime factors of a number.
18. Convert a decimal number to binary.
19. Convert a binary number to decimal.
20. Check whether a given string is palindrome or not.
21. Abbreviate a name.
22. Reverse a string.
23. Find the largest and smallest numbers from a list of 10 integers.
24. Find the numbers and their sums between 100 to 300 which are divisible by 3 and 7.
25. Calculate nCr using function.

CC -7: Discrete Structures [Credit: 5, Lecture Hours: 60]

Unit 1: Introduction: (15 Lectures)

Sets - finite and Infinite sets, uncountably Infinite Sets; functions, relations, Properties of Binary Relations, Closure, Partial Ordering Relations; counting - Pigeonhole Principle, Permutation and Combination; Mathematical Induction, Principle of Inclusion and Exclusion.

Unit 2: Growth of Functions: (8 Lectures)
Asymptotic Notations, Summation formulas and properties, Bounding Summations, approximation by Integrals

Unit 3: Recurrences: (10 Lectures)
Recurrence Relations, generating functions, Linear Recurrence Relations with constant coefficients and their solution, Substitution Method, Recurrence Trees, Master Theorem

Unit 4: Graph Theory (15 Lectures)
Basic Terminology, Models and Types, multigraphs and weighted graphs, Graph Representaion, Graph Isomorphism, Connectivity, Euler and Hamiltonian Paths and Circuits, Planar Graphs, Graph Coloring, Trees, Basic Terminology and properties of Trees, Introduction to Spanning Trees

Unit 5: Prepositional Logic (12 Lectures)
Logical Connectives, Well-formed Formulas, Tautologies, Equivalences, Inference Theory

Suggested Readings

1. C.L. Liu , D.P. Mahopatra, Elements of Discrete mathematics, 2nd Edition , Tata McGraw Hill, 1985,
2. Kenneth Rosen, Discrete Mathematics and Its Applications, Sixth Edition ,McGraw Hill 2006
3. T.H. Coremen, C.E. Leiserson, R. L. Rivest, Introduction to algorithms, 3rd edition Prentice Hall on India, 2009
4. M. O. Albertson and J. P. Hutchinson, Discrete Mathematics with Algorithms , John wiley Publication, 1988
5. J. L. Hein, Discrete Structures, Logic, and Computability, 3rd Edition, Jones and Bartlett Publishers, 2009
6. D.J. Hunter, Essentials of Discrete Mathematics, Jones and Bartlett Publishers, 2008
7. Discrete Mathematical Structures with Applications to Combinatorics, V Ramaswamy, University Press
8. Discrete Mathematics: A Concept-based Approach, Basavaraj S Anami, Venkanna S Madalli, University Press

GE -3: Generic Elective-3 **[Theory: 60 Lectures]**

For GE -3, select another paper from separate discipline given in the list of Step 1 of GE -1 which is not chosen for GE -1 and GE -2 and follow the respective syllabus.

SEC -1 : Any one from the list (E1-E3)

E1: Arduino Sensors **[15 Lectures +30 Labs]**

Unit-1: Introduction (3 Lectures)
Arduino Basics, The Arduino platform, Block Diagram, Architecture.

Unit-2: Arduino Basic programming essentials (3 Lectures)
Arduino Control structure, Arduino Functions, Arduino operators, Arduino Sketch Structure

Unit-3: Interfacing (3 Lectures)
LED with Arduino, Working of LED, Sketch for blinking LED using delay function, Sketch Explanation,Interfacing LCD display with Arduino, Interfacing different sensors with Arduino Sensor

Unit-4: IR Sensor (3 Lectures)

Introduction to IR Sensor, Working of IR sensor, Pinouts of IR sensor, Connection of IR sensor with Arduino, Sketch showing working of IR sensor, Sketch explanation Sensor

Unit-5: Ultrasonic Sensor (HC-SR04) (3 Lectures)
Introduction to HC-SR04, Working of HC-SR04, Pinouts of HC-SR04, Connection of HC-SR04 with Arduino, Sketch for water level monitoring using HC-SR04, Sketch explanation Sensor.

Suggested Readings

1. Make: Sensors, Book by KimmoKarvinen, TeroKarvinen, and Ville Valtokari
2. Getting Started with Sensors: Measure the World with Electronics, Arduino and Raspberry Pi by KimmoKarvinenand TeroKarvinen

Arduino Lab:

1. To build a sensor circuit that senses ambient light level and uses its sensor reading to make a decision (Analogue to Digital Conversion ADC).
2. To use Pulse-Width Modulation (PWM) to fade in and out an LED (Digital to Analogue Conversion DAC).
3. To design a circuit that combines the ADC and DAC points above to create a new function.

E2: Website Design with HTML and PHP [15 Lectures+ 30Labs]

Unit 1: Introduction (1 Lectures)
Introduction to Markup Languages and HTML, need and use; the Head, the Body, Colors, Attributes, Lists, ordered and unordered

Unit 2: Links, Images and Tables (3 Lectures)
Introduction; Relative Links, Absolute Links; Link Attributes; Using the ID Attribute to Link Within a Document; Putting an Image on a Page, Using Images as Links, Putting an Image in the Background, Creating a Table, Table Headers, Captions, Spanning Multiple Columns, Styling Table

Unit 3:Introduction to XML(4 Lectures)
Introduction to XML and its Goals, XML Structure and Syntax, Document classes and Rules, Scripting XML, XML as Data, Linking with XML, XSL –Style Sheet Basics, XSL basics, XSL style sheets. (3L)

Unit 4: Introduction to PHP(3 Lectures)
PHP introduction, inventions and versions, scope, important tools and software requirements (like Web Server, Database, Editors etc.), Basic Syntax, PHP variables and constants, Types of data in PHP , Expressions, scopes of a variable (local, global), PHP Operators: Arithmetic, Assignment, Relational , Logical operators, Bitwise , ternary and MOD operator, PHP operator Precedence and associatively

Unit 5: Handling HTML form with PHP(2 Lectures)
Basic Input and Attributes, Other Kinds of Inputs, Styling forms with CSS, Where to Go from Here Capturing Form Data, GET and POST form methods, Dealing with multi value fields, Redirecting a form after submission

Unit 6: PHP conditional events and Loops(3 Lectures)
PHP IF Else conditional statements (Nested IF and Else), Switch case, while,For and Do While Loops, Goto, Break,Continue and exit

Unit 7: PHP Functions(3 Lectures)

Function, Need of Function, declaration and calling of a function, PHP Function with arguments, Default Arguments in Function,Function argument with call by value, call by reference, Scope of Function Global and Local

Unit 8: String Manipulation and Regular Expression (5 Lectures)

Creating and accessing String , Searching & Replacing String; Formatting, joining and splitting String , String Related Library functions; Use and advantage of regular expression over inbuilt function; Use of preg_match(), preg_replace(), preg_split() functions in regular expression

Unit 9: Array (3 Lectures)

Anatomy of an Array ,Creating index based and Associative array ,Accessing array; Looping with Index based array, with associative array using each() and foreach(); Some useful Library function

Suggested Readings

1. VirginiaDeBolt , Integrated HTML and CSS A Smarter, Faster Way to Learn , Wiley / Sybex , 2006
2. Cassidy Williams, Camryn Williams Introduction to HTML and CSS, O'Reilly, 2015
3. XML in action web technology by William J. Pardi
4. Step by Step XML by Michael J. Young
5. Steven Holzner, "PHP: The Complete Reference Paperback", McGraw Hill Education (India), 2007.
6. Timothy Boronczyk, Martin E. Psinas, "PHP and MYSQL (Create-Modify-Reuse)", Wiley India Private Limited, 2008.
7. Robin Nixon, "Learning PHP, MySQL, JavaScript, CSS & HTML5", 3rd Edition Paperback, O'reilly, 2014.
8. Luke Welling, Laura Thompson, PHP and MySQL Web Development", 4th Edition, Addition Paperback, Addison-Wesley Professional,2008.
9. David Sklar, Adam Trachtenberg, "PHP Cookbook: Solutions & Examples for PHP Programmers", 2014.

Software Lab Based on HTML, XML and PHP(Not limited but including the following):

1. Create an HTML document with the different formatting options i.e. Bold, Italics, Underline, Headings (Using H1 to H6 heading styles), Font (Type, Size and Color), Background (Colored background/Image in background), Paragraph, Line Break, Horizontal Rule, Pre tag
2. Create an HTML document which consists of: Ordered List, Unordered List, Nested ordered and/or unordered List and Image
3. Create an HTML document which implements Internal linking as well as External linking Q. No.4.
4. Create a table using HTML demonstrating use of columns and rows, merging multiple rows and/or columns etc.with data and image values and contents with hyperlinking
5. Create a typical student data capture form for the purpose of admission to your college using different types of
   HTML controls i.e. Text Box,Option/radio buttons, Check boxes, Reset and Submit buttons etc.
6. Create HTML documents having multiple frames in different possible formats/organization
7. Information Structure: In this exercise, student will practice identifying the structure of an information object
8. Deconstructing an XML Document: In this exercise, student will practice identifying the explicit structure within an XML document. In a sense, this is the reverse of what you did in Exercise #7.
9. Creating XML Markup:In this exercise, create some XML markup based on the tree representation from previous exercise and the content from the original sample document.
10. Well-Formedness:This exercise checks your understanding of the constraints for well-formedness based on some sample code document to be provided by the concerned teacher. Are the supplied code document instances well-formed? Explain any NO answers.
11. Create a PHP page using functions for comparing three integers and print the largest number.
12. Write a function to calculate the factorial of a number (non-negative integer). The function accept the number as an argument.

13. WAP to check whether the given number is prime or not.
14. Create a PHP page which accepts string from user. After submission that page displays the reverse of provided string.
15. Write a PHP function that checks if a string is all lower case.
16. Write a PHP script that checks whether a passed string is palindrome or not? (A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run)
17. WAP to sort an array.
18. Write a PHP script that removes the whitespaces
    from a string. Sample string: 'The quick " " brown
    fox' Expected Output :
    Thequick""brownfox
19. Write a PHP script that finds out the sum of first n odd numbers.
20. Create a login page having user name and password. On clicking submit, a welcome message should be displayed if the user is already registered (i.e.name is present in the database) otherwise error message should be displayed.
21. Write a PHP script that checks if a string contains another string.
22. Create a simple 'birthday countdown' script, the script will count the number of days between current day and birth day.
23. Create a script to construct the following pattern, using nested for loop.

```
*
* *
* * *
* * * *
```

24. Write a simple PHP program to check that emails are valid.
25. WAP to print first n even numbers.
26. $color = array('white', 'green', 'red'')

Write a PHP script which will display the colors in the following way

: Output :

white, green, red,

□ green

□ red

□ white

27. Using switch case and dropdown list display a ―Hello‖ message depending on the language selected in drop down list.
28. Write a PHP program to print Fibonacci series using recursion.
29. Write a PHP script to replace the first 'the' of the following string with 'That'. Sample : 'the quick brown fox jumps over the lazy dog.'
    Expected Result : That quick brown fox jumps over the lazy dog.

E3: Python Programming [15 Lectures + 30 Labs]

Unit 1: Planning the Computer Program(2 Lectures)

Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation

Unit 2: Techniques of Problem Solving(2 Lectures)

Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming.

Unit 3: Overview of Programming(3 Lectures)

Structure of a Python Program, Elements of Python

Unit 4: Introduction to Python(4 Lectures)

Python Interpreter, Using Python as calculator, Python shell, Indentation. Atoms, Identifiers and keywords, Literals, Strings, Operators (Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment, Operator, Ternary operator, Bit wise operator, Increment or Decrement operator)

Unit 5: Creating Python Programs(4 Lectures)

Input and Output Statements, Control statements(Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.), Defining Functions, default arguments.

SuggestedReadings

1. T. Budd, Exploring Python, TMH, 1st Ed, 2011
2. Python Tutorial/Documentation www.python.or 2015
3. Allen Downey, Jeffrey Elkner, Chris Meyers , How to think like a computer scientist : learning with Python , Freely available online.2012
4. http://docs.python.org/3/tutorial/index.html
5. http://interactivepython.org/courselib/static/pythonds
6. http://www.ibiblio.org/g2swap/byteofpython/read/

Software Lab Based on Python:

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

Section: A (Simple programs)

1. Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon users choice.
2. WAP to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria :
    a. Grade A: Percentage >=80
    b. Grade B: Percentage>=70 and <80
    c. Grade C: Percentage>=60 and <70
    d. Grade D: Percentage>=40 and <60
    e. Grade E: Percentage<40
3. Write a menu-driven program, using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input paramters from user.
4. WAP to display the first n terms of Fibonacci series.
5. WAP to find factorial of the given number.
6. WAP to implement the use of arrays in Python.
7. WAP to implement String Manipulation in python in Python.
8. WAP to find sum of the following series for n terms: 1 – 2/2! + 3/3! - - - - - n/n!
9. WAP to calculate the sum and product of two compatible matrices.

Section: B (OOPs using Python):

All the programs should be written using user defined functions, wherever possible.

1. WAP to create Class and Objects in Python.
2. WAP to implement Data Hiding in Python.
3. WAP to implement constructor and destructor for a class in Python.
4. WAP to implement constructor and destructor in Python.
5. WAP to implement different types of inheritance in Python.
6. WAP to implement concept of Overriding in Python.
7. Write programs to create mathematical 3D objects using class.
    a. curve b. sphere c. cone d. arrow e. ring f. cylinder

----- End of Semester III-----

2nd Year: Semester IV

CC-8: Computer Networks [Credit: 5, Lecture Hours: 60]

Unit 1: Introduction to Computer Networks (8 Lectures)
Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite.

Unit 2: Data Communication Fundamentals and Techniques (10 Lectures)
Analog and digital signal; data-rate limits; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission; digital to analog modulation-; multiplexing techniques- FDM, TDM; transmission media.

Unit 3: Networks Switching Techniques and Access mechanisms (10 Lectures)
Circuit switching; packet switching- connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.

Unit 4: Data Link Layer Functions and Protocol (10 Lectures)
Error detection and error correction techniques; data-link control- framing and flow control; error recovery protocols-stop and wait ARQ, go-back-n ARQ; Point to Point Protocol on Internet.

Unit 5: Multiple Access Protocol and Networks (5 Lectures)
CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks- repeaters, hubs, switches, bridges, router and gateways;

Unit 6: Networks Layer Functions and Protocols (6 Lectures)
Routing; routing algorithms; network layer protocol of Internet- IP protocol, Internet control protocols

Unit 7: Transport Layer Functions and Protocols (6 Lectures)
Transport services- error and flow control, Connection establishment and release- three way handshaking

Unit 8: Overview of Application layer protocol (5ectures)
Overview of DNS protocol; overview of WWW & HTTP protocol

Suggested Readings

1. B. A. Forouzan: Data Communications and Networking, Fourth edition, THM, 2007.
2. A. S. Tanenbaum: Computer Networks**,** Fourth edition**,** PHI , 2002
3. Dr. Rakesh Kumar Mandal: Computer Networks for Students, First Edition, SPD, 2018

Computer Networks Lab[Credit: 2, Lab Hours: 60]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate and implement stop and wait protocol for noisy channel.
3. Simulate and implement go back n sliding window protocol.
4. Simulate and implement selective repeat sliding window protocol.
5. Simulate and implement distance vector routing algorithm
6. Simulate and implement Dijkstra algorithm for shortest path routing.

CC -9: Software Engineering **[Credit: 4, Lecture Hours: 60]**

Unit 1: Introduction (8 Lectures)
The Evolving Role of Software, Software Characteristics, Changing Nature of Software, Software Engineering as a Layered Technology, Software Process Framework, Framework and Umbrella Activities, Process Models, Capability Maturity Model Integration (CMMI).

Unit 2: Requirement Analysis (10 Lectures)
Software Requirement Analysis, Initiating Requirement Engineering Process, Requirement Analysis and Modeling Techniques, Flow Oriented Modeling, Need for SRS, Characteristics and Components of SRS.

Unit 3: Software Project Management (8Lectures)
Estimation in Project Planning Process, Project Scheduling

Unit 4: Risk Management (8 Lectures)
Software Risks, Risk Identification, Risk Projection and Risk Refinement, RMMM Plan.

Unit 5: Quality Management (8 Lectures)
Quality Concepts, Software Quality Assurance, Software Reviews, Metrics for Process and Projects.

Unit 6: Design Engineering (10 Lectures)
Design Concepts, Architectural Design Elements, Software Architecture, Data Design at the Architectural Level and Component Level, Mapping of Data Flow into Software Architecture, Modeling Component Level Design.

Unit 7: Testing Strategies & Tactics (8 Lectures)
Software Testing Fundamentals, Strategic Approach to Software Testing, Test Strategies for Conventional Software, Validation Testing, System testing, Black-Box Testing, White-Box Testing and their type, Basis Path Testing.

Suggested Readings

1. R.S. Pressman, Software Engineering: A Practitioner„s Approach (7th Edition), McGraw-Hill, 2009.
2. P. Jalote, An Integrated Approach to Software Engineering (2nd Edition), Narosa Publishing House, 2003.
3. K.K. Aggarwal and Y. Singh, Software Engineering (2nd Edition), New Age International Publishers, 2008.
4. I. Sommerville, Software Engineering (8th edition), Addison Wesley, 2006.
5. D. Bell, Software Engineering for Students (4th Edition), Addison-Wesley, 2005.
6. R. Mall, Fundamentals of Software Engineering (2nd Edition), Prentice-Hall of India, 2004.

Software Engineering Lab **[Credit: 2, Lab Hours: 60]**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

Sl. No.	Practical Title
1	• Problem Statement, • Process Model
2	Requirement Analysis: • Creating a Data Flow • Data Dictionary, Use Cases

3	Project Management: • Computing FP • Effort • Schedule, Risk Table, Timeline chart
4	Design Engineering: • Architectural Design • Data Design, Component Level Design
5	Testing: • Basis Path Testing

Sample Projects:

1. **Criminal Record Management**: Implement a criminal record management system for jailers, police officersand CBI officers
2. **DTC Route Information**: Online information about the bus routes and their frequency and fares
3. **Car Pooling**: To maintain a web based intranet application that enables the corporate employees within anorganization to avail the facility of carpooling effectively.
4. Patient Appointment and Prescription Management System
5. Organized Retail Shopping Management Software
6. Online Hotel Reservation Service System
7. Examination and Result computation system
8. Automatic Internal Assessment System
9. Parking Allocation System
10. Wholesale Management System

CC-10: Database Management Systems [Credit: 4, Lecture Hours: 60]

Unit 1: Introduction (6 Lectures)
Characteristics of database approach, data models, database system architectureand data independence.

Unit 2: Entity Relationship(ER) Modeling (8 Lectures)
Entity types, relationships, constraints.

Unit 3: Relation data model (20 Lectures)
Relational model concepts, relational constraints, relational algebra, SQLqueries

Unit 4: Database design (15 Lectures)
Mapping ER/EER model to relational database, functional dependencies,Lossless decomposition, Normal Forms(upto BCNF).

Unit 5: Transaction Processing (3 Lectures)
ACID properties, concurrency control

Unit 6: File Structure and Indexing (8 Lectures)
Operations on files, File of Unordered and ordered records, overview of File organizations, Indexing structures for files(Primary index, secondary index, clustering index), Multilevel indexing using B and B+ trees.

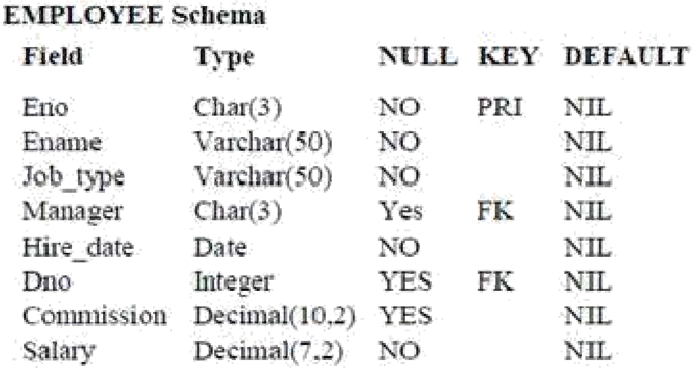
Suggested Readings

1. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education, 2010.
2. R. Ramakrishanan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.

3. A. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGraw Hill, 2010.
4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education,2013.
5. Database Book: The Principles and Practices Using MySQL, NarainGehani, University Press

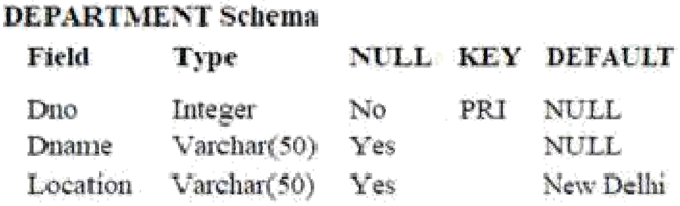
Database Management Systems Lab[Credit: 2, Lab Hours: 60]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:
Create and use the following database schema to answer the given queries.



EMPLOYEE Schema

Field	Type	NULL	KEY	DEFAULT
Eno	Char(3)	NO	PRI	NIL
Ename	Varchar(50)	NO		NIL
Job_type	Varchar(50)	NO		NIL
Manager	Char(3)	Yes	FK	NIL
Hire_date	Date	NO		NIL
Dno	Integer	YES	FK	NIL
Commission	Decimal(10,2)	YES		NIL
Salary	Decimal(7,2)	NO		NIL



DEPARTMENT Schema

Field	Type	NULL	KEY	DEFAULT
Dno	Integer	No	PRI	NULL
Dname	Varchar(50)	Yes		NULL
Location	Varchar(50)	Yes		New Delhi

Query List:

1. Query to display Employee Name, Job, Hire Date, Employee Number; for each employee with the Employee Number appearing first.
2. Query to display unique Jobs from the Employee Table.
3. Query to display the Employee Name concatenated by a Job separated by a comma.
4. Query to display all the data from the Employee Table. Separate each Column by a comma and name the said column as THE_OUTPUT.
5. Query to display the Employee Name and Salary of all the employees earning more than $2850.
6. Query to display Employee Name and Department Number for the Employee No= 7900.
7. Query to display Employee Name and Salary for all employees whose salary is not in the range of $1500 and $2850.
8. Query to display Employee Name and Department No. of all the employees in Dept 10 and Dept 30 in the alphabetical order by name.
9. Query to display Name and Hire Date of every Employee who was hired in 1981.
10. Query to display Name and Job of all employees who don’t have a current Manager.
11. Query to display the Name, Salary and Commission for all the employees who earn commission.
12. Sort the data in descending order of Salary and Commission.
13. Query to display Name of all the employees where the third letter of their name is _A„.
14. Query to display Name of all employees either have two _R„s or have two _A„s in their name and are either in Dept No = 30 or their Manger’s Employee No = 7788.
15. Query to display Name, Salary and Commission for all employees whose Commission Amount is 14 greater than their Salary increased by 5%.
16. Query to display the Current Date.
17. Query to display Name, Hire Date and Salary Review Date which is the 1st Monday after six months of employment.
18. Query to display Name and calculate the number of months between today and the date each employee was hired.

19. Query to display the following for each employee <E-Name> earns < Salary> monthly but wants < 3 * Current Salary >. Label the Column as Dream Salary.
20. Query to display Name with the 1st letter capitalized and all other letter lower case and length of their name of all the employees whose name starts with ‗J‟‟ ‟A‟ and ‗M‟.
21. Query to display Name, Hire Date and Day of the week on which the employee started.
22. Query to display Name, Department Name and Department No for all the employees.
23. Query to display Unique Listing of all Jobs that are in Department # 30.
24. Query to display Name, Dept Name of all employees who have an ‗A‟ in their name.
25. Query to display Name, Job, and Department No. And Department Name for all the employees working at the Dallas location.
26. Query to display Name and Employee no. Along with their Manger's Name and the Manager's employee no; along with the Employees‟ Name who do not have a Manager.
27. Query to display Name, Deptt. No. And Salary of any employee whose department No. and salary matches both the department no. And the salary of any employee who earns a commission.
28. Query to display Name and Salaries represented by asterisks, where each asterisk (*) signifies $100.
29. Query to display the Highest, Lowest, Sum and Average Salaries of all the employees
30. Query to display the number of employees performing the same Job type functions.
31. Query to display the no. of managers without listing their names.
32. Query to display the Department Name, Location Name, No. of Employees and the average salary for all employees in that department.
33. Query to display Name and Hire Date for all employees in the same dept. as Blake.
34. Query to display the Employee No. And Name for all employees who earn more than the average salary.
35. Query to display Employee Number and Name for all employees who work in a department with any employee whose name contains a ‗T‟.
36. Query to display the names and salaries of all employees who report to King.
37. Query to display the department no, name and job for all employees in the Sales department.

GE-4: Generic Elective-4 [Theory: 60 Lectures]

For GE -4, select another paper from separate discipline given in the list of Step 1 of GE -1 which is not chosen for GE -1, GE -2 and GE -3 and follow the respective syllabus.

SEC -2: Any one from the list (E1-E3)

E1: Android Programming [15 Lectures +30 Labs]

Unit 1:Introduction(2 Lectures)
History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture.

Unit 2:Overview of object oriented programming using Java(4 Lectures)

OOPs Concepts: Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.

Unit 3:Development Tools(5 Lectures)
Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating a android project – Hello Word, run on emulator, Deploy it on USB-connected Android device.

Unit 4:User Interface Architecture(2 Lectures)
Application context, intents, Activity life cycle, multiple screen sizes.

Unit 5:User Interface Design(2 Lectures)
Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners(Combo boxes),Images, Menu, Dialog.

Unit 6:Database(2 Lectures)
Understanding of SQLite database, connecting with the database.

Suggested Readings

1. Pradeep Kothari, Andriod Application Development: Black book, Dreamtech Press,
2. James C. Sheusi, Android application development for java programmers, Cengage Learning, 2013.

ONLINE READING / SUPPORTING MATERIAL:

1. http://www.developer.android.com
2. http://developer.android.com/about/versions/index.html
3. http://developer.android.com/training/basics/firstapp/index.html
4. http://docs.oracle.com/javase/tutorial/index.htm (Available in the form of free downloadable ebooks also).
5. http://developer.android.com/guide/components/activities.html
6. http://developer.android.com/guide/components/fundamentals.html
7. http://developer.android.com/guide/components/intents-filters.html.
8. http://developer.android.com/training/multiscreen/screensizes.html
9. http://developer.android.com/guide/topics/ui/controls.html
10. http://developer.android.com/guide/topics/ui/declaring-layout.html
11. http://developer.android.com/training/basics/data-storage/databases.html

Software Lab Based on Android Programming:

1. Create ―Hello World‖ application. That will display ―Hello World‖ in the middle of the screen in the emulator. Also display ―Hello World‖ in the middle of the screen in the Android Phone.
2. Create an application with login module. (Check username and password).
3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, Image will change.
4. Create a menu with 5 options and and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.
6. Create an application with three option buttons, on selecting a button colour of the screen will change.
7. Create and Login application as above. On successful login, pop up the message.
8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.

E2: Programming in MATLAB [15 Lectures + 30 Labs]

Unit 1: Introduction to Programming(2 Lectures)
Components of a computer, working with numbers, Machine code, Software hierarchy.

Unit 2: Programming Environment(3 Lectures)
MATLAB Windows, A First Program, Expressions, Constants, Variables and assignment statement, Arrays.

Unit 3:Graph Plots(2 Lectures)
Basic plotting, Built in functions, Generating waveforms, Sound replay, load and save.

Unit 4: Procedures and Functions(3 Lectures)
Arguments and return values, M-files, Formatted console input-output, String handling.

Unit 5: Control Statements(2 Lectures)
If, Else, Else-if, Repetition statements: While, for loop.

Unit 6: Manipulating Text (2 Lectures)
Writing to a text file, Reading from a text file, Randomising and sorting a list, searching a list.

Unit 7: GUI Interface(2 Lectures)
Attaching buttons to actions, Getting Input, Setting Output.

Recommended Books:

1. MATLAB: An Introduction with Applications, by Amos Gilat, 2nd edition, Wiley, 2004,
2. C.B. Moler, Numerical Computing with MATLAB, SIAM, 2004.

Software Lab Based on MATLAB:
Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Write a program to assign the following expressions to a variable A and then to print out the value of A.
   a. $(3+4)/(5+6)$ b. $2\pi^2$ c. $\sqrt{2}$ d. $(0.0000123 + 5.67\times10^{-3}) \times 0.4567\times10^{-4}$
2. Celsius temperatures can be converted to Fahrenheit by multiplying by 9, dividing by 5, and adding 32. Assign a variable called C the value 37, and implement this formula to assign a variable F the Fahrenheit equivalent of 37 Celsius.
3. Set up a vector called N with five elements having the values: 1, 2, 3, 4, 5. Using N, create assignment statements for a vector X which will result in X having these values:
   a. 2, 4, 6, 8, 10
   b. 1/2, 1, 3/2, 2, 5/2
   c. 1, 1/2, 1/3, 1/4, 1/5
   d. 1, 1/4, 1/9, 1/16, 1/25
4. A supermarket conveyor belt holds an array of groceries. The price of each product (in pounds) is [0.6, 1.2 ,0.5, 1.3] ; while the numbers of each product are [3, 2 ,1 ,5]. Use MATLAB to calculate the total bill.
5. The sortrows(x) function will sort a vector or matrix X into increasing row order. Use this function to sort a list of names into alphabetical order.
6. The ―identity‖ matrix is a square matrix that has ones on the diagonal and zeros elsewhere. You can generate one with the eye() function in MATLAB. Use MATLAB to find a matrix B, such that when multiplied by matrix A=[1 2; -1 0] the identity matrix I=[1 0; 0 1] is generated. That is A*B=I.
7. Create an array of N numbers. Now find a single MATLAB statement that picks out from that array the
   1,4,9,16,…,√Nth entries, i.e. those numbers which have indices that are square numbers.
8. Draw a graph that joins the points (0,1), (4,3), (2,0) and (5,-2).
9. The seeds on a sunflower are distributed according to the formula below. Plot a small circle at each of the first
   1000 co-ordinates :

   $$r_n = \sqrt{n}$$

   $$\theta_n = \frac{137.51}{180}\pi n$$
10. Calculate 10 approximate points from the function y=2x by using the formulae:
    a. i. xn = n
    b. ii. yn = 2n + rand - 0.5

Fit a line of best fit to these points using the function polyfit() with degree=1, and generate co-ordinates from the line of best fit using polyval(). Use the on-line help to find out how to use these functions. Plot the raw data and the line of best fit.

11. Calculate and replay 1 second of a sinewave at 500Hz with a sampling rate of 11025Hz. Save the sound to a file called "ex35.wav". Plot the first 100 samples.
12. Calculate and replay a 2 second chirp. That is, a sinusoid that steadily increases in frequency with time, from say 250Hz at the start to 1000Hz at the end.
13. Build a square wave by adding together 10 odd harmonics: 1f, 3f, 5f, etc. The amplitude of the nth harmonic should be 1/n. Display a graph of one cycle of the result superimposed on the individual harmonics.
14. Write a function called FtoC (ftoc.m) to convert Fahrenheit temperatures into Celsius. Make sure the program has a title comment and a help page. Test from the command window with:
    a. FtoC(96)
    b. lookfor Fahrenheit
    c. help FtoC
15. Write a program to input 2 strings from the user and to print out (i) the concatenation of the two strings with a space between them, (ii) a line of asterisks the same length as the concatenated strings, and (iii) the reversed concatenation. For example:
    a. Enter string 1: Mark
    b. Enter string 2: Huckvale
    c. Mark Huckvale
    d. *************
    e. elavkcuHkraM

E3: VB.NET Programming [15 Lectures + 30 Labs]

Unit 1: Introduction to .NET Framework (2 Lectures)
Introduction, Characteristics, Web Services; The Framework, Components, Common Language Runtime (CLR),
.NET Class Framework vs. A Language - Specific Class Library;

Unit 2: Using .NET Components (3 Lectures)
.NET Windows Forms, Web Forms and Web Services, Console Applications; Microsoft Intermediate Language (MSIL), .NET Compilers, Common Language Runtime, Code reuse, Multiple Language Support in .NET, Cross-Language Interoperability, Garbage Collection, Error Handling; .NET Class Framework, Namespaces, Inheritance: To Use vs. Not To Use, Interface Based Polymorphism vs. Inheritance-Based Polymorphism

Unit 3: Introduction to VB.NET(5 Lectures)
Introduction, Elements, Types, Windows programming, Menus and Dialog Boxes, Using ADO.NET,

Unit 4: Application Development and Deployment using VB.NET (5 Lectures)
Developing Components in Visual Basic .NET; Deploying Applications- Introduction to web services, Assemblies, Deploying Applications

Suggested Readings

1. Introduction to Visual basic.NET - NIIT PHI, 2005
2. Introducing Microsoft .NET- David S. Platt, Microsoft Press, Saarc Edition, 2001

Software Lab based on VB.NET:
Students are to implement different simple suitable applications based on the VB.NET technologies to be assigned by teachers

^

----- End of Semester IV-----

3rd Year: Semester V

CC-11: Internet Technologies **[Credit: 4, Lecture Hours: 60]**

Unit 1:Java (5 Lectures)

Use of Objects, Array and ArrayList class

Unit 2:JavaScript (15 Lectures)

Data types, operators, functions, control structures, events and event handling.

Unit 3:JDBC (10 Lectures)

JDBC Fundamentals, Establishing Connectivity and working with connection interface, working with statements, Creating and Executing SQL Statements, Working with Result Set Objects.

Unit 4:JSP (20 Lectures)

Introduction to Java Server Pages, HTTP and Servlet Basics, The Problem with Servlets, The Anatomy of a JSP Page, JSP Processing, JSP Application Design with MVC, Setting Up the JSP Environment, Implicit JSP Objects, Conditional Processing, Displaying Values, Using an expression to Set an Attribute, Declaring Variables and Methods, Error Handling and Debugging, Sharing Data Between JSP Pages, Requests, and Users, Database Access.

Unit 5:Java Beans (10 Lectures)

Java Beans Fundamentals, JAR files, Introspection, Developing a simple Bean, Connecting to DB

Suggested Readings

1. DT Editorial Services, Web Technologies, Dreamtech Press
2. Ivan Bayross, Web Enabled Commercial Application Development Using Html, Dhtml, javascript, Perl CGI , BPB Publications, 2009.
3. Cay Horstmann, BIG Java, Wiley Publication, 3rd Edition., 2009
4. Herbert Schildt, Java 7, The Complete Reference, 8th Edition, 2009.
5. Jim Keogh, The Complete Reference J2EE, TMH, 2002.
6. O'Reilly, Java Server Pages, Hans Bergsten, Third Edition, 2003.

Internet Technologies Lab **[Credit: 2, Lab Hours: 60]**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

Create event driven program for following:

1. Print a table of numbers from 5 to 15 and their squares and cubes using alert.
2. Print the largest of three numbers.
3. Find the factorial of a number n.
4. Enter a list of positive numbers terminated by Zero. Find the sum and average of these number s.
5. A person deposits Rs 1000 in a fixed account yielding 5% interest. Compute the amount in the account at the end of each year for n years.

Read n numbers. Count the number of negative numbers, positive numbers and zeros in the

CC -12: Theory of Computation **[Credit: 5, Lecture Hours: 60]**

Unit 1:Languages (8 Lectures)

Alphabets, string, language, Basic Operations on language, Concatenation, KleeneStar

Unit 2: Finite Automata and Regular Languages (20 Lectures)

Regular Expressions, Transition Graphs, Deterministics and non-deterministic finite automata, NFA to DFA Conversion, Regular languages and their relationship with finite automata, Pumping lemma and closure properties of regular languages.

Unit 3: Context free languages (17 Lectures)
Context free grammars, parse trees, ambiguities in grammars and languages, Pushdown automata (Deterministic and Non-deterministic), Pumping Lemma, Properties of context free languages, normal forms.

Unit 4: Turing Macines and Models of Computations (15 Lectures)
RAM, Turing Machine as a model of computation, Universal Turing Machine, Language acceptability, decidability, halting problem, Recursively enumerable and recursive languages, unsolvability problems.

Suggested Readings

1. Daniel I.A.Cohen, Introduction to computer theory, John Wiley,1996
2. Lewis & Papadimitriou, Elements of the theory of computation , PHI 1997.
3. Hoperoft, Aho, Ullman, Introduction to Automata theory, Language & Computation –**3rd** Edition, Pearson Education. 2006
4. P. Linz, An Introduction to Formal Language and Automata 4th edition Publication Jones Bartlett, 2006

DSE -1 :DSE: Any one of the following (E1 to E3) [Theory: 60 Lectures]

E1: Microprocessor [Theory: 60 Lectures]

Unit 1: Microprocessor architecture (20 Lectures)
Internal architecture, system bus architecture, memory and I/O interfaces

Unit 2: Microprocessor programming (20 Lectures)
Register Organization, instruction formats, assembly language programming

Unit 3: Interfacing (20 Lectures)
Memory address decoding, cache memory and cache controllers, I/O interface, keyboard, display, timer, interrupt controller, DMA controller, video controllers, communication interfaces.

Suggested Readings

1. Barry B. Brey: The Intel Microprocessors: Architecture, Programming and Interfacing. Pearson Education, Sixth Edition, 2009.
2. Walter A Triebel, Avtar Singh; The 8088 and 8086 Microprocessors Programming, Interfacing, Software, Hardware, and Applications. PHI, Fourth Edition 2005.
3. Ramesh Gaonkar; Microprocessor Architecture, Programming, And Applications With The 8085 5/E, Penram International Publishing(India)
4. B. Ram, Fundamentals of Microprocessors and Microcontrollers, Dhanpat Rai Publications

E2: Information Security [Theory: 60 Lectures]

Unit 1: Introduction (5 Lectures)
Security, Attacks, Computer Criminals, Security Services, Security Mechanisms.

Unit 2: Cryptography (10 Lectures)

Substitution ciphers, Transpositions Cipher, Confusion, diffusion, Symmetric, Asymmetric Encryption. DES Modes of DES, Uses of Encryption, Hash function, key exchange, Digital Signatures, Digital Certificates.

Unit 3: Program Security (10 Lectures)
Secure programs, Non malicious Program errors, Malicious codes virus, Trap doors, Salami attacks, Covert channels, Control against program

Unit 4: Threats (10 Lectures)
Protection in OS: Memory and Address Protection, Access control, File Protection, User Authentication.

Unit 5: Database Security (10 Lectures)
Requirements, Reliability, Integrity, Sensitive data, Inference, Multilevel Security.

Unit 6: Security in Networks (10 Lectures)
Threats in Networks, Security Controls, firewalls, Intrusion detection systems, Secure e-mails

Unit 7: Administrating Security(5 Lectures)
Security Planning, Risk Analysis, Organisational Security Policy, Physical Security. Ethical issues in Security:
Protecting Programs and data. Information and law.

Suggested Readings

1. C. P. Pfleeger, S. L. Pfleeger; Security in Computing, Prentice Hall of India, 2006
2. W. Stallings; Network Security Essentials: Applications and Standards, 4/E, 2010

E3: Modelling and Simulation **[Theory: 60 Lectures]**

Unit 1: System models and System studies (5 Lectures)
Concept of a System, Deterministic and Stochastic Activities, Continuous and DiscreteSystems, System Modeling, Types of Models, Principles used in Modeling, CorporateModel, System Design.

Unit 2: What is Simulation? (7 Lectures)
Technique of Simulation, The Monte-Carlo Method, Comparison of Simulation and AnalyticalMethods, Experimental nature of Simulation, Types of System Simulation, NumericalComputation Techniques for Continuous Models, Numerical Computation Techniques forDiscrete Models, Distributed Lag Models, Cobweb Models.

Unit 3: Continuous System Simulation (7 Lectures)
Continuous System Models, Differential Equations & Applications, Feedback Systems, Simulation of an Autopilot, Interactive systems, Real Time Systems.

Unit 4: Concepts in Discrete Event Simulation (7 Lectures)
The Event Scheduling / Time Advance Algorithm, World Views, Manual Simulation using Event Scheduling, List Processing: Lists-Basic properties and operations, Use of arrays for List Processing, Using Dynamic Allocation and Linked Lists.

Unit 5: Queuing Models (9 Lectures)
Characteristics of Queuing Systems, Arrival and Service Patterns, Queue Discipline, Long RunMeasures of Performance of Queuing Systems, Time-Average Number in System, Server Utilization, Costs in Queuing Problems, Steady State behavior of Infinite Population Markovian Models, Multiserver Queue: M/M/C/∞/∞.

Unit 6: Simulation software (7 Lectures)
Comparison of Simulation Packages with Programming Languages, Classification of Simulation Software: General

Purpose vs Application Oriented Simulation Packages, Desirable Software Features: General Capabilities, H/w and S/w Requirements, Animation and Dynamic Graphics, Statistical Capabilities, General Purpose Simulation Packages, Object Oriented Simulation, Examples of Application-oriented Simulation packages, Simulation in GPSS.

Unit 7: Random Number, Non-Uniform random variate Generation and Monte-Carlo Method (9 Lectures)

Linear Congruential Generators, Testing Random Number Generators: Empirical and Theoretical tests, Non-Uniform
Random Variate Generator: Inverse Transform, Composition. Generating Continuous Random Variates: Uniform,
Exponential, Gamma, and Normal. Generating Discrete Random Variates: Bernoulli, Binomial, Poisson. Monte-Carlo
Method: Evaluation of Integral-Hit or Miss Method.

Unit 8: Analysis of Simulation Data (5 Lectures)

Identifying the Distribution with Data, Types of Simulations with respect to Output Analysis, Stochastic nature of Output Data, Measures of Performance and their Estimation: Point Estimation, Confidence-Interval Estimation, Output Analysis for Terminating Simulations: Statistical Background, Confidence-Intervals with Specified Precision, Output Analysis forSteady State Simulations, Variance-Reduction Technique-Antithetic variants.

Unit 9: Verification and Validation of Simulation Models (4 Lectures)

Model Building- Verification and Validation, Verification of Simulation Models, Calibration and Validation of Models: Validation of Model Assumptions, Validating Input-output Transformations, Input-Output Validation

Suggested Readings

1. Raj Jain, Art of Computer Systems Performance Analysis, John Wiley and Sons, Inc,1991
2. Sheldon M. Ross, Simulation, 4thEd., Elsevier 2008
3. Averill M. Law and W. David Kelton, Simulation Modeling and Analysis, 3rdEd., Tata McGraw-Hill, 2003
4. Geofeerey Gordon, System Simulation, 2ndEd., PHI, 1987
5. Jerry Banks and John S. Carson, Barry L Nelson, Discrete-Event System Simulation, 5thEd., Prentice Hall, 2010
6. NarsinghDeo, System Simulation with Digital Computers, Prentice Hall of India, 1979

DSE -2: Dissertation/OJT/Project/Internship

The students will be allowed to work on any project based on the concepts studied in core / elective or skill based elective courses. The group size should be maximum of three (03) students. Each group will be assigned a teacher as a supervisor who will handle both their theory as well lab classes. A maximum of Four (04) projects would be assigned to one teacher.

----- End of Semester V-----

3rd Year: Semester VI

CC -13: .Mobile Computing **[Credit: 4, Lecture Hours: 60]**

Unit – 1: Introduction to Mobile Communications And Computing (6 Lectures)

Mobile Computing (MC): Introduction to MC, Novel Applications, Limitations, and Architecture. GSM: Mobile Services, System Architecture, Radio Interface, Protocols, Localization and Calling, Handover, Security, and New Data Services.

Unit – 2: (Wireless) Medium Access Control (6 Lectures)

Motivation for a Specialized MAC (Hidden and Exposed Terminals, Near and Far Terminals), SDMA, FDMA, TDMA, CDMA.

Unit – 3: Mobile Network Layer (8 Lectures)
Mobile IP (Goals, Assumptions, Entities and Terminology, Ip Packet Delivery, Agent Advertisement and Discovery, Registration, Tunneling And Encapsulation, Optimizations), Dynamic Host Configuration Protocol (DHCP).

Unit – 4: Mobile Transport Layer (8 Lectures)
Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast Retransmit/Fast Recovery, Transmission /Time-Out Freezing, Selective Retransmission, Transaction Oriented TCP.

Unit – 5: Database Issues (8 Lectures)
Hoarding Techniques, Caching Invalidation Mechanisms, Client Server Computing With Adaptation, Power-Aware And Context-Aware Computing, Transactional Models, Query Processing, Recovery, And Quality Of Service Issues.

Unit – 6: Data Dissemination (8 Lectures)
Communications Asymmetry, Classification of New Data Delivery Mechanisms, Pushbased Mechanisms, Pull-Based Mechanisms, Hybrid Mechanisms, Selective Tuning (Indexing) Techniques.

Unit – 7: Mobile Ad Hoc Networks (MANETS) (8 Lectures)
Overview, Properties of A MANET, Spectrum Of MANET Applications, Routing And Various Routing Algorithms, Security In MANET.

Unit – 8: Protocols and Tools (8 Lectures)
Wireless Application Protocol-WAP. (Introduction, Protocol Architecture, and Treatment Of Protocols Of All Layers), Bluetooth (User Scenarios, Physical Layer, Mac Layer, Networking, Security, Link Management) And J2me.

Suggested Readings

1. JochenSchiller,“MobileCommunications”,Addison-Wesley. second edition, 2004.
2. Stojmenovic and Cacute, “Handbook of Wireless Networks and Mobile Computing”, Wiley, 2002, ISBN 0471419028.
3. Reza Behravanfar, “Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML”, ISBN: 0521817331, Cambridge University Press, October 2004,
4. Adelstein, Frank, Gupta, Sandeep KS, Richard III, Golden ,Schwiebert, Loren, “Fundamentals of Mobile and Pervasive Computing”, ISBN: 0071412379, McGraw-Hill Professional, 2005.
5. Hansmann, Merk, Nicklous, Stober, “Principles of Mobile Computing”, Springer, second edition, 2003.
6. Martyn Mallick, “Mobile and Wireless Design Essentials”, Wiley DreamTech, 2003.

Mobile Computing Lab **[Credit: 2, Lab Hours: 60]**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Write a WML program to create a card.
2. Write a WML program to create a deck that contain two cards and provide the Functionality of calling two cards from one another.
3. Write a WML program to implement the functionality of Login by username.
4. Write a WML program to display special characters on the screen.

5. Write a WML program to display the image on the screen after 5 seconds.
6. Write a WML program to develop the calculator.
7. Write a WML program to develop a salary calculator.
8. Write a WML program to develop a loan calculator.

CC -14: Design and Analysis of Algorithms [Credit: 4, Lecture Hours: 60]

Unit 1: Introduction (5 Lectures)
Basic Design and Analysis techniques of Algorithms, Correctness of Algorithm

Unit 2: Algorithm Design Techniques (8 Lectures)
Iterative techniques, Divide and Conquer, DynamicProgramming,

Greedy Algorithms

Unit 3: Sorting and Searching Techniques (20 Lectures)

Elementary sorting techniques–Bubble Sort, InsertionSort, Merge Sort, Advanced Sorting techniques - Heap Sort, Quick Sort, Sorting in Linear Time - Bucket Sort, Radix Sort and Count Sort, Searching Techniques, Medians & Order Statistics, complexity analysis;

Unit 4: Lower Bounding Techniques (5 Lectures)
Decision Trees

Unit 5: Balanced Trees (7 Lectures)
Red-Black Trees

Unit 6: Advanced Analysis Technique (5 Lectures)
Amortized analysis

Unit 7: Graphs (5 Lectures)
Graph Algorithms–Breadth First Search, Depth First Search and its Applications,Minimum Spanning Trees.

Unit 8: String Processing (5Lectures)
String Matching, KMP Technique

Suggested Readings

1. T.H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein Introduction to Algorithms, PHI, 3rd Edition 2009
2. Sarabasse& A.V. Gelder Computer Algorithm – Introduction to Design and Analysis, Publisher – Pearson 3rd Edition 1999
3. Rajesh K. Shukla, Analysis and Design of Algorithms: A Beginner's Approach, Wiley

Design and Analysis of Algorithms Lab [Credit: 2, Lab Hours: 60]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Implement Insertion Sort (The program should report the number of comparisons)
2. Implement Merge Sort(The program should report the number of comparisons)
3. Implement Heap Sort(The program should report the number of comparisons)
4. Implement Randomized Quick sort (The program should report the number of comparisons)
5. Implement Radix Sort
6. Create a Red-Black Tree and perform following operations on it:

6.1. Insert a node
6.2. Delete a node
6.3. Search for a number & also report the color of the node containing this number.

7. Write a program to determine the LCS of two given sequences
8. Implement Breadth-First Search in a graph
9. Implement Depth-First Search in a graph
10. Write a program to determine the minimum spanning tree of a graph
11. For the algorithms at S.No 1 to 3 test run the algorithm on 100 different inputs of sizes varying from 30 to 1000. Count the number of comparisons and draw the graph. Compare it with a graph of *n*log*n*.

DSE -3:Any one from the following should be chosen (E1-E3)

E1: Operational Research **[Theory: 60 Lectures]**

Unit 1: Introductory Linear Algebra (8 Lectures)
System of linear equations, Matrices, Rank and Determinant of a matrix, Linearly dependent and independent vectors, Basis of a matrix.

Unit 2: Linear programming - I (12 Lectures)
Optimization Problems, Introduction to LP Formulation, Convex sets, Extreme points, Geometry of Linear Programs, Basic feasible solutions (BFS), Neighborhoods, Local and global optima, Profitable Column, Pivoting, Simplex Algorithm with initial BFS, Graphical method

Unit 3: Linear programming - II (12 Lectures)
Degeneracy and Bland'sAnticycling rule (Definition), Simplex Algorithm without initial BFS, Artificial variable techniques – two phase method, M-Charnes method, special cases in LPP.

Unit 4: Transportation Models (8 Lectures)
Transportation Algorithm, Assignment model, Hungarian Method

Unit 6: Network Analysis(10 Lectures)
Shortest Path: Floyd Algorithm; Maximal Flow Problem (Ford-Fulkerson); PERT-CPM : Introduction, Critical Path, Cost Analysis.

Unit 7: Game Theory (10 Lectures)
Introduction; 2-Person Zero-sum Game; Saddle Point; Mini-Max and Maxi-Min Theorems (statement only) and problems; Games without Saddle Point; Principle of Dominance.

Suggested Readings

1. G. Hadley: Linear Programming. Narosa, 2002 (reprint).
2. A. Ravindran, D. T. Phillips and James J. Solberg: Operations Research-Principles and Practice, John Wiley & Sons, 2005.
3. Hamdy A. Taha: Operations Research-An Introduction, Prentice Hall, 8th Edition, 2008.
4. F.S. Hillier. G.J.Lieberman: Introduction to Operations Research- Concepts and Cases, 9th Edition, Tata McGraw Hill. 2010.

E2: Combinatorial Optimization **[Theory: 60 Lectures]**

Unit 1: Introduction

Optimization problems, neighbourhoods, local and global optima, convex sets and functions, simplex method, degeneracy; duality and dual simplex algorithm, computational considerations for the simplex and dual simplex algorithms-Dantzig-Wolfe algorithms.

Unit 2: Integer Linear Programming
Cutting plane algorithms, branch and bound technique and approximation algorithms for travelling salesman problem.

Suggested Readings

1. C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall of India, 2006
2. K. Lange, *Optimization*, Springer, 2004.
3. Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali, *Linear Programming and Network Flows*, John Wiley and Sons, 2004.
4. H.A. Taha, *Operations Research: An Introduction,* 8th Ed., Prentice Hall, 2006.

E3: Numerical Methods **[Theory: 60 Lectures]**

Unit 1: Introduction(20 Lectures)
Floating point representation and computer arithmetic, Significant digits, Errors: Round-off error, Local truncation error, Global truncation error,Absolute error, Relative error, Percentage Error, Rules of rounding off, Convergence
and terminal conditions, Efficient computations: Bisection method, Regula−Falsi method , Newton−Raphson method, Gauss elimination method and Gauss−Jordan method, Matrix inversion.

Unit 2: Iterative methods(8 Lectures)
Jacobi and Gauss-Seidel iterative methods

Unit 3: Interpolation(8 Lectures)
Lagrange's Interpolation, Newton forward and backward Interpolation

Unit 4: Numerical integration(10 Lectures)
Trapezoid rule, Simpson's rule (only method)

Unit 5: Extrapolation methods(7 Lectures)
Gaussian quadrature, Ordinary differential equation: Euler's method ,Taylor Series Expansion, Runge-Kutta second method , Classical 4th order Runge-Kutta method

Unit 6: Eigen-values & Eigen-vectors(3 Lectures)
Definitions and properties

Unit 7: Fitting(4 Lectures)
Fitting of a straight line, Least square method of curve fitting.
Suggested Readings

1. Laurence V. Fausett, Applied Numerical Analysis, Using MATLAB, Pearson, 2/e (2012)
2. M.K. Jain, S.R.K. Iyengar and R.K. Jain, Numerical Methods for Scientific and Engineering Computation, New Age International Publisher, 6/e (2012)
3. Steven C Chapra, Applied Numerical Methods with MATLAB for Engineers and Scientists, Tata McGraw Hill, 2/e (2010)

DSE Lab **[Theory: 60 Lectures]**

E1L: Microprocessor Lab **[2Credits; 60 Lectures]**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems: ASSEMBLY LANGUAGE PROGRAMMING in 8085

1. Write an assembly language program to check whether a number is even or odd.
2. Write an assembly language program to find the number of ones and the number of zeros in an 8-bit number.
3. Write an assembly language program to find the smaller of two numbers.
4. Write an assembly language program to multiply two 8-bit numbers.
5. Write an assembly language program to find the smallest among 10 integers stored in memory locations starting from 2050H
6. Write an assembly language program to find the largest among 10 integers stored in memory locations starting from 2050H
7. Write an assembly language program to sort 10 numbers using bubble sort.
8. Write an assembly language program to generate the first 10 fibonacci series and store the result at memory location starting from FC50H
9. Write an assembly language program to find the sum of the series 12 + 22 + 32 + 42 + ….. + 102
10. Write an assembly language program to find the perfect square of any number and if the number is not a perfect square, display FFH
11. Write an assembly language program for linear search.
12. Write an assembly language program to calculate the following expression using a single register: Y = X2 + 2X + 3XZ
13. Write an assembly language program to find the sum of the first 10 even natural numbers.
14. Write an assembly language program to find the sum of the first n odd natural numbers.
15. Write an assembly language program to create an odd parity generator.
16. Write an assembly language program to create an even parity generator.
17. Write an assembly language program to find the sum of five 8-bit numbers.
18. Write an assembly language program to convert decimal to binary.
19. Write an assembly language program to convert octal to binary.
20. Write an assembly language program to convert hexadecimal to binary.
21. Write an assembly language program to convert hexadecimal to decimal.
22. Write an assembly language program to check whether an 8-bit number is palindrome or not.
23. Write an assembly language program to display the truth table for and AND gate.
24. Write an assembly language program to display the truth table for and OR gate.
25. Write an assembly language program to display the truth table for and XOR gate.
26. Write an assembly language program to perform n byte addition of two numbers.
27. Write an assembly language program to implement a simple sub routine call.
28. Write an assembly language program to check whether a number is prime or not.

E2L: Information Security Lab [2 Credits, 60 Lectures]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

b. Demonstrate the use of Network tools: ping, ipconfig, ifconfig, tracert, arp, netstat, whois
c. Use of Password cracking tools: John the Ripper, Ophcrack. Verify the strength of passwords using these tools.
d. Perform encryption and decryption of Caesar cipher. Write a script for performing these operations.
e. Perform encryption and decryption of a Rail fence cipher. Write a script for performing these operations.
f. Use nmap/zenmap to analyse a remote machine.
g. Use Burp proxy to capture and modify the message.
h. Demonstrate sending of a protected word document.
i. Demonstrate sending of a digitally signed document.
j. Demonstrate sending of a protected worksheet.
k. Demonstrate use of steganography tools.
l. Demonstrate use of gpg utility for signing and encrypting purposes.

E3L: Modelling and Simulation Lab [2 Credits, 60 Lectures]

Practical exercises based on theory.

DSE 54L: Any one from the following corresponding to paper selected for DSE54

E1L: Operational Research Lab [2 Credits; 60 Lectures]

Practical based on topics covered in the paper.

E2L: Combinatorial Optimization Lab [2 Credits, 60 Lectures]

Practical exercises based on concepts listed in theory.

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems recommended to be done in any one of Computer Algebra Systems: MATLAB / MATHEMATICA / MAPLE:

1. Find the roots of the equation by bisection method.
2. Find the roots of the equation by secant/Regula−Falsi method.
3. Find the roots of the equation by Newton„s method.
4. Find the solution of a system of nonlinear equation using Newton„s method.
5. Find the solution of tridiagonal system using Gauss Thomas method.
6. Find the solution of system of equations using Jacobi/Gauss-Seidel method.
7. Find the cubic spline interpolating function.
8. Evaluate the approximate value of finite integrals using Gaussian/Romberg integration.
9. Solve the boundary value problem using finite difference method.

DSE-4: Any one from the list should by chosen (E1-E3)

E1: Digital Image Processing [4 Credits; 60 Lectures]

Unit 1:Introduction (6 Lectures)
Light, Brightness adaption and discrimination, Pixels, coordinate conventions, Imaging Geometry, Perspective Projection, Spatial Domain Filtering, sampling and quantization.

Unit 2:Spatial Domain Filtering (7 Lectures)
Intensity transformations, contrast stretching, histogram equalization, Correlation and convolution, Smoothing filters, sharpening filters, gradient and Laplacian.

Unit 3:Filtering in the Frequency domain (8 Lectures)
Hotelling Transform, Fourier Transforms and properties, FFT (Decimation in Frequency and Decimation in Time Techniques), Convolution, Correlation, 2-D sampling, Discrete Cosine Transform, Frequency domain filtering.

Unit 4:Image Restoration (8 Lectures)
Basic Framework, Interactive Restoration, Image deformation and geometric transformations, image morphing, Restoration techniques, Noise characterization, Noise restoration filters, Adaptive filters, Linear, Position invariant degradations, Estimation of Degradation functions, Restoration from projections.

Unit 5: Image Compression (10 Lectures)
Encoder-Decoder model, Types of redundancies, Lossy and Lossless compression, Entropy of an information source, Shannon's 1st Theorem, Huffman Coding, Arithmetic Coding, Golomb Coding, LZW coding, Transform Coding, Sub-image size selection, blocking artifacts, DCT implementation using FFT, Run length coding, FAX compression

(CCITT Group-3 and Group-4), Symbol-based coding, JBIG-2, Bit-plane encoding, Bit-allocation, Zonal Coding, Threshold Coding, JPEG, Lossless predictive coding, Lossy predictive coding, Motion Compensation

Unit 6:Wavelet based Image Compression (5 Lectures)
Expansion of functions, Multi-resolution analysis, Scaling functions, MRA refinement equation, Wavelet series expansion, Discrete Wavelet Transform (DWT), Continuous Wavelet Transform, Fast Wavelet Transform, 2-D wavelet Transform, JPEG-2000 encoding, Digital Image Watermarking.

Unit 7:Morphological Image Processing (7 Lectures)
Basics, SE, Erosion, Dilation, Opening, Closing, Hit-or-Miss Transform, Boundary Detection, Hole filling, Connected components, convex hull, thinning, thickening, skeletons, pruning, Geodesic Dilation, Erosion, Reconstruction by dilation and erosion.

Unit 8: Image Segmentation (9 Lectures)
Boundary detection based techniques, Point, line detection, Edge detection, Edge linking, local processing, regional processing, Hough transform, Thresholding, Iterative thresholding, Otsu's method, Moving averages, Multivariable thresholding, Region-based segmentation, Watershed algorithm, Use of motion in segmentation

Suggested Readings

1. R C Gonzalez , R E Woods, Digital Image Processing, 3rd Edition, Pearson Education.2008.
2. A K Jain, Fundamentals of Digital image Processing, Prentice Hall of India.1989.
3. K R Castleman, Digital Image Processing, Pearson Education.1996
4. Schalkoff, Digital Image Processing and Computer Vision, John Wiley and Sons.1989.
5. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins,' Digital Image Processing using MATLAB', Pearson Education, Inc., 2004.

E2: Introduction to Data Sciences [4 Credits; 60 Lectures]

Unit 1:Data Scientist’s Tool Box(10 Lectures)
Turning data into actionable knowledge, introduction to the tools that will be used in building data analysis software:
version control, markdown, git, GitHub, R, and RStudio

Unit 2:R Programming Basics(15 Lectures)
Overview of R, R data types and objects, reading and writing data, Control structures, functions, scoping rules, dates and times, Loop functions, debugging tools, Simulation, code profiling

Unit 3:Getting and Cleaning Data(15 Lectures)
Obtaining data from the web, from APIs, from databases and from colleagues in various formats, basics of data cleaning and making data ―tidy

Unit 4:Exploratory Data Analysis(10 Lectures)
Essential exploratory techniques for summarizing data, applied before formal modeling commences, eliminating or sharpening potential hypotheses about the world that can be addressed by the data, common multivariate statistical techniques used to visualize high-dimensional data.

Unit 5:Reproducible Research(10 Lectures)
Concepts and tools behind reporting modern data analyses in a reproducible manner, To write a document using R markdown, integrate live R code into a literate statistical program, compile R markdown documents using knitr and related tools, and organize a data analysis so that it is reproducible and accessible to others.

Suggested Readings

1. D. Cielen, Arno D. B. Meysman, M. Ali, Introducing Data Science, Dreamtech Press

2. Rachel Schutt, Cathy O'Neil, "Doing Data Science: Straight Talk from the Frontiline" by Schroff/O'Reilly, 2013.
3. Foster Provost, Tom Fawcett, "Data Science for Business" What You Need to Know About Data Mining and Data-Analytic Thinking" by O'Reilly, 2013.
4. John W. Foreman, "Data Smart: Using data Science to Transform Information into Insight" by John Wiley & Sons, 2013.
5. Ian Ayres, "Super Crunchers: Why Thinking-by-Numbers Is the New Way to Be Smart" Ist Edition by Bantam, 2007.
6. Eric Seigel, "Predictive Analytics: The Power to Predict who Will Click, Buy, Lie, or Die", 1st Edition, by Wiley, 2013.
7. Matthew A. Russel, "Mining the Social Web: Data mining Facebook, Twitter, Linkedln, Goole+, GitHub, and More", Second Edition, by O'Reilly Media, 2013.
8. Michael J. Crawley, The R Book, Wiley

E3: Data Mining **[4 Credits; 60 Lectures]**

Unit 1: Introduction (6 Lectures)
Basic Data Mining Tasks, Data Mining Issues, Data Mining Metrics, Data Mining from a Database Perspective

Unit 2: Data Mining Techniques (10 Lectures)
A Statistical Perspective on Data Mining, Similarity Measures, Decision Trees, Neural Networks, Genetic Algorithms

Unit 3: Classification (15 Lectures)
Statistical-Based Algorithms, Distance-Based Algorithms, Decision Tree-Based Algorithms, Neural Network-Based Algorithms, Rule-Based Algorithms, Combining Techniques.

Unit 4: Clustering (12 Lectures)
Similarity and Distance Measures, Hierarchical Algorithms, Partitional Algorithms, Clustering Large Databases, Clustering with Categorical Attributes

Unit 5: Association Rules (12 Lectures)
Basic Algorithms, Parallel and Distributed Algorithms, Incremental Rules, Advanced Association Rule Techniques, Measuring the Quality of Rules.

Unit6: Advanced Techniques (5 Lectures)
Web Mining, Spatial Mining, Temporal Mining.

Suggested Readings

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Introduction to Data Mining, Pearson Education.2005.
2. Richard Roiger, Michael Geatz, Data Mining: A Tutorial Based Primer, Pearson Education 2003.
3. G.K. Gupta, Introduction to Data Mining with Case Studies, PHI,2006.
4. Soman K P, DiwakarShyam, Ajay V Insight Into Data Mining: Theory And Practice, , PHI, 2006
5. Data Mining Techniques, Arun Pujari, Fourth Edition, University Press

DSE-4 Lab: Any one of the following (E1L-E3L)

E1L: Digital Image Processing Lab **[2 Credits; 60 Lectures]**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

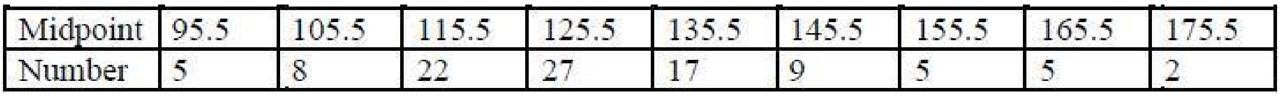
1. Write program to read and display digital image using MATLAB or SCILAB 1.1. Become familiar with SCILAB/MATLAB Basic commands 1.2. Read and display image in SCILAB/MATLAB
   1.3. Resize given image
   1.4. Convert given color image into gray-scale image
   1.5. Convert given color/gray-scale image into black & white image
   1.6. Draw image profile
   1.7. Separate color image in three R G & B planes
   1.8. Create color image using R, G and B three separate planes
   1.9. Flow control and LOOP in SCILAB
   1.10.Write given 2-D data in image file
2. To write and execute image processing programs using point processing method 2.1. Obtain Negative image
   2.2. Obtain Flip image
   2.3. Thresholding
   2.4. Contrast stretching
3. To write and execute programs for image arithmetic operations 3.1. Addition of two images
   3.2. Subtract one image from other image
   3.3. Calculate mean value of image
   3.4. Different Brightness by changing mean value
4. To write and execute programs for image logical operations 4.1. AND operation between two images 4.2. OR operation between two images
   4.3. Calculate intersection of two images
   4.4. Water Marking using EX-OR operation
   4.5. NOT operation (Negative image)
5. To write a program for histogram calculation and equalization using 5.1. Standard MATLAB function
   5.2. Program without using standard MATLAB functions
   5.3. C Program
6. To write and execute program for geometric transformation of image 6.1. Translation
   6.2. Scaling c. Rotation
   6.3. Shrinking
   6.4. Zooming
7. To understand various image noise models and to write programs for 7.1. image restoration
   7.2. Remove Salt and Pepper Noise
   7.3. Minimize Gaussian noise
   7.4. Median filter and Weiner filter
8. Write and execute programs to remove noise using spatial filters
   8.1. Understand 1-D and 2-D convolution process
   8.2. Use 3x3 Mask for low pass filter and high pass filter
9. Write and execute programs for image frequency domain filtering
   9.1. Apply FFT on given image
   9.2. Perform low pass and high pass filtering in frequency domain
   9.3. Apply IFFT to reconstruct image

10. Write a program in C and MATLAB/SCILAB for edge detection using different edge detection mask
11. Write and execute program for image morphological operations erosion and dilation.
12. To write and execute program for wavelet transform on given image and perform inverse wavelet transform to reconstruct image.

E2L: Introduction to Data Sciences Lab **[2 Credits; 60 Lectures]**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Write a program that prints „Hello World„ to the screen.
2. Write a program that asks the user for a number n and prints the sum of the numbers 1 to n
3. Write a program that prints a multiplication table for numbers up to 12.
4. Write a function that returns the largest element in a list.
5. Write a function that computes the running total of a list.
6. Write a function that tests whether a string is a palindrome.
7. Implement linear search.
8. Implement binary search.
9. Implement matrices addition , subtraction and Multiplication
10. Fifteen students were enrolled in a course. There ages were:
    20 20 20 20 20 21 21 21 22 22 22 22 23 23 23
    a. Find the median age of all students under 22 years
    b. Find the median age of all students
    c. Find the mean age of all students
    d. Find the modal age for all students
    e. Two more students enter the class. The age of both students is 23. What is now mean, mode and median?
11. Following table gives a frequency distribution of systolic blood pressure. Compute all the measures of dispersion



Midpoint	95.5	105.5	115.5	125.5	135.5	145.5	155.5	165.5	175.5
Number	5	8	22	27	17	9	5	5	2

12. Obtain probability distribution of, where X is number of spots showing when a six-sided symmetric die (i.e. all six faces of the die are equally likely) is rolled. Simulate random samples of sizes 40, 70 and 100 respectively and verify the frequency interpretation of probability.
13. Make visual representations of data using the base, lattice, and ggplot2 plotting systems in R, apply basic principles of data graphics to create rich analytic graphics from available datasets.
14. Use Git / Github software to create Github account. Also, create a repo using Github.

E3L: Data Mining Lab **[2 Credits; 60 Lectures]**

Practical exercises based on concepts listed in theory.

----- End of Semester VI-----