UG Syllabus Under CBCS
Data Science Management (DSM) (2020)
Scheme of UG DSM Under Choice Based Credit System (CBCS)
Courses for Honours Students

SEM.	COURSE	COURSE CODE	MARKS	CREDITS	TITLE	REMARKS
I	Core	C-1	100	6	Digital Logic	Compulsory
		C-2	100	6	Programming using C	
	Generic Elective	GE-1	100	6	Organizational Behaviour	Compulsory
	Ability Enhancement	AEC-1	100	4	Environmental Science	Compulsory
II	Core	C-3	100	6	Data Structure	Compulsory
		C-4	100	6	Introduction to Probability and Statistics	
	Generic Elective	GE-2	100	6	A. Financial Management	The Department may offer any one of the two or both the courses
					B. General Management	
	Ability Enhancement	AEC-2	100	4	MIL (Odia/Hindi/ English)	Compulsory; The students have to choose any one of the courses
III	Core	C-5	100	6	Computer Organization	Compulsory
		C-6	100	6	Operating Systems	
		C-7	100	6	Database Management Systems	

SEM.	COURSE	COURSE CODE	MARKS	CREDITS	TITLE	REMARKS
	Generic Elective	GE-3	100	6	Decision Management Systems	Compulsory
	Skill Enhancement	SEC-1	100	4	Communicative English	Compulsory
IV	Core	C-8	100	6	Foundation of Data Science and Analytics	Compulsory
		C-9	100	6	JAVA Programming	
		C-10	100	6	Data Communications & Networking	
	Generic Elective	GE-4	100	6	Marketing Analytics	Compulsory
	Skill Enhancement	SEC-2	100	4	Quantitative Aptitude and Logical Reasoning	Compulsory

SEM.	COURSE	COURSE CODE	MARKS	CREDITS	TITLE	REMARKS
V	Core	C-11	100	6	Big Data Analytics	Compulsory
		C-12	100	6	Cloud Computing	
	Discipline Specific	DSE-1	100	6	Software Engineering	Compulsory
		DSE-2	100	6	Python Programming	Compulsory
	Skill Enhancement	SEC-3	100	4	R Programming	Compulsory
VI	Core	C-13	100	6	Machine Learning	Compulsory
		C-14	100	6	Internet of Things (IOT)	Compulsory
	Discipline Specific	DSE-3	100	6	Wireless Communications	Compulsory
		DSE-4	100	6	Project	Compulsory
	Skill Enhancement	SEC-4	100	4	Leadership and Personality Development	Compulsory

DSM (HONOURS)
SEMESTER I
Core Course
C-1: DIGITAL LOGIC (6 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1

Character Codes, Decimal System, Binary System, Decimal to Binary Conversion, Hexadecimal Notation, Boolean Algebra, Basic Logic Functions: Electronic Logic Gates, Synthesis of Logic Functions, Minimization of Logic Expressions, Minimization using Karnaugh Maps, Synthesis with NAND and NOR Gates, Tri-State Buffers

Unit-2

Arithmetic: Addition and Subtraction of Signed Numbers, Addition/ Subtraction Logic Unit, Design of Fast Adders: Carry-Lookahead Addition, Multiplication of Positive Numbers, Signed-Operand Multiplication: Booth Algorithm, Fast Multiplication: Bit-Pair Recodng Multipliers, Carry-Save Addition of Summands, Integer Division, Floating-Point Numbers and Operations: IEEE Standard for Floating-Point Numbers, Arithmetic Operations on Floating-Point Numbers, Guard Bits and Truncation, Implementing Floating-Point Operations.

Unit-3

Flip-Flops, Gated Latches, Master-Slave Flip-Flops, Edge-Triggering, T Flip-Flops, JK Flip-Flops. Registers and Shift Registers, Counters, Decoders, Multiplexers, Programmable Logic Devices (PLDs), Programmable Array Logic (PAL), Complex Programmable Logic Devices (CPLDs), Field-Programmable Gate Array (FPGA), Sequential Circuits, UP/ DOWN Counters, Timing Diagrams, The Finite State Machine Model, Synthesis of Finite State Machines.

Unit-4

Memory System: Semiconductor RAM Memories, Internal Organization of Memory Chips, Static Memories, Asynchronous DRAMS, Synchronous DRAMS, Structure of Large Memories, Memory System Considerations, RAMBUS Memory. Read-Only Memories: ROM, PROM, EPROM, EEPROM, Flash Memory, Speed, Size, and Cost of Memory. Secondary Storage: Magnetic Hard Disks, Optical Disks, Magnetic Tape Systems.

Text Books:

1. Carl Hamacher, Z. Vranesic, S. Zaky: Computer Organization, 5/e (TMH)

Reference Books:

1. M. Morris Mano: Digital Logic and Computer Design, Pearson

CORE–1 Practical: Digital Logic Lab

1. Introduction to Xilinx software (VHDL)

Write the VHDL code for

2. Realizing all logic gates.

3. Combination Circuit.

4. ADDER.

5. SUBTRACTOR.

6. MUX.

7. DE-MUX.

8. Encoder.

9. Decoder.

10. PAL.

11. PLA.

Write the VHDL program for the following Sequential Logic Circuits

12. Flip Flops.

13. Shift Registers.

14. Counters.

15. Memory Elements.

DSM (HONOURS)
SEMESTER I
Core Course
C-2: PROGRAMMING USING C (6 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1

Introduction:Introduction to Programming Language, Introduction to C Programming, Keywords & Identifiers, Constants, Variables, Input and Output Operations, Compilation and pre-processing, **Data types**: Different data types, Data types qualifier, modifiers, Memory representation, size and range, **Operators:** Operators (Arithmetic, Relational, Logical, Bitwise, Assignment & compound assignment, Increment & Decrement, Conditional),Operator types(unary, binary,ternary). Expressions, Order of expression (Precedence and associativity)

Control structures: Decision Making and Branching (Simple IF Statement, IF…ELSE Statement, Nesting IF… ELSE Statement, ELSE IF Ladder), Selection control structure (Switch Statement).

Unit-2

Loops: The WHILE Statement, The DO…WHILE Statement, The FOR Statement, Jumps in Loops, **Array:** Concept of Array, Array Declaration, types of array (one and multiple dimension), Character Arrays and Strings, Subscript and pointer representation of array, Array of Pointers, Limitation of array, **Pointers**: Concept of Pointer (null pointer, wild pointer, dangling pointer, generic pointer), Pointer Expressions, Accessing the Address of a Variable, Declaring Pointer Variables, Initializations of Pointer Variable, Accessing a Variable through its Pointer, Pointer arithmetic.

Unit-3

class: Types (auto,register,static,extern), scope rules, declaration and definition.**Function**: Function & types (User defined function, library function) Function Definition, Declaration, Function Calls, Header file and library, Function Arguments,string handling function (strlen,strcmp,strcpy,strncpy,strcat,strstr),Function recursion, Functions Returning Pointers, Pointers to Functions, Command line arguments, Application of pointer(dynamic memory allocation).

Unit-4

Structure and Union: Defining, Declaring, Accessing, Initialization Structure, nested structure, self-referential structure, bit-field, Arrays of Structures, Structures and Functions, Unions, difference between structure and union, active data member, structure within union, Self-referential Structure.

File: File Management in C, Defining and Opening a File, File opening modes(read,write,append), Closing a File, File operations, file and stream, Error Handling During I/O Operations, sequential and random access file,low level and high level file.

Text Books:

1. E. Balagurusamy, "Programming in ANSI C",4/e, (TMH)

Reference Books:

1. B. Kernighan & Dennis Ritchie, "The C Programming Language", 2/e PHI
2. Paul Deitel, Harvey Deitel, "C: How to Program", 8/e, Prentice Hall.
3. P.C. Sethi, P.K. Behera, "Programming using C", Kalyani Publisher, Ludhiana

Practical/Tutorial: Programming Fundamentals using C Lab

1. Write a Program to find greatest among three numbers.
2. Write a Program to all arithmetic operation using switch case.
3. Write a Program to print the sum and product of digits of an integer.
4. Write a Program to reverse a number.
5. Write a Program to compute the sum of the first n terms of the following series

   S = 1+1/2+1/3+1/4+……
6. Write a Program to compute the sum of the first n terms of the following series

   S =1-2+3-4+5…….
7. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
8. Write a function to find whether a given no. is prime or not. Use the same to generate theprime numbers less than 100.
9. Write a Program to compute the factors of a given number.
10. Write a program to swap two numbers using macro.
11. Write a Program to print a triangle of stars as follows (take number of lines from user):

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

12. Write a Program 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.
13. Write a Program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
14. Write a program that swaps two numbers using pointers.
15. Write a program in which a function is passed address of two variables and then alter its contents.
16. 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.
17. Write a program to find sum and average of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc( ) functions.
18. Write a menu driven program to perform following operations on strings:
    a) Show address of each character in string
    b) Concatenate two strings without using strcat function.

c) Concatenate two strings using strcat function.
d) Compare two strings
e) Calculate length of the string (use pointers)
f) Convert all lowercase characters to uppercase
g) Convert all uppercase characters to lowercase
h) Calculate number of vowels
i) Reverse the string

19. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
20. Write a program to copy the content of one file to other.

ITM (HONOURS)
SEMESTER I
General Elective Course
GE-1: ORGANIZATIONAL BEHAVIOUR (Theory: 4 Credits; Practical: 2 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1
Organizational Behaviour - Meaning, Definition and importance, Foundations of OB, OB Models, and Challenges to OB.

Unit-2
Individual Behaviour
Perception: Definition &Concept; Personality: Concept, Determinants and Personality Types (Type A and Type B, Big Five Model, MBTI Model); Learning: Concept and Theories (Classical and Operant Conditioning); Attitude: Components & Formation

Unit-3
Group Behaviour
Group Dynamics: Meaning, Formation and Types of Groups (Formal & Informal Groups), Stages of Group Development, Individual vs. Group decision making. Group vs Team. Types of Team.
Group Communication
Communication Types, Communication Process, Barriers to communication; Effective Communication Methods.

Unit-4
Motivation - Meaning, Nature &Importance. Motivational Theories (Maslow's Need Hierarchy Theory, Herzberg's two factor Theory, McClelland's Need Theory, Vroom's Expectancy Theory, Equity Theory); Motivational Challenges.
Leadership - Leadership: Nature and Importance; Leadership Styles; Leadership Theories (Trait Theory, Behaviour Theory, Contingency Theory)

Textbooks:
1. Organisational Behaviour: L.M. Prasad
2. Organisational Behaviour: Rao & Narayana
3. Organizational Behaviour: Gupta and Joshi (KP)

Reference books:
1. Organisational Behaviour: K Aswathappa (HPH)
2. Organisational Behaviour: Stephen Robbins (PHI)

GE-1 Practical/Tutorial: Oraganizational Behaviour

1. Organisation's adaptability towards artificial intelligence.
2. Leadership Challenges and transformation using AI.

3. Social media and group behaviour.
4. People analytics in organisational behaviour.
5. Technology enabled work practices in organisations.
6. Converging technologies and employee perception.
7. Industry 4.0
8. Case Study Need Hierarchical theory in Team building.
9. Expectancy Theory towards Technological Adaptation
10. Practice of Telecommuting and remote working in IT / ITES.
11. Team building Exercises.
12. Personality Types.

DSM (HONOURS)

SEMESTER I

Ability Enhancement Compulsory Course

AEC-1: ENVIRONMENTAL SCIENCE (4 Credits)

Full marks -100 (Mid-Sem: 20; End-Sem: 80)

Unit-1

The Environment: The Atmosphere, Hydrosphere, Lithosphere, Biosphere, Ecology, Ecosystem, Biogeochemical Cycle (Carbon Cycle, Nitrogen Cycle), Environment Pollution: Air Pollution, Water Pollution, Soil Pollution, Radiation Pollution.

Unit-2

Population Ecology: Individuals, Species, Pollution, Community, Control Methods of Population, Urbanization and its effects on Society, Communicable Diseases and its Transmission, Non-Communicable Diseases.

Unit-3

Environmental Movements in India: Grassroot Environmental movements in India, Role of women, Environmental Movements in Odisha, State Pollution Control Board, Central Pollution Control Board.

Unit-4

Natural Resources: Conservation of Natural Resources, Management and Conservation of Wildlife, Soil Erosion and Conservation, Environmental Laws: Water Act, 1974, Air Act, 1981, The Wildlife (Protection) Act, 1972, Environment Protection, 1986, Natural Disasters and their Management.

Suggested Readings:

Carson, R. 2002. *Silent Spring*. Houghton Mifflin Harcourt.

Gadgil, M., & Guha, R. 1993. *This Fissured Land: An Ecological History of India*. Univ. of California Press.

Gleeson, B. and Low, N. (eds.) 1999. *Global Ethics and Environment*, London, Routledge.

Gleick, P. H. 1993. *Water in Crisis*. Pacific Institute for Studies in Dev., Environment & Security. Stockholm Env. Institute, Oxford Univ. Press.

Groom, Martha J., Gary K. Meffe, and Carl Ronald Carroll. *Principles of Conservation Biology*. Sunderland: Sinauer Associates, 2006.

Grumbine, R. Edward, and Pandit, M.K. 2013. Threats from India's Himalaya dams. *Science*, 339: 36-37.

McCully, P. 1996. *Rivers no more: the environmental effects of dams* (pp. 29-64). Zed Books.

McNeill, John R. 2000. Something New Under the Sun: An Environmental History of the Twentieth Century.

Odum, E.P., Odum, H.T. & Andrews, J. 1971. *Fundamentals of Ecology*. Philadelphia: Saunders.

Pepper, I.L., Gerba, C.P. & Brusseau, M.L. 2011. Environmental and Pollution Science. Academic Press.

DSM (HONOURS)
SEMESTER II
Core Course
C-3: DATA STRUCTURE (6 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1

Introduction: Basic Terminology, Data structure, Time and space complexity, Review of Array, Structures, Pointers.
Linked Lists: Dynamic memory allocation, representation, Linked list insertion and deletion, Searching, Traversing in a list, Doubly linked list, Sparse matrices.

Unit-2
Stack: Definition, Representation, Stack operations, Applications (Infix–Prefix–Postfix Conversion & Evaluation, Recursion).
Queues: Definition, Representation, Types of queue, Queue operations, Applications.

Unit-3

Trees: Tree Terminologies, General Tree, Binary Tree, Representations, Traversing, BST, Operations on BST, Heap tree, AVL Search Trees, M-way search tree, Applications of all trees.

Unit-4

Sorting: Exchange sorts, Selection Sort, Bubble sort, Insertion Sorts, Merge Sort, Quick Sort, Radix Sort, Heap sort.
Searching: Linear search, Binary search.

Text Books:

1. Classic Data Structure ,P.Samanta , PHI , 2/ed.

Reference Books:
1. Ellis Horowitz, Sartaj Sahni, “Fundamentals of Data Structures”, Galgotia Publications, 2000.
2. Sastry C.V., Nayak R, Ch. Rajaramesh, Data Structure & Algorithms, I.K.International Publishing House Pvt.Ltd, New Delhi.

Practical/Tutorial: Data Structure Lab
Write a C/ C++ Program for the followings:
1. To insert and delete elements from appropriate position in an array.
2. To search an element and print the total time of occurrence in the array.
3. To delete all occurrence of an element in an array.
4. Array implementation of Stack.

5. Array implementation of Linear Queue.
6. Array implementation of Circular Queue.
7. To implement linear linked list and perform different operation such as node insert and delete, search of an item, reverse the list.
8. To implement circular linked list and perform different operation such as node insert and delete.
9. To implement double linked list and perform different operation such as node insert and delete.
10. Linked list implementation of Stack.
11. Linked list implementation of Queue.
12. Polynomial representation using linked list.
13. To implement a Binary Search Tree.
14. To represent a Sparse Matrix.
15. To perform binary search operation.
16. To perform Bubble sort.
17. To perform Selection sort.
18. To perform Insertion sort.
19. To perform Quick sort.
20. To perform Merge sort.

DSM (HONOURS)
SEMESTER II
Core Course
C-4: INTRODUCTION TO PROBABILITY and STATISTICS (6 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1

Introduction to Statistics: Data Collections and Descriptive Statistics, Inferential Statistics and Probability Models, Populations and Samples.

Descriptive Statistics: Describing and Summarizing Data Sets, Chebyshev's Inequality, Normal and Paired Data Sets, Sample Correlation Coefficient Problems.

Elements of Probability: Sample Space and Events, Venn Diagrams and Algebra of Events, Axioms of Probability, Sample Spaces Having Equally Likely Outcomes, Conditional Probability, Bayes' Formula, Independent Events.

Unit-2

Random Variables and Expectations: Random Variables, Types of Random Variables, Jointly distributed Random Variables, Expectation, Properties of the expected value, Variance, Covariance and Variance of Sums of Random Variables, Moment Generating Functions. Chebyshev's Inequality and the Weak Law of Large Numbers Problems.

Special Random Variables: Bernoulli and Binomial Distribution Function, Poisson Random Variables, Hypergeometric Random Variable, Uniform Random Variable, Normal Random Variables, Exponential Random Variables, Poisson Process, Gamma Distribution, Chi-Square Distribution, t-Distribution, f-Distribution, Logistics Distribution.

Unit-3

Distributions of Sampling Statistics: The Sample Mean, Central Limit Theorem, Sample Variance, Sampling Distributions from a Normal Population, Sampling from a Finite Population.
Parameter Estimation: Maximum Likelihood Estimators, Internal Estimates, Estimating the difference in Means of Two Normal Populations, Approximate Confidence Interval for the Mean of a Bernoulli Random Variable,

Unit-4

Hypothesis Testing: Significance levels, Tests Concerning the Mean of a Normal Population, Testing the Equality of Means of Two Normal Populations, Hypothesis Tests in Bernoulli Populations, Testing the Relationship Between Two Poisson Parameters.

Regression: Introduction, Least Squares Estimators of the Regression Parameters, The Coefficient of Determination and the Sample Correlation Coefficient, Analysis of Residual, Transforming to Linearity, Weighted Least Squares, Polynomial Regression, Multiple Linear Regression, Predicting Future Responses, Logistic Regression Models for Binary Output Data.

Text Books:

1. Introduction to Probability and Statistics for Engineers and Scientists by Sheldon M. Ross, Elsevier. Chapter 1-9.
2. Introduction to Probability and Statistics Using R by G. Jay Kerns,

Practical/Tutorial: Introduction to Probability and Statistics Lab: Using R programming

In the Lab students will learn R from the book by G. Jay Kerns and then conduct the following six (6) experiments. The detail experiments given in the Lab.

Lab 1: Space Vacation Histogram experiment using R function called SpaceVacation.r
Lab 2: Testing of Chebyshev's Theorem using R.
Lab 3: Using R to simulate experiments.
Lab 4: Using R to create Binomial Distributions.
Lab 5: Understanding of Normal Distributions and compare theoretical distribution to some real data
Lab 6: Using R, empirically study how Central Limit Theorem works.

Laboratory Project 1

Reminder for Projects: Every week or so, you will be required to conduct a statistical experiment on the computer in the lab and, most importantly, write-up a short report of the results at home. The report should answer ALL the questions asked and will almost always include some graphics created by **R** in the computer lab.

Tips for doing Projects: Some simple hints for doing a good job on the Lab Projects.

- READ the entire assignment through, including the Thought Questions before doing anything.
- THINK about the *'essay'* questions while doing the computer work.
- Use a word processor to write-up the results. Neat results and presentation will win you respect and praise from your professors!

1 The Story:

The year is 2020 and interplanetary vacation travel is all the rage. For your upcoming July vacation, you and your friend Bob have looked through the travel brochures and decided upon two possible planets to visit; either PlanetX or PlanetY.

"I don't like it too hot", says Bob as you try to decide, "In fact, I always get very cranky and unreasonable if the temperature is much above 80 degrees."

You look through the brochures and find out that the average temperature of both PlanetX and PlanetY in July is a comfortable 63 degrees. Even better, the very sophisticated brochure for PlanetY indicates that the `median` temperature there in July is a cool 55 degrees.

"I like PlanetY", says Bob rather emphatically. "We should go there."

You, on the other hand, have taken a statistics class and know a bit about these things. To avoid making a big mistake and vacationing with a hot, cranky and unreasonable Bob, you decide to download data containing daily July temperature readings for the two planets. You are going to make an INFORMED CHOICE based on statistics.

Your task is using ideas you've learned about descriptive statistics, make a clear and organized presentation to Bob about why you should NOT go to PlanetY on vacation. Remember, besides being a bit heat sensitive, Bob knows nothing about statistics so you will have to explain things clearly.

2 The Laboratory:

To create 600 samples of temperature data for PlanetX and PlanetY, you will need to access an R function called `SpaceVacation()` in Prof. Champanerkar's directory. Type:

```
> source(file=url("http://www.math.csi.cuny.edu/~abhijit/113/data/SpaceVacation.r"))
```

To create the data and store it in a data frame called `Temps`,

```
> Temps = SpaceVacation()
```

Note, every time you execute the `SpaceVacation` function, you are actually *creating* new data. Your data will be slightly different than anyone else's data.

To get at the data, `attach` it and see how it is named:

```
> attach(Temps)
> names(Temps)
> length(TempX)  # Should return '600' for 600 pieces of data
```

3 The Histograms:

Now that you have the data, do some analysis and organization.

1. Create and print histogams for temperatures on both planets. Make sure that your name is in the title of the histogram.
2. Compare the `range` of July temperatures on the two planets. What is the hottest day you can expect on either planet? What is the coldest July day ever recorded on each? Write these answers on the histograms.
3. A check to make sure the brochures weren't lying. What is the mean and median July temperature on each planet? Write the answers on the histograms.

4 Questions:

Answer the following questions on a separate sheet of paper. Write a sentence or two for each question. To answer some of the questions, you may use R to make your answer more precise.

1. Using your histograms, how would you estimate the probability (chance) that the temperature on a given day will be *less* than 50 degrees? Find this probability for PlanetX and PlanetY.
2. Again using the histograms, how would you determine the probability (chance) that the temperature on a given day will be *greater* than 80 degrees? Find this probability for PlanetX and PlanetY.
3. Explain the following statement: "On an 'average day', the temperature on PlanetX is about 64 degrees. On PlanetY, the temperature on an 'average day' is probably NOT the average temperature."

5 Final Report

Write a final report to Bob. Your report should be at least half a page long and explain why you should NOT go to PlanetY on vacation. Remember again, besides being a bit heat sensitive, Bob knows nothing about statistics so you will have to explain things clearly.

6 To hand in:

1. 2 histograms (1 for each planet) with the answers to the questions listed in (3)
2. 1 page with the answers to the questions listed in (4).
3. Your final report.

Laboratory Project 2

Tips for doing Projects: Some simple hints for doing a good job on the Lab Projects.

- READ the entire assignment through, including the Thought Questions before doing anything.
- THINK about the *'essay'* questions while doing the computer work.
- The report should answer ALL the questions asked and will almost always include some graphics created by **R** in the computer lab.
- Use a word processor to write-up the results. Neat results and presentation will win you respect and praise from your professors!

1 Chebyshev's Theorem - Part I:

First lets test Chebyshev's Theorem on some real data, the length of time of eruptions of the Old Faithful Geyser in Yellowstone. In R, load the data:

```
> data(faithful)
> help(faithful)  %% Shows info on the data
> attach(faithful)  %% Loads the 'names' in faithful into R
> names(faithful)
```

Lets look at the eruption times now in variable `eruptions`.

```
> length(eruptions)     %% How much data
> hist(eruptions,20)
```

To test Chebyshev's theorem we need to know (1) the mean of the data and (2) the standard deviation. Use R to find these and store them in variables named, for example, `emean` and `esd`. Print out the histogram (with your name on it). On the histogram, mark the mean and ± 1 standard deviation on the axis.

One way to see what percentage of the data is within one, two or three standard deviations is to compute the Z-score of the data. Do this using R. Hint: z = (x - mean(x))/sd(x). Make a new variable of the zscores of the eruption data and plot a histogram of this. Print this out (with your name on it) and hand it in later.

Estimate on the Z-Score histogram the percentage of data within:

- ± 1 standard deviation of the mean
- ± 2 standard deviations of the mean
- ± 3 standard deviations of the mean

To actually calculate the above numbers, use the R command `sum`. For example, if the standard deviation of eruptions is stored in `esd` and the mean in `emean`, then to count the number of data points in eruptions within ± 1 standard deviation of the mean,

```
> sum(eruptions>emean-esd & eruptions<emean+esd)
```

The percentage is easy to get, just divide by the total number of data points. Use this procedure to actually calculate the percentages you previously estimated.

QUESTIONS:

1. According to Chebyshev's theorem, what percentage of measurements are expected to fall in the interval $\overline{x} \pm 1.25s$?
2. How many measurements actually fall in this range?
3. According to Chebyshev's theorem, what percentage of measurements are expected to fall in the interval $\overline{x} \pm 1.75s$?
4. How many measurements actually fall in this range?
5. According to Chebyshev's theorem, what percentage of measurements are expected to fall in the interval $\overline{x} \pm 1.00s$?
6. How many measurements actually fall in this range?
7. How does this compare with the Empirical Rule?
8. Compare the actual results with Chebyshev's prediction. Comment.

2 Chebyshev's Theorem - Part II:

Chose another data set from R's library of data. (To list all data sets, type `data()`). Follow the same procedure as above to test Chebyshev's theorem and the Empirical Rule.

- Load the data, find its length and make a histogram
- On the histogram, pencil in the mean and ± 1 standard deviation.
- Convert the data to Z-SCORES and make a histogram.
- From the Z-SCORE histogram, estimate the percentage of data within one, one and a half, one and three-quarters, two and three standard deviations from the mean.
- Calculate these percentages using `sum` and compare the results to Chebyshev's Theorem and the Empirical Rule.

Introduction to Probability and Statistics

Laboratory Project 3

Part I: Using R to SIMULATE Experiments

Although Einstein said that God does not play dice, R can play dice (and cards) rather easily. In this Lab assignment we will learn how to make R roll die and select cards. We can then have R quickly and conveniently play games MANY, MANY times. The output of the games can be analyzed statistically.

The main R command we will use is `sample`. For example, suppose we wanted to roll one die 10 times. Since the output of any die roll is a number between 1 and 6, we could try:

```
> sample(1:6,10,replace=T)   %Select a number from 1-6, ten times w/replacement
[1] 2 1 5 2 1 6 4 3 3 6
```

or we could write a function:

```
> Roll1Die = function(n) sample(1:6,n,rep=T)
> Roll1Die(20)
 [1] 2 3 3 5 4 2 1 3 6 1 4 6 2 1 3 2 1 5 1 4
```

Now we can 'Roll1Die' as many times as we want.

Suppose we want to flip 3 coins and count the number of heads that appear. How would you tell R to flip a coin 15 times? Try

```
> sample(0:1,15,rep=T) #Pick either 0=Tails or 1=Heads)
 [1] 1 1 0 1 0 1 1 1 0 0 0 1 1 1 0
```

or, perhaps more useful, write a function:

```
> Flip1Coin = function(n) sample(0:1,n,rep=T)
> Flip1Coin(30)
 [1] 1 1 1 1 0 0 0 1 0 0 0 0 1 0 0 1 1 0 1 0 0 0 1 0 1 1 1 1 1 1
```

To be answered on a separate sheet and handed in:

1. How many Heads appeared in these 30 experiments? (use R's `sum` command)
2. Is this number what you would expect? Why?
3. Try flipping 1,000 coins. How many heads occurred? How does this relate to the 'Empirical Definition' of probability?

Part II: Probability Distributions and Histograms

We know that a Probability Distribution (or Probability Function) assigns a probability to each value of the Random Variable. Now we want to investigate how well a Theoretical Probability Function fits with actual data. For example, suppose our experiment consists of rolling two die and adding the results. We can figure out the probability of getting any of the possible outcomes: P(2) = 1/36, P(3) = 2/36, ..., P(7) = 6/36, P(8) = 5/36 ...

We can also easily *simulate* this experiment using R. One way to get the results from one hundred experiments would be

```
> TwoDie = Roll1Die(100) + Roll1Die(100)
```

To look at the results, try a histogram

```
> hist(TwoDie,breaks=c(1.5:12.5)) #Centers intervals on 1,2,....12
```

To be answered on a separate sheet and handed in:

1. What percentage of the 100 experiments resulted in 7?
2. What percentage of the 100 experiments resulted in 4?
3. How does this compare with the Probability Function?

We can easily make an Empirical Probability Function from the data, all we have to do is tell R to plot the relative frequency instead of the frequency on the histogram.

```
> hist(TwoDie,breaks=c(1.5:12.5),prob=T) #Plot Relative Frequency
```

Do the following and hand in the answers:

1. Use R to conduct the experiment 50 TIMES. (ie Roll two dice fifty times)
2. Plot and print the resulting Relative Histogram.
3. On the histogram, sketch the theoretical Probability Function, P(X).
4. How well do the two compare?
5. Repeat the process using 500 experiments. Comment.
6. Repeat the process using 5000 experiments. Comment.

Part III: Mean of a Random Variable

We know that the Probability Function, $P(X)$ can be used to calculate the *mean* of a Random Variable. For Discrete Probability Distributions, the mean of a random variable X is written as

$$\mu = \sum X_i P(X_i) \quad .$$

For example, compute the mean (or expected value) in the Two Dice experiment

$$\mu = 2 \cdot 1/36 + 3 \cdot 2/36 + \ldots + 12 \cdot 1/36 = 7$$

To Do: Find the experimental (sample) mean for the Two Dice experiment using R. For example, to roll two dice and add up the results 100 times:

```
> TwoDie = Roll1Die(100) + Roll1Die(100)
```

The experimental mean value is simply:

```
> mean(TwoDie)
[1] 7.32
```

Your answer might be slightly different, but should be close to 7.
Your tasks - hand in the answers:

1. Try this 10 times - Does the experimental mean change? Write down the numbers you obtained.

2. Compare the mean value as computed above to the mean of the simulated data when you take 50, 500, 5000 *realizations* of the experiment in R. Comment on what happens as the number of experiments increases.

3. Pick one value of n, say $n = 50$ and run 50 experiments 10 or 20 times. Record the value of the sample mean each time. Plot a histogram of these mean values. What is the mean of the sample means? Explain this odd statement:

   "The sample mean is a Random Variable."

Part IV: A coin problem (hand in all answers)

1. Four fair coins are tossed simultaneously and the number of HEADS is recorded each time.

   (a) Use your knowledge of Probability to construct the Probability function for this experiment. (What is the Random Variable? What is its Range?) Make a chart showing X and $P(X)$.

   (b) Use R to simulate conducting this experiment 50 times.

   (c) Compare your theoretical $P(X)$ to the relative frequency histogram of the simulated experiments.

   (d) Use R to simulate 1,000 four-coin experiments. Again, compare the relative frequency histogram you get to $P(X)$.

   (e) Calculate the theoretical mean, μ. Compare this to the sample mean when you find when simulating 50, 100 and 1,000 realizations of the experiment.

The Binomial Distribution

Laboratory Project 4

Part I: Using R to create Binomial Distributions

To steal Prof. Verzani's words: "Binomial random numbers are *discrete* random numbers. They have the distribution of the number of successes in n independent Bernoulli trials where a Bernoulli Trial results in either a Success or a Failure, with probability of Success given by p."

R can easily produce binomial random numbers. We can then *simulate* various experiments easily on the computer. For example, lets consider a True/False test with 8 questions. If a student simply guesses at each question, the number of correct answers on the test will be a Binomial random number. $n = 8$ is the number of Bernoulli trials, $p = 1/2$ is the probability of getting a correct answer for this (dumb!) student. To simulate 5 such (dumb) students taking the test, use the R command `rbinom(number,n,p)`

```
> rbinom(5,8,0.5)
[1] 3 4 4 2 5
```

The results show that the first student got 3 correct answers, the second two got 4 correct answers, the fourth (unlucky) student got only 2 correct answers and the fifth (lucky) student got 5 of the 8 questions correct ... she actually PASSED the test!

Of course, its absolutely no big deal to simulate 10,000 students taking this test.

```
> testdat = rbinom(10000,8,0.5)
```

Try this and look at the relative frequency histogram of `testdat`. Compare this to the Theoretical Binomial Probability Density. (Remember, the formula for the probability of x successes in n trials is

$$P(x) = p^x q^{n-x}{}_nC_x$$

For example, in this 8 question multiple choice test with straight guessing,

$$P(1) = (1/2)^1(1/2)^{8-1}{}_8C_1 = (1/2)^8 \cdot 8 = 8/256 = 0.03125$$

To Do:

1. True/False Test:
   (a) Redo the above simulation changing the value of p.
   (b) Plot and print the resulting Relative Histogram for Binomial test data for smart students (i.e. $p > 0.5$) and for really dumb students (i.e. $p < 0.5$) who are more likely to guess wrong than to guess correctly.
   (c) Explain in words how the shape of the Distribution changes with changing values of p.

(d) For these values of p (including the original $p = 1/2$ experiment), compute the mean value of the simulated test scores and compare this to the Theoretical mean value for a binomial distribution.

2. Multiple Choice Test: Now consider a multiple choice exam with 20 questions and 5 possible choices for each question.

   (a) Use R's `rbinom` function to simulate the number of correct answers for 500 students taking this test if every student simply guesses at each question.

   (b) For your sample of 500 students, how many passed the test? (Passing is getting at least 13 correct answers.)

   (c) Plot a relative frequency histogram of the simulated data. Compare the simulated value of P(10) to the Theoretical Value of P(10).

   (d) Find the mean of your 500 student sample. Compare this to the Theoretical Mean for the binomial distribution. Are they close?

   (e) If a Professor gives the test to a class of 100 students and finds that 75% of the students passed, explain, using statistics and probability, why the Professor can conclude that the student's were not simply guessing at the answers.

3. ACME Light Bulbs Inc.: A light bulb manufacturer claims that their advanced production facility only produces one defective light bulb in every batch of 100 light bulbs made. ACME ships light bulbs in boxes of 20.

   (a) Use R to simulate the Probability Distribution of the number of GOOD light bulbs in a box of 20 light bulbs. (What is n? What is p? Have R simulate 1,000 boxes of 20.)

   (b) Use the binomial distribution to find the average number of BAD light bulbs in a box. Compare this to the results of the simulation.

   (c) What are the chances of getting a box of ACME light bulbs containing 1 BAD bulb? 4 BAD bulbs? 10 BAD bulbs?

   (d) Suppose the new production manager turns out to be completely unqualified and the probability of producing a bad bulb rises to 10%. Explain in words and pictures what will happen to the shape of the Distribution Function. Now what are the chances of getting 1 BAD bulb in a box?

Laboratory Project 5

How *Normal* is a Distribution?

A large part of statistical analysis is based on the properties of normally distributed random variables. There's a famous (infamous?) LAW that states that data coming from a large number of independent experiments will produce a Normal Distribution and as such, a lot of statistical models assume 'Normality'.

In this exercise, we will take what we know about normal distributions and compare the theoretical distribution to some real data.

Begin an R session. First, lets clear out any old junk that's lying around in the workspace. (Warning, this command will delete any work you have done!)

```
> rm(list=ls())   ## Removes (rm) all variables (good for saving space)
```

Next, lets load a data set containing information on the air quality in New York.

```
> data(airquality)
> attach(airquality)   ## To allow us to access the named variables by name.
```

The data set contains measurements of

```
> names(airquality)
[1] "Ozone"   "Solar.R" "Wind"    "Temp"    "Month"   "Day"
```

Ozone is the primary ingredient in 'smog'; the more ozone in the air, the worse the air quality. Too much ozone is dangerous to one's health and poses real hazards for the elderly and those with lung ailments. A 'bad air' day is one with high ozone concentrations.

Preliminary Question: Take a look at histograms of the three main measurements, Ozone, Wind and Temp. From the histograms, which could best be described by a normal distribution? Why?

Lets concentrate on the Wind measurements. Since we want to compare these measurements to the Standard Normal Distribution, the first thing we'd like to do is normalize the data. This is easy, all we want is the *z-scores* of the Wind data. In R, define a new measurement:

```
> zwind = (Wind - mean(Wind))/sd(Wind)
> hist(zwind, prob=T)
```

Now we can compare the distribution of *zwind* to the Standard Normal Distribution. First, we know that 68% of Normally Distributed data lies between ± 1 standard deviation of the mean. To see if this is true for the Wind data, we need to compute the percentage of data in *zwind* that lies between ± 1. In R:

```
> sum( zwind > -1 & zwind < 1) ## Number of data points in 1 sd of mean
[1] 100
> sum( zwind > -1 & zwind < 1)/length(zwind)  ## Percentage of data in 1 sd
[1] 0.6535948
```

Pretty close. Using the table of areas for the Standard Normal Distribution, we find that 86.64 % of Normally Distributed data lies within ± 1.5 standard deviations of the mean. Check this for the Wind data:

```
> sum(zwind > -1.5 & zwind < 1.5)/length(zwind)
[1] 0.869281
```

Again, the Normal Distribution prediction is quite close. Is this true for other values of zwind? Is this true for the Temperature and the Ozone Distributions as well? Answers the questions below and hand in your answers.

Questions:

1. Compare the Wind data distribution to the Standard Normal curve for 5 values of z. i.e. Compare the percentages in the Table in the book (for the standard normal distribution) to the percentages found using R.

2. Compute the following Probabilities using (a) the Wind Data and (b) the Normal Distribution.

   (a) Prob(Wind $>$ 10 mph)

   (b) Prob(Wind $>$ 15 mph)

   (c) Prob(Wind $>$ 20 mph)

   (d) Prob(Wind $<$ 5 mph)

3. Rescale the Ozone and Temperature data to get zscores. Re-do the comparisons in Question 1 above for these distributions. Which of the three is the 'most normal'? Why? Give numbers to support your conclusion. Which of the three is 'least' normal.

4. A (lazy) statistician decides to build a model for Ozone concentration in New York based on a normal distribution. Assume ozone concentration is normally distributed with mean and standard deviation given by the data in Ozone. What does this Normal model predict for the chances of finding an Ozone Concentration (1) Greater than 70, (2) Greater than 110 ? How do these predictions compare to the actual data? Explain how and why a normal model might underpredict the chances of 'bad air' days.

Introduction to Probability and Statistics

Laboratory Project 6

Investigating The Central Limit Theorem

Key to understanding Inferential Statistics is the most popular statistical LAW known as the *Central Limit Thoerem*. In a nutshell, this powerful theorem states three facts about the statistics of *sample means*.

Given random samples of size n selected from some population with mean $= \mu$ and standard deviation σ, the following relationships hold:

- The mean of the population and the mean of the samples are EQUAL.

$$\mu_{\overline{x}} = \mu$$

- The standard deviation of the population (σ) and the standard deviation of the sample means ($\sigma_{\overline{x}}$) are related by the formula:

$$\sigma_{\overline{x}} = \frac{\sigma}{\sqrt{n}}$$

- No matter what the distribution of the random variable x is, the distribution of the sample means is NORMAL.

In this exercise, we will use R to take a look at these 3 facts and see, empirically, how the Central Limit Theorem works.

First start up an R session and lets clear out any old junk that might be lying about in the workspace. (Warning, this command will delete any work you have done!! Use with care!)

```
> rm(list=ls())   ## Removes (rm) all variables (good for saving space)
```

Next, lets create a bunch of data. Let's consider a game like Yahtzee! where you roll 6 dice. We want to keep track of the number of ONES that appear in each roll. Can you figure out what the probability distribution of this random variable (X = number of ONES in six rolls) is? Is it binomial? If so, what are p, q and n?

It is a binomial distribution. To get R to roll 6 dice 10,000 times and count the number of ONES in each roll and store it in variable x, try

```
> x = rbinom(10000,6,1/6)
> hist(x,breaks=c(-0.5:6.5),prob=T)
```

Look at the distribution, you've produced. It's definitely not normal. What should the mean and standard deviation be? Check them using R. Are they close to the theoretical values?

Now we want to try sampling the population data. Suppose we want to take samples of size 40. We can do this in R with the `sample` command. Lets store one sample of size 40 in `xsamp` and look at the sample distribution.

```
> xsamp = sample(x,40)
> hist(xsamp,breaks=c(-0.5:6.5),prob=T)
```

How does this sample compare with the population? What is the sample mean?

We can take a number of different samples reusing the `sample` command, looking at the mean each time.

```
> xsamp = sample(x,40)
> mean(xsamp)
[1] 0.975
> xsamp = sample(x,40)
> mean(xsamp)
[1] 1.025
> xsamp = sample(x,40)
> mean(xsamp)
[1] 1.225
```

Notice the **sample mean** is a random variable. It is different for each sample. The Central Limit Theorem is concerned with the statistics of this sample mean.

Suppose we want to look at the mean value of 100 different samples of 40. We can easily create this random variable (lets call it `sampmean`) in R, using a loop. Try this:

```
> sampmean = numeric(0)  # make a place to store the sample means
> for (i in 1:100) sampmean[i] = mean(sample(x,40)) #find mean of 100 samples of 40
```

Now lets investigate the three parts of the Central Limit Theorem. First, the mean of x should equal the mean of the sample means. Check it.

```
> mean(x)
[1] 1.0126
> mean(sampmean)
[1] 1.0175
```

Not perfect but pretty darn close.

Second, the standard deviation of the sample means and the population are related by $\sigma_{\overline{x}} = \sigma/\sqrt(n)$. In R, we compare

```
> sd(sampmean)   # $\sigma{\overline{x}}$
[1] 0.1375462
```

to the population standard deviation devided by $\sqrt(n)$

```
> sd(x)/sqrt(40)
[1] 0.1445272
```

Ok, not perfect, but still pretty close.

The last thing the Central Limit Theorem says is that, no matter what the distribution of x, the sample means should be Normally Distributed. Is this the case here? Look at

```
> hist(sampmean)
```

Is it 'normal looking'?

TO DO:

1. Redo the above analysis for samples of size 50, 100 and 500. Comment on the following:

   (a) How do the histograms of `sampmean` change as the sample size is increased? Does the standard deviation increase or decrease? Is the sample mean looking 'normal'?

   (b) How do the first two predictions of the central limit theorem compare to the actual data as the sample size is increased? Does $\mu_{\overline{x}}$ approach μ? How about the second part of the Central Limit Theorem?

2. Redo the analyis for a different population distribution with sample sizes 50, 100 and 500. You may want to create data using a different binomial distribution or you may try out the R commands `rexp(10000,.1)` (exponential, long-tails) or `rpois(1000,4)` (Poisson Distribution, non-normal). Whatever you chose as the population, examine what happens to various sized sample means. Check each part of the Central Limit Theorem.

```
SpaceVacation <- function(n=100)
     {T1 = 90 + rnorm(2*n)*rexp(2*n,0.5)
     T2 = 50 + rnorm(4*n,0,8)
     TempY = c(T1,T2)

     TempX =rnorm(6*n,63.5,7)

     Vacation = data.frame(TempX,TempY)

     return(Vacation)}
```

DSM (HONOURS)
SEMESTER II
Generic Elective Course
GE-2: FINANCIAL MANAGEMENT (6 Credits)
Full marks -100 (Mid-Sem: 20; End-Sem: 80)

Unit-1
Introduction & Basic Concepts: Important functions of Financial Management, Objectives of the firm: Profit maximization vs. Value maximization, Role of Chief Financial Officer. Financial environment in which a firm has to operate, Concepts of Annuity and Perpetuity, Risk-return relationship (concepts only)

Unit-2
Basic Theoretical Framework: The financial system and its technology; The factors affecting the stability of the financial system; Development finance vs. universal banking; Financial intermediaries and Financial Innovation; RBI-Central Banking.

Unit-3
Financial Institutions: A brief historical perspective. An update on the performance of IDBI, ICICI, IFCI and SFCs, LIC &GIC, Banking Institutions: Commercial banks - the public and the private sectors - structure and comparative performance, problems of competition; interest rates, spreads, and NPAs. Bank capital - adequacy norms and capital market support.
Non-banking financial institutions: Evolution, control by RBI and SEBI. A perspective on future role, Unit Trust of India and Mutual Funds, Reserve bank of India Framework for/Regulation of Bank Credit .

Unit-4
Sources of Finance and Cost of Capital/ Financing Decisions: Different sources of finance; long term and short term sources, Cost of capital: concept, relevance of cost of capital, Implicit and Explicit cost, specific costs and weighted average cost, rationale of after tax weighted average cost of capital, marginal cost of capital. (Concepts only).

Text Books:

1. Fundamentals of Financial Management, Van Horne,Pearson
2. Essentials of Financial Management, IM Pandey, Vikas
3. Financial Management, Khan & Jain, McGraw Hill,
4. Financial Management, Srivastav & Misra, Oxford.

Reference Books:

1. Financial Management, G Sudarsan Reddy, HPH
2. Financial Management – Tulsian (S Chand)
3. Fundamentals of Financial Management, Brigham, Cengage
4. Financial Management by PrasannaChandra , Tata McGraw Hill
5. Financial Management, Rustogi, Galgotia Publishing.

DSM (HONOURS)
SEMESTER II
General Elective Course
GE-2: GENERAL MANAGEMENT (6 Credits)
Theory Full marks: 100 (Mid-Sem: 20; End-Sem: 80)

Unit-1

Nature of Management: Meaning, Definition, it's nature purpose, importance & Functions, Management as Art, Science & Profession- Management as social System Concepts of management-Administration-Organization.

Evolution of Management Thought: Contribution of F.W.Taylor, Henri Fayol ,Elton Mayo, Chester Barhard& Peter Drucker to the management thought. Various approaches to management (i.e. Schools of management thought)Indian Management Thought.

Unit-2

Functions of Management (Part-I)

Planning - Meaning - Need & Importance, types levels– advantages & limitations, Forecasting - Need & Techniques, Decision making - Types - Process of rational decision making & techniques of decision making, **Organizing** - Elements of organizing & processes: Types of organizations, Delegation of authority - Need, difficulties in delegation – Decentralization,

Unit-3

Functions of Management (Part-II)

Staffing - Meaning & Importance, Direction - Nature – Principles, Communication - Types & Importance, Motivation - Importance – theories, Leadership - Meaning - styles, qualities & functions of leaders **Controlling**-Need, Nature, importance, Process & Techniques, Coordination - Need, Importance.

Unit-4

Strategic Management
Definition, Classes of Decisions, Levels of Decision, Strategy, Role of different Strategist, Relevance of Strategic Management and its Benefits, Strategic Management in India.

Text Books:

1. Horold Koontz and IteinzWeibrich, Essential of Management, McGraw Hills International
2. K.Aswathapa, Essential of Business Administration, Himalaya Publishing House

Reference Books:

1. L.M.Parasad Principles & practice of management - Sultan Chand & Sons - New Delhi
2. Tripathi, Reddy, Principles of Management, Tata McGraw Hill

DSM (HONOURS)

SEMESTER II

Ability Enhancement Compulsory Course

AEC-2: MIL (English/Odia/Hindi) (4 Credits)

Full marks -100 (Mid-Sem: 20; End-Sem: 80)

English

Unit 1: Short Story

(i) Jim Corbett – The Fight between Leopards
(ii) Dash Benhur – The Bicycle
(iii) Dinanath Pathy – George V High School
(iv) Alexander Baron – The Man Who knew too much
(v) Will F Jenkins – Uneasy Homecoming

Unit 2: Prose

(i) Mahatma Gandhi – The way to Equal Distribution
(ii) S Radhakrishnan – A Call to Youth
(iii) C. V. Raman – Water- The Elixir of Life
(iv) Harold Nicolson – An Educated Person
(v) Claire Needell Hollander – No Learning Without Feeling

Unit 3:

(i) Comprehension of a passage and answering the questions

Unit 4:

(ii) Language exercises-test of vocabulary and grammar

Text Books:

All Stories and Prose pieces

Reference Books:

(i) The Widening Arc: A Selection of Prose and Stories, Ed. A R Parhi, S Deepika, P Jani,Kitab Bhavan, Bh ubaneswar.
(ii) A Communicative Grammar of English, Geoffrey Leech.
(iii) A University Grammar of English, Randolph Quirk and Sidney Greenbaum
(iv) Developing Reading Skills. F. Grellet. Cambridge: Cambridge University Press, 1981

Odia

ସବିଶେଷ ପାଠ୍ୟ

ଯୋଗାଯୋଗମୂଳକ ମାତୃଭାଷା – ଓଡ଼ିଆ **(AECC)**

ପାଠ୍ୟ-୧ / **Course – 1**: ଯୋଗାଯୋଗ ଅନୁବିଧି, ରୀତି ଓ ମାଧ୍ୟମ

୧ମ ଏକକ : ଯୋଗାଯୋଗର ପରିଭାଷା, ଅନୁବିଧି, ପରିସର ଓ ପ୍ରକାରଭେଦ

୨ୟ ଏକକ : ସାକ୍ଷାତକାର, ଭାଷଣ କଳା

୩ୟ ଏକକ : ସମ୍ବାଦର ପରିଭାଷା, ପରିସର ଓ ସମ୍ବାଦ ପ୍ରସ୍ତୁତି

୪ର୍ଥ ଏକକ : ଓଡ଼ିଆ ଭାଷାର ବର୍ଣ୍ଣମାଳା, ବର୍ଣ୍ଣାଶୁଦ୍ଧିର ନିରାକରଣ । (ବନାନ ତ୍ରୁଟି - ସାଦୃଶ୍ୟଜନିତ ଅଶୁଦ୍ଧି, ଲିଙ୍ଗଗତ ଅଶୁଦ୍ଧି, ସନ୍ଧିଗତ ଅଶୁଦ୍ଧି, ସମାସଗତ ଅଶୁଦ୍ଧି, ବଚନ ଓ ବିଭକ୍ତିଗତ ଅଶୁଦ୍ଧି, ବାକ୍ୟ ବିଧିଜନିତ ଅଶୁଦ୍ଧି, ସମାର୍ଥବୋଧକ ଶବ୍ଦାଶୁଦ୍ଧି, ପ୍ରତ୍ୟୟ ଜନିତ ଅଶୁଦ୍ଧି, ଶବ୍ଦ ସଂଯୋଗାତ୍ମକ ଓ ସ୍ୱରସଙ୍ଗତି ଜନିତ ଅଶୁଦ୍ଧି)

ସହାୟକ ଗ୍ରନ୍ଥସୂଚୀ (ପାଠ୍ୟ-୧ / **Course – 1**)

୧. ଯୋଗାଯୋଗ ମୂଳକ ମାତୃଭାଷା (ଓଡ଼ିଆ) ସାମଲ ବିରଞ୍ଚି ନାରାୟଣ, ସତ୍ୟନାରାୟଣ ବୁକ ଷ୍ଟୋର, କଟକ ।

୨. ସଂଯୋଗ ଅନୁବିଧି, ସନ୍ତୋଷ କୁମାର ଦିପାଠୀ, ନାଳନ୍ଦା, କଟକ

୩. ଭାଷଣ କଳା ଓ ଅନ୍ୟାନ୍ୟ ପ୍ରସଙ୍ଗ - କୃଷ୍ଣଚନ୍ଦ୍ର ପ୍ରଧାନ, ସତ୍ୟନାରାୟଣ ବୁକ୍ ଷ୍ଟୋର,କଟକ

୪. ପ୍ରାୟୋଗିକ ଓଡ଼ିଆ ଭାଷା – ଓଡ଼ିଶା ରାଜ୍ୟପାଠ୍ୟ ପୁସ୍ତକ ପ୍ରଣୟନ ଓ ପ୍ରକାଶନ ସଂସ୍ଥା, ଭୁବନେଶ୍ୱର

୧. ସମ୍ବାଦ ଓ ସାମ୍ବାଦିକତା – ଚନ୍ଦ୍ରଶେଖର ମହାପାତ୍ର, ଓଡ଼ିଶା ରାଜ୍ୟ ପାଠ୍ୟପୁସ୍ତକ ପ୍ରଣୟନ ଓ ପ୍ରକାଶନ ସଂସ୍ଥା, ଭୁବନେଶ୍ୱର

୧. ନିର୍ଭୁଲ ଲେଖାର ମୂଳସୂତ୍ର, ନୀଳାଦ୍ରିଭୂଷଣ ହରିଚନ୍ଦନ, ପି.ସି.ଆର ପବ୍ଲିକେସନ, ଭୁବନେଶ୍ୱର

୨. ସର୍ବସାର ବ୍ୟାକରଣ - ନାରାୟଣ ମହାପାତ୍ର ଓ ଶ୍ରୀଧର ଦାସ, ନିଉ ଷ୍ଟୁଡେଣ୍ଟସ୍ ଷ୍ଟୋର, କଟକ

Hindi

AECC : HINDI (MIL)

UNIT - I
कविता

(i) कबीर - साखी : 1 से 10
(ii) तुलसी - विनयपत्रिका - पद 1 और 2
(iii) प्रसाद - मधुमय देश
(iv) निराला - भिक्षुक
(v) अज्ञेय - हिरोशिमा

UNIT - II
गद्य

(i) रामचन्द्र शुक्ल - उत्साह
(ii) हजारी प्रसाद द्विवेदी - कुटज
(iii) हरिशंकर परसाई - सदाचार का तावीज

UNIT - III
शब्द ज्ञान

(i) शब्द शुद्धि
(ii) वाक्य शुद्धि
(iii) पर्यायवाची शब्द
(iv) विलोम शब्द

UNIT - IV
सामान्य ज्ञान

(i) निबंध लेखन (Essay Writing)

पाठ्य पुस्तक :

1. हिन्दी प्रसून - सं. डॉ. अंजुमन आरा, प्लानेट वी, कटक

DSM (HONOURS)
SEMESTER III
Core Course
C-5: COMPUTER ORGANIZATION (6 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1

Basic Structure of Computers: Computer Types, Functional Units, Input Unit, Memory Unit, Arithmetic and Logic Unit, Output Unit, Control Unit, Basic Operational Concepts, Bus Structures, Software. Machine Instructions and Programs: Numbers, Arithmetic Operations, and Characters: Number Representation, Addition of Positive Numbers, Addition and Subtraction of Signed Numbers, Overflow of Integer Arithmetic, Floating-Point Numbers & Operations, Characters, Memory Locations and Addresses, Byte Addressability, Word Alignment, Accessing Numbers, Characters, and Character Strings, Memory Operations, Instructions and Instruction Sequencing, Register Transfer Notation, Basic Instruction Types, Instruction Execution and Straight-Line Sequencing, Branching, Condition Codes, Generating Memory Addresses, Addressing Modes, Implementation of Variables and Constants, Indirection and Pointers, Indexing and Arrays, Relative Addressing.

Unit-2

Basic Processing Unit: Register Transfers, Performance on Arithmetic or Logic Operation, fetching a Word from Memory, Storing a Word in Memory. Execution of a Complete Instruction, Branch Instruction, Multiple Bus Organization Hardwired Control, A Complete Processor. Micro-programmed Control: Microinstructions, Microprogram Sequencing, Wide- Branch Addressing, Microinstructions with Next-Address Field, Prefetching Microinstructions, Emulation.

UNIT-3

Input/ Output Organization: Accessing I/O Devices, Interrupts, Interrupt Hardware, Enabling & Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions. Direct Memory Access, Bus Arbitration, Buses, Synchronous Bus, Asynchronous Bus, Interface Circuits: Parallel Port, Serial Port, Standard I/O Interfaces, Peripheral Component Interconnect (PCI) Bus, SCSI Bus, Universal Serial Bus (USB)

Unit-4

Pipelining: Role of Cache Memory, Pipeline Performance, Data Hazards: Operand Forwarding, Handling Data Hazards in Software, Side Effects. Instruction Hazards: Unconditional Branches, Conditional Branches and Branch Prediction. Influence on Instruction Sets: Addressing Modes, Condition Codes, Data path and Control Considerations. Superscalar Operation: Out-of-Order Execution, Execution Completion, Dispatch Operation, RISC & CISC Processors.

Text Books

1. Carl Hamacher, Z. Vranesic, S. Zaky: Computer Organization, 5/Ed (TMH)

Reference Books

1. William Stallings: Computer Organization and Architecture (Design for Performance), 9/Ed
2. S. Brown, & Z. Vranesic, “Fundamentals of Digital Logic Design with VHDL”, 2/Ed, McGraw-Hill.

Practical/Tutorial: Computer Organization Lab

1. Study of the complete Architecture of 8085 Microprocessor along with its instruction set.
2. Introduction to GNU Simulator 8085, with its features.
3. Write an Assembly Language Program to add N consecutive numbers.
4. Write an Assembly Language Program to find the smallest and largest number from a given series.
5. Write an Assembly Language Program for subtraction of two 8-bit numbers.
6. Write an Assembly Language Program for displaying a Rolling message “Hello 123”.
7. Write an Assembly Language Program to perform ASCII to Decimal conversion.
8. Write an Assembly Language Program to add two unsigned binary numbers.
9. Write an Assembly Language Program to subtraction of two unsigned binary numbers.

Demonstrate the followings:

10. Assembling and Dis-assembling of computer.
11. Trouble shooting in Computer.

DSM (HONOURS)

SEMESTER III

Core Course

C-6: OPERATING SYSTEMS (6 Credits)

Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)

Practical Full marks: 25 (End semester evaluation)

Unit–1

Introduction to Operating System, System Structures: Operating system services, system calls, system programs, Operating system design and implementation, Operating system structure.

Unit–2

Process Management: Process Concept, Operations on processes, Process scheduling and algorithms, Inter-process Communication, Concepts on Thread and Process, Deadlocks: Deadlock detection, deadlock prevention, and deadlock avoidance fundamentals.

Unit-3

Memory Management Strategies: Swapping, Contiguous Memory Allocation, Paging, Segmentation, Virtual Memory Management: Concepts, implementation (Demand Paging), Page Replacement, Thrashing.

Unit–4

Storage Management: File System concept, Access Methods, File System Mounting, File Sharing and File Protection, Implementing File Systems, Kernel I/O Systems.

Text book:

1. Operating System Concepts, Abraham Silberschatz, Peter B. Galvin, and Greg Gagne, Eighth Edition, Wiley Student Edition 2009.

Reference book:

1. Modern Operating System, Tanenbaum,Pearson, 4/Ed. 2014
2. Richard F Ashley, Linux with Operating System Concepts, Chapman and Hall/CRC Published August 26, 2014
3. Richard Blum, Linux Command Line and Shell Scripting Bible, O' Reilly

Practical/Tutorial: Operating System Lab

1. Write a program (using *fork()* and/or *exec()* commands) where parent and child execute:

    a) same program, same code.

    b) same program, different code.

c) before terminating, the parent waits for the child to finish its task.

2. Write a program to report behavior of Linux kernel including kernel version, CPU type and model. (CPU information)
3. Write a program to report behavior of Linux kernel including information on configured memory, amount of free and used memory. (memory information)
4. Write a program to print file details including owner access permissions, file access ime,
5.
6. file name is given as argument.
7. Write a program to copy files using system calls.
8. Write a programusing C to implement FCFS scheduling algorithm.
9. Write a program using C to implement Round Robin scheduling algorithm.
10. Write a program using C to implement SJF scheduling algorithm.
11. Write a program using C to implement non-preemptive priority based scheduling algorithm.
12. Write a program using C to implement preemptive priority based scheduling algorithm.
13. Write a program using C to implement SRTF scheduling algorithm.
14. Write a program using C to implement first-fit, best-fit and worst-fit allocation Strategies.

DSM (HONOURS)
SEMESTER III
Core Course
C-7: DATABASE MANAGEMENT SYSTEMS (6 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1

Introduction to Database and Database Users, Database System Concepts and Architecture: data Models, schema, and instances, Conceptual Modeling and Database Design: Entity Relationship (ER) Model: Entity Types, Entity Sets, Attributes, Keys, Relationship Types, Relationship Sets, Roles and Structural Constraints, Weak Entity Types, ER Naming Conventions. Enhanced Entity-Relationship (EER) Model.

Unit-2

Database Design Theory and Normalization: Functional Dependencies, Normal Forms based on Primary Keys, Second and third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

Unit-3

Relational data Model and SQL: Relational Model Concepts, Basic SQLs, SQL Data Definition and Data types, Constraints in SQL, Retrieval Queries in SQL, INSERT, DELETE, UPDATE Statements in SQL, Relational Algebra and Relational Calculus: Unary Relational Operations: SELECT and PROJECT, Binary Relation: JOIN and DIVISION.
No SQL: Introduction to NoSQL Database, Types and examples of NoSQL Database Systems, Comparative study of SQL and NoSQL.

Unit-4

Introduction to Transaction ProcessingConcepts and Theory: Introduction to Transaction Processing, Transaction and System Concepts, Properties of Transactions, Recoverability, Serializability, Concurrency Control Techniques, Locking techniques for Concurrency Control, Concurrency Control based on Time-Stamp Ordering.

Text Book:

1. Fundamentals of Database Systems, 6th edition, RamezElmasri, ShamkantB.Navathe, Pearson Education

Reference Book:

1. An Introduction to Database System, Date C.J.- Pearson Education, New Delhi- 200

Practical/Tutorial: Database Systems Labs

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 Mangers 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, Department Name of all employees who have an ‘A’ in their name.
25. Query to display Name, Job, 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, Department 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 thanthe 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 Sale

DSM (HONOURS)

SEMESTER III

Generic Elective Course

GE-3: DECISION MANAGEMENT SYSTEMS (6 Credits)

Theory Full marks: 100 (Mid-Sem: 20; End-Sem: 80)

Unit–1

Introduction Artificial Intelligence, Digital Decisioning, Digital Decisioning principles.

Unit–2

Discover and Model Decisions - Characteristics of Suitable Decisions - A Decision Taxonomy - Finding Decisions - Documenting Decisions - Prioritizing Decisions. Design and Implement Decision Services - Build Decision Services - Integrate Decision Services - Best Practices for Decision Services Construction.

Unit–3

Monitor and Improve Decisions - What Is Decision Analysis? - Monitor Decisions - Determine the Appropriate Response - Develop New Decision-Making Approaches - Confirm the Impact Is as Expected - Deploy the Change.

Unit–4

Enablers for Decision Management Systems: People Enablers, Process Enablers, Technology Enablers.

Text books:

1. James Taylor, “Decision Management Systems-A Practical guide to using Business rules and Predictive Analytics”, IBM Press, 2012.
2. James Tayler, ``Digital decisioning: Using Decisioning Management to deliver Business Impact on IA, Meghan-Kiffer Press, 2019.

DSM (HONOURS)
SEMESTER III
Skill Enhancement Course
SEC-1: COMMUNICATIVE ENGLISH (4 Credits)
Full marks -100 (Mid-Sem: 20; End-Sem: 80)

Unit-1: Introduction

(i) What is communication?
(ii) Types of communication (Horizontal, Vertical, Interpersonal, Grapevine),
(iii) Uses of Communication, Inter-cultural communication, Communication today: (iv)Distinct features of Indianisation, alternative texts of language learning, global English
and English in the print and electronic media in India.

Unit-2: The Four Skills and Prospect of new material in language learning

(i) Listening-Passive and active, Speaking effective, intelligibility and clarity
(ii) Methods and techniques of reading such as skimming, scanning and searching for information; Reading to understand the literal, metaphorical and suggested meaning of a passage,
(iii)Identifying the tone (admiring, accusatory, ironical, sympathetic, evasive, indecisive, ambiguous, neutral) of the writer and view-points.
(iv)Cohesive and Coherent writing

Unit-3: Grammatical and Composition Skills

(i) Doing exercises like filling in the blanks, correcting errors, choosing correct forms out of alternative choices, joining clauses, rewriting sentences as directed, and replacing indicated sections with single words / opposites / synonyms, choosing to use correct punctuation marks, getting to understand and use formal and informal styles, learning to understand the usages of officialese, sexism, racism, jargon.
(ii) Learning to understand information structure of the sentence such as topic-focus relationship; strategies of thematization, postponement, emphasis, structural compression (deletion of redundant parts, nominalization, cleft and pseudo-cleft sentences, elliptical structures etc.), Logical Connectors between sentences, Methods of developing a paragraph, structure of an essay and methods of developing an essay

Unit-4: Exercises in Written Communication

(i) Précis writing
(ii) Note-taking skills
(iii) Writing reports
(iv)Guidelines and essentials of official correspondence for making enquiries, complaints and replies

(v) Making representations; writing letters of application for jobs; writing CV, writing letters to the editor and social appeals in the form of letters/pamphlets.

Text Books:

1. State Model Syllabus for Under Graduate Couse in Skill Enhancement Course (I), pdf file is available in the internet: http://dheodisha.gov.in/Higher-Education/Listmodule-syllabus.aspx

Reference Books:

Ways of Reading: Advanced reading Skills for Students of English Literature. Martin Montgomery et al. London: Routledge, 2007.

Applying Communication Theory for Professional Life: A Practical Introduction. Dainton and Zelley, http://tsime.uz.ac.zw/claroline/backends/download.php?url=L0ludHJvX3RvX2NvbW11bmljYXRpb25fVGhlb3J5LnBkZg%3D%3D&cidReset=true&cidReq=MBA563

Literature and the art of Communication, Cambridge University Press.

Vistas and Visions. Orient Black Swan (writing and grammar exercises at the end of lessons are recommended) From *Remapping An Anthology for Degree Classes*, ('Writing Skills'), Orient Black Swan.

*Indian English through Newspapers (*Chapter 4,5 and 6), Concept, New Delhi,2008.

Contemporary Communicative English, S Chand

Technical Communication: A Reader Centred Approach. P.V. Anderson. Wadsworth, Cengage.

DSM (HONOURS)

SEMESTER IV

Core Course

C-8: FOUNDATION OF DATA SCIENCE & ANALYTICS (6 Credits)

Theory Full marks: 100 (Mid-Sem: 20; End-Sem: 80)

.

Unit–1

Definition of Big Data, Big data characteristics & considerations, Data Repositories – analyst perspective, Business drivers for analytics, Typical analytical architecture, Business Intelligence Vs Data Science, Drivers of Big Data analytics, Role of data scientist in Big data ecosystem, Application of Big data analytics.

Unit-2

Need of Data analytic lifecycle, Key roles for successful analytic project, various phases of Data analytic lifecycle: Discovery, Data Preparation, Model Planning, Model Building, Communicating Results, Operationalization.

Introduction to R: GUI of R, Getting data into & out of R, Data types in R, Basic operations, Descriptive Statistics.

Unit–3

Overview of Clustering, K- means, Association Rules, Apriori Algorithm, Linear Regression, Logistic Regression.

Unit–4

Naïve Bayesian Classifier, Decision Tress, Time Series analysis, Text Analysis.

Text book:

1. David Dietrich, Barry Hiller, “Data Science & Big Data Analytics”, EMC education services, Wiley publications, 2012

2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning", Springer, Second Edition, 2011.

DSM (HONOURS)
SEMESTER IV
Core Course
C-9: JAVA PROGRAMMING (6 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1

Introduction to Java: Java History, Architecture and Features, Understanding thesemantic and syntax differences between C++ and Java, Compiling andExecuting a Java Program, Variables, Constants, Keywords (super, this, final, abstract, static, extends, implements, interface) , Data Types, Wrapper class,Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, DoingBasic Program Output, Decision Making Constructs (conditional statements andloops) and Nesting, Java Methods (Defining, Scope, Passing and ReturningArguments, Type Conversion and Type and Checking, Built-in Java ClassMethods). Input through keyboard using Commandline Argument, the Scanner class, BufferedReader class.

Unit-2

Object-Oriented Programming Overview: Principles of Object-Oriented Programming, Defining & Using Classes, Class Variables & Methods, Objects, Object reference, Objects as parameters, final classes, Garbage Collection.

Constructor- types of constructor, this keyword, super keyword. Method overloading and Constructor overloading. Aggregation vs Inheritance, Inheritance: extends vs implements, types of Inheritance, Interface, Up-Casting, Down-Casting, Auto-Boxing, Enumerations,Polymorphism, Method Overriding and restrictions. Package: Pre-defined packages and Custom packages.

Unit-3

Arrays: Creating & Using Arrays(1D, 2D, 3D and Jagged Array), Array of Object, Referencing Arrays Dynamically. Strings and I/O: Java Strings: The Java Stringclass, Creating & Using String Objects, Manipulating Strings, String Immutability& Equality, Passing Strings To & From Methods, StringBuffer Classes and StringBuilder Classes. IO package: Understanding StreamsFile class and its methods, Creating, Reading,Writing using classes: Byte and Character streams,FileOutputStream, FileInputStream, FileWriter, FileReader, InputStreamReader, PrintStream, PrintWriter. Compressing and Uncompressing File.

Unit-4

Exception Handling, Threading, Networking and Database Connectivity:Exception types, uncaught exceptions, throw, built-in exceptions, Creating yourown exceptions; Multi- threading: The Thread class and Runnable interface,creating single and multiple threads, Thread prioritization, synchronization andcommunication, suspending/resuming threads. Using java.net package,Overview of TCP/IP and Datagram programming. Accessing and manipulatingdatabases using JDBC.

Text Books:

1. E. Balagurusamy, “Programming with Java”, TMH, 4/Ed,

Reference books:

1. Herbert Schildt, “The Complete Reference to Java”, TMH, 10/Ed.

Practical/Tutorial: Java Programming Lab

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 convert a decimal to binary number.
4. To check if a number is prime or not, by taking the number as input from the keyboard.
5. 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
6. Write a program that show working of different functions of String and StringBufferclasss like setCharAt( ), setLength( ), append( ), insert( ), concat( )and equals( ).
7. Write a program to create a – "distance" class with methods where distance is computed interms of feet and inches, how to create objects of a class and to see the use of this pointer
8. Modify the – "distance" class by creating constructor for assigning values (feetandinches)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.
9. 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)
10. 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.
11. 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 userand 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 Fibonacci 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 invokes Arithmetic 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 different mouse handling events like mouseClicked( ), mouseEntered( ), mouseExited( ), mousePressed( ),

mouseReleased( ) &mouseDragged().

23. Write a program to demonstrate different keyboard handling events.

ITM (HONOURS)
SEMESTER IV
Core Course
C-10: DATA COMMUNICATIONS & NETWORKING
(Theory: 4 Credits; Practical: 2 Credits)
Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit-1

Introduction to Data Communications and Network Models: Protocols and Standards, Layers in OSI Models, Analog and Digital Signals, Transmission Modes, Transmission Impairment, Data Rate Limits, Performance, Digital Transmission, Network Devices & Drivers: Router, Modem, Repeater, Hub, Switch, Bridge (fundamental concepts only).

Unit-2

Signal Conversion: Digital-to-Digital Conversion, Analog-to-Digital Conversion, Digital-to-analog Conversion, Analog-to-analog Conversion.
Transmission Media: Guided Media, Unguided Media, Switching Techniques: Packet Switching, Circuit Switching, Datagram Networks, Virtual-Circuit Networks, and Structure of a Switch.

Unit-3

Error Detection and Correction: Checksum, CRC, Data Link Control: Framing, Flow and Error Control, Noiseless Channels, Noisy channels, (Stop and Wait ARQ, Slidding Window Protocol , Go Back N, Selective Repeat) HDLC, Point-to-Point Protocol. Access Control: TDM,CSMA/CD, and Channelization (FDMA, TDMA, and CDMA).

Unit-4

Network Layer: Logical Addressing, IPv4 Addresses, IPv6 Addresses, Virtual-Circuit

Networks: Frame Relay and ATM, Transport Layer: Process-Process Delivery: UDP, TCP.

Application layers: DNS, SMTP, POP, FTP, HTTP, Basics of WiFi (Fundamental concepts only), Network Security: Authentication, Basics of Public Key and Private Key, Digital Signatures and Certificates (Fundamental concepts only).

Text Books:

1. Data Communications and Networking, Fourth Edition by Behrouza A. Forouzan,TMH

Reference Books:

Computer Networks, A.S.Tanenbaum, 4th edition, Pearson Education.

C-14: Practical/Tutorial Computer Networks Lab

Use C/C++/ any Network Simulator

1. Simulate Even Parity generator and checker.
2. Simulate two dimensional Parity generator and checker.
3. Simulate checksum generator and checker.
4. Simulate Hamming code method.
5. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
6. Simulate and implement stop and wait protocol for noisy channel.
7. Simulate and implement go back n sliding window protocol.
8. Simulate and implement selective repeat sliding window protocol.
9. Simulate and implement distance vector routing algorithm.

DSM (HONOURS)
SEMESTER IV
Generic Elective Course
GE-4: MARKETING ANALYTICS (6 Credits)
Theory Full marks: 100 (Mid-Sem: 20; End-Sem: 80)

Unit–1

Marketing Data Summarization: Slicing and Dicing Marketing Data with PivotTables – Using Excel charts to Summarize Marketing Data – Using Excel Functions to Summarize Marketing Data.

Unit–2

Forecasting Techniques: Simple Linear Regression and Correlation – Using Multiple Regression to Forecast Sales – Forecasting in the Presence of Special Events – Modeling Trend and Seasonality – Ratio to Moving Average Forecasting Method – Winter's Method – Using Neural Networks to Forecast Sales.

Unit-3

Customer Needs: Conjoint Analysis – Logistic Regression – Discrete Choice Analysis – Customer Value – Introduction to Customer Value, Benefits.

Marketing Segmentation: Cluster Analysis, User-Based Collaborating Filtering – Collaborative Filtering – Using Classification Trees for Segmentation.

Unit–4

Retailing and Market Research Tools: Retailing – Introduction to retailing. Market Basket Analysis – Marketing Research Tools – Principal Components Analysis.

Text book:

1. Marketing Analytics: Data driven Techniques with MS-Excel, by Wayne L. Winston, Wiley, 2014
2. Marketing Analytics: Strategic models and metrics, by Stephan Sorger, CreateSpace Independent Publishing Platform, 2013.

ITM (HONOURS)
SEMESTER IV
Skill Enhancement Course
SEC-2: QUANTITATIVE APTITUDE and LOGICAL REASONING (4 Credits)
Full marks -100 (Mid-Sem: 20; End-Sem: 80)

Unit-1

Whole Numbers, Integers, Rational and irrational numbers, Fractions, Square roots and Cube roots, Surd and Indices, Problems with numbers, Divisibility.
Different formulae of Percentage, Profit and loss, Discount, Simple interest, Ratio and Proportion, Mixture, Mixture
Time and work, Pipes and Cisterns, Basic concepts of Time, Distance and Speed: relationship among them.

Unit-2

Concept of Angles, Different Polygons like triangles, rectangular, square, right angled triangle, Pythagorean Theorem, Perimeter and Area of Triangles, Rectangles, Circles
Raw and Grouped Data, Bar Graphs, Pie Chart, Mean, Median, Event and Sample Space, Probability.

Unit-3

Analogy basing on kinds of relationships, Simple Analogy: Pattern and Series of Numbers, Letters, Figures. Coding-Decoding of Numbers, Letters, Symbols (Figures), Blood relations.

Unit-4

Logical statement: Two premise argument, More than two premise argument using connectivity.
Venn Diagram, Mirror Images, Problems on Cubes and Dices.

Text Books:

1. State Model Syllabus for Under Graduate Couse in Skill Enhancement Course (II), pdf file is available in the internet: http://dheodisha.gov.in/Higher-Education/Listmodule-syllabus.aspx

DSM (HONOURS)

SEMESTER V

Core Course

C-11: BIG DATA ANALYTICS (6 Credits)

Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)

Practical Full marks: 25 (End semester evaluation)

Unit–1

Linear Methods for Regression and Classification: Overview of supervised learning, Linear regression models and least squares, Multiple regression, Multiple outputs, Subset selection, Ridge regression, Lasso regression, Linear Discriminant Analysis, Logistic regression, Perceptron learning algorithm.

Unit–2

Model Assessment and Selection: Bias, Variance, and model complexity, Bias-variance trade off, Optimism of the training error rate, Estimate of In-sample prediction error, Effective number of parameters, Bayesian approach and BIC, Cross- validation, Boot strap methods, conditional or expected test error.

Additive Models, Trees, and Boosting: Generalized additive models**,** Regression and classification trees, Boosting methods-exponential loss and AdaBoost, Numerical Optimization via gradient boosting, Examples (Spam data, California housing, New Zealand fish, Demographic data).

Unit-3

Neural Networks (NN) , Support Vector Machines (SVM), and K-nearest Neighbour: Fitting neural networks, Back propagation, Issues in training NN, SVM for classification, Reproducing Kernels, SVM for regression, K-nearest –Neighbour classifiers(Image Scene Classification)

Unsupervised Learning and Random forests: Association rules, Cluster analysis, Principal Components, Random forests and analysis.

Unit–4

Inferential Statistics and Prescriptive Analytics: Assessing Performance of a classification Algorithm (t-test, McNemar's test, Paired t-test, paired f-test), Analysis of Variance, Creating data for analytics through designed experiments.

Practical: Implementation of following methods using R

Simple and multiple linear regression, Logistic regression, Linear discreminant analysis, Ridge regression, Cross-validation and boot strap, Fitting classification and regression trees, K-nearest neighbours, Principal component analysis, K-means clustering.

Text books:

1.Trevor Hastie, Robert Tibshirani, Jerome Friedman , *The Elements of Statistical Learning-Data Mining, Inference, and Prediction* ,Second Edition , Springer Verlag, 2009.
[chapters: 2,3(3.1-3.4,3.6),4(4.3-4.5),7(excluding 7.8 and 7.9),9(9.1,9.2),(10.1-10.5,10.8,10.10,10.14),11(11.3-11.6),12(12.1-12.3),13.3,14(14.1-14.3.8,14.5.1),15]

2. E.Alpaydin*, Introduction to Machine Learning*, Prentice Hall Of India,2010,(Chapter-19)

3. G.James, D.Witten, T.Hastie, R.Tibshirani-*An introduction to statistical learning with applications in R*, Springer,2013.(2.3,3.6.1-3.6.3,4.6.1-4.6.3,5.3,6.6.1,8.3.1,8.3.2,10.4,10.5.1)

References:

1.C.M.Bishop –Pattern Recognition and Machine Learning,Springer,2006
2. L.Wasserman-All of statistics

DSM (HONOURS)

SEMESTER V

Core Course

C-12: CLOUD COMPUTING (6 Credits)

Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)

Practical Full marks: 25 (End semester evaluation)

Unit–1

Cloud computing definition, Private, public and hybrid cloud, Types of cloud services: IaaS, PaaS, SaaS, Benefits and challenges of cloud computing, Evolution of cloud computing, Usage scenarios and applications, Business models around cloud, Major players in cloud computing, Issues in cloud, Eucalyptus, Nimbus, Open Nebula, CloudSim.

Unit–2

Software as a Service, Platform as a Service, Infrastructure as a Service, Database as a Service, Monitoring as a Service, Communication as a Service, Service providers: Google App Engine, Amazon EC2, Microsoft Azure, Sales force, Introduction to MapReduce, GFS, HDFS, Hadoop Framework.

Unit–4

Collaborating on Calendars, Schedules and Task Management, Collaborating Event Management, Contact Management, Project Management, Collaborating on Word Processing, Database: Storing and Sharing Files, Collaborating via Web-based communication tools, Evaluating Web mail Service, Collaborating via Social Networks, Collaborating via Blogs and Wikis.

Unit–4

Need for Virtualization, Pro and Cons of Virtualization, Types of Virtualization, System VM, Process VM, Virtual Machine Monitor, Virtual Machine Properties, Interpretation and binary Translation, HLL VM, Hypervisors, Xen, KVM, VMWare, Virtual Box, Hyper – V.

Cloud security challenges, Software as a Service Security, Common Standards, The Open Cloud Consortium, The Distributed Management Task Force, Standards for application developers, Standards for Messaging, Standards for Security, End user access to cloud computing, Mobile Internet device and the cloud.

Practical: Cloud Computing

1. Create virtual machines that access different programs on same platform.

2. Create virtual machines that access different programs on different platforms.
3. Exploring Google cloud for the following
   a) Storage
   b) Sharing of data
   c) Manage your calendar, to-do lists
   d) A document editing tool
4. Exploring Microsoft cloud
5. Exploring Amazon cloud
6. Exploring Open cloud

Text books:

1. Cloud Computing for Dummies, by J. Hurwitz , R. Bloor, M. Kanfman, and F. Haiper, Wiley India Edition, 2010 (Unit I).
2. Cloud Computing Implementation Management and Security by J. Rittinghouse and J. Ransome, CRC Press, 2010 (Unit II).
3. Cloud Computing: A Practical Approach by A. Velte, T. Velte and R. Elsenpeter, Tata McGraw Hill, 2009 (Unit II).
4. Cloud Computing: Web-based Applications That Change the Way You work and Collaborate Online by M.. Miller, Que Publishing, August 2008 (Unit III).
5. Virtual Machines by J. E. Smith and R. Nair, Morgan Kaufmann Publishers, 2006 (Unit IV).
6. http://cloud-standards.org/wiki/index.php?title=Main_Page (Unit –V).

Reference Books:

1. Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS), by M. Kavis, Wiley, 2014.
2. Mastering In Cloud Computing by R. Buyya, C. Vecchiola and T. Selvi, Tata Mcgraw-Hill Education, 2013.
3. Cloud Computing: SaaS, PaaS, IaaS, Virtualization, Business Models, Mobile, Security and more by K. Jamsa, Jones & Bartlett Learning Company LLC, 2013.

DSM (HONOURS)

SEMESTER V

Discipline Specific Elective Course

DSE-1: SOFTWARE ENGINEERING (6 Credits)

Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)

Practical Full marks: 25 (End Semester Evaluation)

Unit–1

Introduction to Software Engineering, Software Crises, Overview of software development activities, Process Models, Classical Waterfall Model, Code and Fix model, Iterative waterfall model, Prototyping mode, Spiral model, RAD model, Agile models: Extreme Programming, and Scrum.

Requirement identification, Requirement Gathering and Analysis, Functional and Non-functional requirements, Trawling of Requirements, Software Requirement Specification (SRS), Requirement Traceability, Requirement Validation, Functional Specification.
Introduction to UML and UML Diagrams: Use Case Model, Class Diagram, Object Diagram, State Chart Diagram.

Unit–2

Software Project Management; Feasibility Study, Project Planning Activities, Estimation in Project Planning Process: LOC and Function Point, Project Estimation Techniques: COCOMO Model, Halstead's Software Science, Staff Estimation, Project Scheduling: Activity Networks & Critical Path Method, Gantt Chart, PERT Chart, Project Monitoring & Control, Risk Management: Software Risks, Risk Identification and Assessment, Risk Projection and Risk Refinement.

Unit-3

Overview of design process: High-level and detailed design, Cohesion and coupling, Modularity and layering, System Modeling: Traditional and Object-oriented Approaches, Traditional approach: Data Flow Diagram (DFD), Entity-Relationship Diagram (ERD), State Transition Diagram (STD) and Decision Table. Structural Analysis using Data Flow Diagrams: Structure Chart. Design Specification, Basic concepts of Object-Oriented Analysis and Design, Design Patterns.

Unit–4

V-model of software Development and Testing: Unit, Integration, System, Acceptance and Regression Testing. Differences between fault (bug, defect), error, and failure. Definition of software Testing: Verification and Validation, Black-box and While-box testing. Coding, Code Review: Walk-through, and Inspection, Defect Prevention, Debugging.

Introduction to Software Maintenance: Corrective, Adaptive, Perfective, and Preventive. Lehman's Laws for Closed Source Software and Open Source Software.

Text books:

Fundamentals of Software Engineering, Rajib Mall , PHI, 2014.

Reference Books:

Software Engineering, A Practitioner's Approach, Roger S. Pressman ,TMG Hill.

Software Engineering, I. Sommerville, 9th Ed. , Pearson Education. SOFTWARE

Practical/Tutorial: Software Engineering Lab

S. 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

forjailers, police officers and CBI officers.

2. **Route Information**: Online information about the bus routes and their frequency andfares
3. **Car Pooling**: To maintain a web based intranet application that enables the corporateemployees within an organization 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

ITM (HONOURS)

SEMESTER V

Discipline Specific Elective Course

DSE-2: PYTHON PROGRAMMING (6 Credits)

Full marks - 75 (Mid-Sem: 15; End-Sem: 60)

Practical Full marks: 25 (End semester evaluation)

Unit-1

Python: Features of Python , Installing Python for windows and setting up paths, writing and Executing of a python programs, Python Virtual machine, Frozen binaries, Comparison between C, Java and python , Comments , Docstrings ,How python sees variables, Data types in Python, built in types, sequences in python, sets, literals in Python, user defined data types, identifiers & reserved words, Naming convention in python,

Unit-2

various Operators in Python , Input & Output , Control statements, if statements, while loop, for loop, infinite loop, nested loop ,else suit, break, continue, pass ,assert, return statements, command line arguments.

Arrays in python, advantages using arrays, creating arrays, importing the array module, indexing and slicing on arrays, Processing the arrays, Comparing arrays.

Strings in Python, Creating strings, Length of a string, Indexing in strings, Slicing strings, Concatenation and Comparing strings,Finding SubStrings,Replacing a String.

Unit-3

Functions in Python , Define a function, Calling a function, return from function, pass by object Reference, Positional arguments, Default arguments, Recursive functions.

Introduction to OOP, features of OOP, creating classes, the self variable, constructor, types of variables, namespaces, types of methods.

Unit-4

Inheritance: Define inheritance, types of inheritance, constructors in inheritance, overriding super class constructors & methods, the super() method, MRO

Polymorphism: Duck typing philosophy of Python, operator overloading, method overriding, interfaces in python

Exceptions: Errors in a python program, Exceptions, Exception handling, Types of Exceptions, The Exception block, the assert statement, user defined exceptions

Python Database Connectivity: DBMS, types of databases used with Python, installation of MySQL database , setting path, verifying MySQL , installing MySQL connector, Working with MySQL database, Using MySQL from python, retrieving rows ,deleting rows, updating rows in a table.

Text Books

1. T. Budd, Exploring Python, TMH, 1st Ed, 2011.
2. Core Python Programming, Dr. R. Nageswar Rao , Dreamtech Press
3. Python Programming for Absolute Beginners, Michael Dawson, CENGAGE Learning

Reference Books

1. Allen Downey, Jeffrey Elkner, Chris Meyers , How to think like a computer scientist : learning with Python , Freely available online.2012

OnlineReferences:

1. Python Tutorial/Documentation www.python.or 2015

2. http://docs.python.org/3/tutorial/index.html

3. http://interactivepython.org/courselib/static/pythonds

4. http://www.ibiblio.org/g2swap/byteofpython/read/

Pratical: Software Lab based on Python Programming:

1. Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon users choice.

2. Write a Program 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:

Grade A: Percentage >=80

Grade B: Percentage>=70 and <80

Grade C: Percentage>=60 and <70

Grade D: Percentage>=40 and <60

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 parameters from user.

4. Write a Program to display the first n terms of Fibonacci series.

5. Write a Program to find factorial of the given number.

6. Write a Program to find sum of the following series for n terms: 1 – 2/2! + 3/3! - - - - - n/n!

7. Write a Program to calculate the sum and product of two compatible matrices.
8. Install MySQL and connector. Write Python programs to retrie, inserting, delete, update rows in a table.

DSM (HONOURS)
SEMESTER V
Skill Enhancement Course
SEC-3: R PROGRAMMING (4 Credits)
Full marks - 75 (Mid-Sem: 15; End-Sem: 60)
Practical Full marks: 25 (End semester evaluation)

Unit 1:
Introduction: Overview and History of R, Getting Help, Data Types, Subsetting, Vectorized, Operations.

Unit 2:
Reading and Writing Data. Control Structures, Functions, lapply, tapply, split, mapply, apply. Coding Standards.

Unit 3:
Scoping Rules. Debugging Tools, Simulation, R Profiler

Uni 4:

Data Analysis Using R: Visualization before Analysis, Dirty Data, Visualizing a Single Variables, Examining Multi Variable.

Statistical Methods for Evaluation: Hypothesis Testing, Difference of Means, Wilcoxon Rank-Sum Test, Type I and Type II Errors, Power and Sample Size, ANOVA.

Text Books:

1. William N. Venables and David M. Smith, An Introduction to R. 2nd Edition. Network Theory Limited. 2009.
2. Norman Matloff, The Art of R Programming -A Tour of Statistical Software Design, No Starch Press. 2011.
3. Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", McGraw Hill Publishers, ISBN 0-07-120413-X, 6th edition (chapter 3 only)

Practical: R Programming
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 the following sorting algorithms: Selection sort, Insertion sort, Bubble Sort
8. Implement linear search.
9. Implement binary search.
10. Implement matrices addition, subtraction and Multiplication.

11. Other programs to visualize single variables and multiple variables such as histogram, density plot, Dotchart, Barplot, Box-and-Whisker Plot, Hexbin Plot, Scatterplot Matrix.
12. Write R code for Student's *t*-test, Welch's *t*-test, Wilcoxon Rank-sum Test,

DSM (HONOURS)

SEMESTER VI

Core Course

C-13: MACHINE LEARNING (6 Credits)

Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)

Practical Full marks: 25 (End semester evaluation)

Unit–1

Introduction – Types of Machine Learning , Designing a Learning System, Issues in Machine Learning; -The Concept Learning Task - General-to-specific ordering of hypotheses, Find-S, List then eliminate algorithm, Candidate elimination algorithm, Inductive bias - Decision Tree Learning - Decision tree learning algorithm, Instance based Learning, Nearest neighbors method.

Unit–2

Artificial Neural Networks – Perceptrons, Learning rules, Gradient descent and the Delta rule, Adaline, Madaline Network, Multilayer networks, Derivation of Backpropagation rule- Backpropagation Algorithm- Convergence, Generalization; – Evaluating Hypotheses – Estimating Hypotheses Accuracy, Basics of sampling Theory, Radial basis function networks, Support Vector Machine.

Unit-3

Supervised Learning- Linear Regression (Gradient Descent, Normal Equations), Weighted Linear Regression (LWR), Logistic Regression, Generative Models (Gaussian Discriminant Analysis, Naive Bayes), Learning – Bayes theorem, Concept learning, Bayes Optimal Classifier, Naïve Bayes classifier, Bayesian belief networks, Tree Ensembles (Decision trees, Random Forests, Boosting and Gradient Boosting).

Unit–4

Unsupervised Learning- K-means, Gaussian Mixture Model (GMM), Expectation Maximization (EM), Variational Auto-encoder (VAE), Factor Analysis, Principal Components Analysis (PCA), Independent Components Analysis (ICA), Linear Discriminant Analysis (LDA), Vector Quantization – Self Organizing Feature Map.

Reinforcement learning: Markov decision process (MDP), Hidden Markov Model(HMM), Bellman equations, Value iteration and policy iteration, Linear quadratic regulation, Linear Quadratic Gaussian, Q-learning, Monte Carlo Methods.

Text books:

1. T. Mitchell, “Machine Learning” Mcgraw Hill Publisher.
2. T. Hastie, R. Tibshirani, J. Friedman “The Element of Statistical Learning” 2e 2008
3. E. Alpaydin, Introduction to Machine Learning. Eastern Economy Edition, Prentice Hall of India.
4. C. M. Bishop, Pattern recognition and Machine Learning, Springer

Lab Experiments:

1. Basic operations in Python implementation.
2. Loading data from Training set and testing the Models.
3. Learn to predict values with Linear Regression.
4. Learn to predict states using Logistic Regression.
5. Learn the definition of a Perceptron as a building block for neural networks, and the perceptron algorithm for classification.
6. Learn the definition of a Neural Network, Learn to train them using Backpropagation network.
7. Train Decision Trees to predict states and classification.
8. Learn the Bayes rule, and how to apply it to predicting data using the Naive Bayes algorithm.
9. Learn to train a Support Vector Machine to separate data linearly.
10. Use Kernel Methods in order to train SVMs on data that is not linearly separable.
11. Learn the basics of clustering Data, Cluster data with the K-means algorithm.
12. Cluster data with Gaussian Mixture Models.
13. Optimize Gaussian Mixture Models with Expectation Maximization.
14. Learn to scale features in your data, Learn to select the best features for training data.
15. Reduce the dimensionality of the data using Principal Component Analysis and Independent Component Analysis and LDA
16. Learn how to define Markov Decision Processes to solve real-world problems.
17. Learn about policies and value functions, Derive the Bellman Equations.
18. Write your own implementations of iterative policy evaluation, policy improvement, policy Iteration, and value Iteration.
19. Implement classic Monte Carlo prediction and control methods.
20. Learn how to tune hyper parameters of an estimator.
21. Plotting of Validation curve and learning curve to evaluate the model.
22. Evaluating Estimator performance, Cross validation

DSM (HONOURS)

SEMESTER VI

Core Course

C-14: INETERNET OF THINGS (6 Credits)

Theory Full marks: 75 (Mid-Sem: 15; End-Sem: 60)

Practical Full marks: 25 (End semester evaluation)

Unit–1

Introduction to Internet of Things, Definitions and Characteristics of IoT, Physical Design of IoT, Things in IoT, IoT Protocols, Logical Design of IoT, IoT Functional Blocks, IoT Communication Models, IoT Communication APIs, IoT Enabling Technologies - Wireless Sensor Networks, Cloud Computing, Big Data Analytics, Communication Protocols, Enbeedded Systems

Textbook 1:1.1 –1.5

Unit–2

IoTlevels and Development Templates, IoT Level-1, IoT Level-2, IoT Level-3, IoT Level-4, IoT Level-5, IoT Level-6.

IoT and M2M, Introduction, M2M, Difference between IoT and M2M, SDN and NFV for IoT - Software Defined Networking, and Network Function Virtualization, IoT Platform Design Methodology - Introduction, IoT Design Methodology, Step1: Purpose and requirement specification, Step2: Process Specification, Step 3: Domain Model Specification, Step 4: Information Model Specification, Step 5: Service Specification, Step 6: IoT Level Specification, Step 7: Function View Specification, Step 8: Operational View Specification, Step 9: Device and Component Integration, Step 10: Application Development, IoT System Logical Design Using Python

Textbooks 1:3.1-3.4, 5.1-5.4, 6.1-6.11

Unit-3

IoT Physical Devices and End Points: What is an IoT Device, Exemplary Device Raspberry Pi, About the Board, Linux on Raspberry Pi, Raspberry pi interfaces, programming raspberry pi with python, IoT physical servers and cloud offerings - introduction to cloud storage models and communication Networks, wamp-autobahn for IoT, xively cloud for IoT, python web application frame work-django, designing a RESTful web API

Textbook 1: 7.1-7.7, 8.1-8.7

Unit–4

Data Analytics for IoT; Introduction AppacheHadoop, using HadoopMapReduce for Batch Data Analysis,

Textbook 1: 10.1 -10.8

Ethics:Characterizing the IoT,Privacy, Control,Distributing Control and Crowd Sourcing, Environment,Physical Thing, Electronics,InternetService,Solutions,Internet of Things as Part of Solution, Cautious Optimizing, The Open IoT definition.

Textbook 2: Chapter 11

Text books:

1. Internet Of Things-A Hands on Approach, by Arshdeep Bahga and Vijay Madisetti, University of Penn, http://www.internet-of-things-book.com/
2. Designing the Internet of Things, by Adrian McEwen andHakimCassimally, Wiley Publication.

References:

1. Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems. By Ovidiu Vermesan and Peter Friess, River Publishers Series in Communication.

List of Experiments:

1. Define and Explain Eclipse IoT Project.
2. List and summarize few Eclipse IoT Projects.
3. Sketch the architecture of IoT Toolkit and explain each entity in brief.
4. Demonstrate a smart object API gateway service reference implementation in IoT toolkit.
5. Write and explain working of an HTTP- to-CoAP semantic mapping proxy in IoT toolkit.
6. Describe gateway-as-a-service deployment in IoT toolkit.
7. Explain application framework and embedded software agents for IoT toolkit.
8. Explain working of Raspberry Pi.
9. Connect Raspberry Pi with your existing system components.
10. Give overview of Zetta.

Design based Problems (DP)/Open Ended Problem:

1. How do you connect and display your Raspberry Pi on a Monitor Or TV?
2. Create any circuitry project using Arduino.

Major Equipment:

Raspberry pi, Arduino

List of Open Source Software/learning website:

- https://github.com/connectIOT/iottoolkit
- https://www.arduino.cc/
- http://www.zettajs.org/
- Contiki (Open source IoT operating system)
- Arduino (open source IoT project)
- IoT Toolkit (smart object API gateway service reference implementation)
- Zetta (Based on Node.js, Zetta can create IoT servers that link to various devices and sensors)

DSM (HONOURS)

SEMESTER VI

Discipline Specific Elective Course

DSE-3: WIRELESS COMMUNICATIONS (6 Credits)

Theory Full marks: 100 (Mid-Sem: 20; End-Sem: 80)

Unit–1

Introduction to Wireless Networks: Evolution of Wireless Networks, Applications, Challenges, Wireless Communication Principles and fundamentals: The Electromagnetic Spectrum, Transmission Bands and their Characteristics, Spectrum Regulation, Analog and Digital Data Transmission: Voice Coding, Multiplexing (Space Division Multiplexing, Frequency Division Multiplexing, Time Division Multiplexing, Code Division Multiplexing), Modulation Techniques for Wireless Systems: Analog Modulation (AM, FM), Digital Modulation(ASK,FSK,PSK,QAM)

Unit–2

Motivation for a specialized MAC: Hidden and exposed terminal, Near and far terminals. Multiple Access for Wireless Systems: FDMA, TDMA, CDMA, ALOHA-Carrier Sense Multiple Access (CSMA).

GSM: Mobile services, System architecture, Localization and calling, Handover, Security, Mobile IP: Goals, assumptions and requirements, Entities and terminology, IP packet delivery, Agent discovery, Registration, Tunneling and encapsulation Optimizations

Unit-3

Wireless LAN: Infra red Vs. radio transmission, Infrastructure and ad-hoc network, IEEE 802.11: System architecture, Protocol architecture, Physical layer (FHSS,DSSS), Medium access control layer (Basic DFWMAC-DCF using CSMA/CA,) MAC frames, MAC management(Registration, Handoff, Power Management).

Unit–4

Security Issues in Wireless Systems: The Need for Wireless Network Security, Attacks on Wireless Networks, Security Services, Wired Equivalent Privacy (WEP) Protocol, Mobile IP Weaknesses in the WEP Scheme.

Text books:

1. Wireless Networks., P. Nicopolitidis, M. S. Obaidat, G. I. Papadimitriou, A. S. Pomportsis, John Wiley & Sons, Ltd.

2. Mobile Communications, by Dr. Jochen Schiller

3. Principle of Wireless Networks, Kaveh Pahlavan, Prashant Krishnamurthy.

Reference Books:

DSM (HONOURS)
SEMESTER VI
Discipline Specific Elective Course
DSE-4: PROJECT (6 Credits)

Objective:

An elective course designed to acquire special/advanced knowledge, such as supplement study/support study to a project work, and a candidate studies such a course on his own with an advisory support by a teacher/faculty member is called dissertation/project.

Guidelines:

As the project work constitutes a major component in most of the professional programs and it is to be carried out with due care and should be executed with seriousness by the candidates.

Type of Project

As majority of the students are expected to work out a real-life project in some industry/research and development laboratories/educational institutions/software companies, it is suggested that the project is to be chosen which should have some direct relevance in day-to-day activities of the candidates in his/her institution. It is not mandatory for a student to work on a real-life project. The student can formulate a project problem with the help of Guide.

Project Proposal (Synopsis)

The project proposal should be prepared in consultation with the guide. The project proposal should clearly state the project objectives and the environment of the proposed project to be undertaken. The project work should compulsorily include the software development. The project proposal should contain complete details in the following form:

1. Title of the Project
2. Introduction and Objectives of the Project
3. Project Category (RDBMS/OOPS/Networking/Multimedia/Artificial Intelligence/Expert Systems etc.)
4. Analysis (DFDs at least up to second level, ER Diagrams/ Class Diagrams/ Database Design etc. as per the project requirements).
5. A complete structure which includes: Number of modules and their description to provide an estimation of the student's effort on the project. Data Structures as per the project requirements for all the modules. Process Logic of each module. Testing process to be used. Reports generation
6. Tools / Platform, Hardware and Software Requirement specifications
7. Future scope and further enhancement of the project.

Evaluation of the Project

Following Scheme shall be followed for evaluation of the project:
Background of the Problem: 10 marks
Review of Literature: 20 marks
Methodology: 10 marks
Observation and Analysis: 20
Viva Voce: 20 marks
Seminar: 20 marks
Total: 100 marks

DSM (HONOURS)
SEMESTER VI
Skill Enhancement Course
SEC-4: LEADERSHIP & PERSONALITY DEVELOPMENT (4 Credits)
Full marks – 100 (Mid-Sem: 20; End-Sem: 80)

Unit-1

Leadership: Definition and meaning, Importance, Leadership and Management, Leader vs Manager, Essential qualities of an effective leader.

Theories of Leadership: Trait theory, Behavioral theories, Contingency theory

Unit-2

Types of Leaders, Leadership styles: Traditional, Transactional, Transformational, Inspirational and servant leadership and Emerging issues in leadership: Emotional Intelligence and leadership, Trust as a factor, Gender and Leadership

Unit-3

Personality: Concept and Definition, Determinants of personality, Personality traits, Personality characteristics in organizations: Self evaluation, Locus of control, Self-efficacy, Self-esteem, Self-monitoring: Positive and negative Impact.

Unit-4

Organizational Context of Leadership and Personality, Contemporary Business Leaders.

Text Books:

1. Organisational Behaviour, M.Parikh and R.Gupta , TataMcGraw Hill Education Private Limited
2. Organisational Behaviour, D. Nelson, J.C Quick and P. Khandelwal, Cengage Publication.