



CMR Engineering College

Affiliated to JNTU, Hyderabad and Approved by AICTE-New Delhi

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LABORATORY IMPROVEMENT FOR FUTURE TRENDS (LIFT) - - -I

A Guide for execution of Lab Courses

VISION OF THE INSTITUTE:

To be recognized as a premier institution in offering value based and futuristic quality technical education to meet the technological needs of the society.

MISSION OF THE INSTITUTE:

- To impart value based quality technical education through innovative teaching and learning methods.
- To continuously produce employable technical graduates with advanced skills to meet the current and future technological needs of the society.
- To prepare the graduates for higher learning with emphasis on academic and industrial research.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION:

To produce globally competent and industry ready graduates in Computer Science & Engineering by imparting quality education with a know-how of cutting edge technology and holistic personality

MISSION:

- To offer high quality education in Computer Science & Engineering in order to build core competence for the graduates by laying solid foundation in Applied Mathematics, and program framework with a focus on concept building
- The department promotes excellence in teaching, research, and collaborative activities to prepare graduates for professional career or higher studies
- Creating intellectual environment for developing logical skills and problem solving strategies, thus to develop, able and proficient computer engineer to compete in the current global scenario

Program Educational Objectives (PEOs)

- Excel in professional career or higher education by acquiring knowledge in mathematical, computing and engineering principles
- To provide intellectual environment for analyzing and designing computing systems for technical needs
- Exhibit professionalism, multidisciplinary teamwork and adapt to current trends by engaging in lifelong learning and practice their profession with legal, social and ethical responsibilities

PROGRAM SPECIFIC OUTCOMES (PSO's)

- Professional Skills and Foundations of Software development: Ability to analyze, design and develop applications by adopting the dynamic nature of Software developments
- Applications of Computing and Research Ability: Ability to use knowledge in cutting edge technologies in identifying research gaps and to render solutions with innovative ideas

1. AIM OF THE LIFT:

The main aim of the LIFT programme is to innovate, modify the existing facilities in labs, to create awareness among the students and develop Industry –Institution interactions and reach the standards in laboratories

2. FUNCTIONS OF THE LIFT:

- I. To create better understanding concepts of LIFT and other lab related activities among the staff and lab technicians for better improvement.
- II. To Arrange LIFT Presentations from each department about the lab activities by the staff handling the labs. (Lab Planners)
- III. To Prepare GAP ANALYSIS: This involves collection of requirements from each lab of every department, information about expansion of labs, repairs and maintenance of labs etc.
- IV. To arrange Industrial Visits/ Industrial training programs in coordination with concerned lab staff and Heads of the departments.
- V. A Report on Shadow Engineering: This involves arrangement of Industrial and Practical learning, Submission of Industrial Visit report, Technical Survey reports and Market Survey of a product for development in laboratories.
- VI. Verification of all the laboratories in every department by the LIFT Team along with the Principal and the concerned HODs, to check whether the activities are going according to LIFT guidelines, to check the Record Keeping, Lab Manuals and Viva sessions etc.
- VII. Check for LEAD Experiments and its follow up.
- VIII. Submission of proposals related to R&D, Project and Consultancy from lab staff to the Principal for further approvals.

LAB IMPROVEMENT FOR FUTURE TRENDS PROGRAMME (LIFT)

INDEX

CONTENTS:

- 1. OBJECTIVES AND RELEVANCE**
- 2. SCOPE**
- 3. PREREQUISITES**
- 4. SYLLABUS AS PER JNTUH**
- 5. LAB SCHEDULE**
- 6. SUGGESTED BOOKS**
- 7. WEBSITES (USEFUL LINKS)**
- 8. EXPERT DETAILS**
- 9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D**
- 10. PROPOSALS**
- 11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
- 12. CALIBRATION, TESTING AND INSPECTION**
- 13. PREVENTIVE MAINTENANCE SCHEDULES**
- 14. TROUBLESHOOTING**

1. OBJECTIVES AND RELEVANCE:

The main objective of the LIFT concept in lab course is to provide practical hands on experience for each student by providing them with good exposure to different experiments and to uplift the knowledge levels of the student, with different applications in various fields.

2. SCOPE:

The main scope of the LIFT lab course is to cover all the experiments as per the schedule given in the prescribed week wise periods. With this, a student can better understand the concepts and operating systems so that he could get better knowledge about each lab.

3. PREREQUISITES:

The basic level idea related to each experiment should be provided to the students before conducting main lab course. Following details are to be explained related to experiment:

1. Introduction to experiment – 30 min
2. The Operating of the equipment/instrument/software
3. Record of Experimental Results.
4. Sample Calculations / Executable Programs

4. SYLLABUS AS PER JNTUH:

The lab course should be planned as per the JNTUH syllabus. In this, LEAD experiments should also be included in the cycle of experiments.

5. (A) LAB SCHEDULE:

The lab schedule should be planned once in a week. The week wise scheduled experiment should be completed.

CYCLE 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 |
|---------|--------|--------|--------|--------|--------|--------|--------|
| B1 | Demo | Exp.1 | Exp.2 | Exp.10 | Exp.9 | Exp.8 | Test |
| B2 | Demo | Exp.2 | Exp.10 | Exp.9 | Exp.8 | Exp.1 | test |
| B3 | Demo | Exp.10 | Exp.9 | Exp.8 | Exp.1 | Exp.2 | test |
| B4 | Demo | Exp.9 | Exp.8 | Exp.1 | Exp.2 | Exp.10 | test |
| B5 | Demo | Exp.8 | Exp.1 | Exp.2 | Exp.10 | Exp.9 | test |

CYCLE 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 |
|---------|--------|--------|--------|--------|--------|--------|--------|
| B1 | Exp.3 | Exp.4 | Exp.6 | Exp.11 | Exp.12 | Exp.5 | test |
| B2 | Exp.7 | Exp.6 | Exp.11 | Exp.12 | Exp.5 | Exp.4 | test |
| B3 | Exp.3 | Exp.11 | Exp.12 | Exp.5 | Exp.4 | Exp.6 | test |
| B4 | Exp.10 | Exp.12 | Exp.5 | Exp.4 | Exp.6 | Exp.11 | test |
| B5 | Exp.9 | Exp.5 | Exp.4 | Exp.6 | Exp.11 | Exp.12 | test |

(B) VIVA SCHEDULE:

The viva schedule should be planned prior to the lab experiment.

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 |
|----------|--------|--------|--------|--------|--------|
| B1,B2,B3 | viva | | | | |
| B1,B2,B3 | | viva | | | |
| B1,B2,B3 | | | viva | | |
| B1,B2,B3 | | | | viva | |
| B1,B2,B3 | | | | | viva |

ROUND - 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 |
|---------|--------|--------|--------|--------|--------|
| SG1 | viva | | | | |
| SG2 | | viva | | | |
| SG3 | | | viva | | |
| SG4 | | | | viva | |
| SG5 | | | | | viva |

*SG: Selected Group with a maximum of 6 or 12 students

(C) Scheme of Evaluation:

The scheme of evaluation for internal and external exams as follows:

| LAB INTERNAL: | | | | | | | |
|---------------------------------|----------------------|-------------------------------|---------|---------|-------------------------|---------------------|---------|
| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-4 | Marks-3 | Marks-3 |
| Total Marks-25 | | | | | | | |

| LAB EXTERNAL: | | | |
|-----------------------|---|---|--|
| S.NO | Write up | Final Evaluation | Viva |
| 1 | 1. Aim 2. Procedure 3. Program 4. Expected output. | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

6. SUGGESTED BOOKS:

The suggested books should be recommended to the students as per the JNTUH syllabus prescribed.

7. WEBSITES (USEFUL LINKS):

The useful links should be provided to the students, where they can get an easy access to the knowledge of the experiment.

8. EXPERT DETAILS:

The expert details should be provided based on the experimental importance.

9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D:

The lab course should be designed in such a way that it should meet the requirements of research and development as well as consultancy projects. Also the Proposals of Project/R&D/Consultancy are as follows:

Proposal 1: Project Design & Execution

Proposal 2: R& D Level Project Design & Execution

Proposal 3: Consultancy Task / Project Design & Development

Proposal 1: Project Design & Execution:

A Proposal of a hobby/mini/proto/general/model/proto type project including the abstract, Block Diagram/Circuit/Flow diagram and clear references may be presented and executed.

Proposal 2: R& D Level Project Design & Execution:

1. An exact paper from a National/International journal in this entitled area/subject/area (IEEE Format)

AND/OR

2. An article/white paper from a magazine /journal/weekly/any periodical in the entitled Subject

AND/OR

3. An Advanced technology development/ proposal/article publication from any source of information.

Proposal 3: Consultancy Task / Project Design & Development:

A program/machine/product of utility may be proposed to develop for in-house usage/ Industrial requirements which may be useful for any outside agency, that can be marketable in order to generate revenue through consultancy

FUNDED/UNFUNDED PROPOSALS (if any):

The proposals for AICTE grants like (SDPs, RPS and MODROBES etc) UGC grants, DST CPRI and other funding agencies may be submitted by giving Title and Abstract/Objective OR Self Funded program proposals may be submitted for Management approvals.

10. PROPOSALS:

(WEEK WISE INDUSTRIAL VISITS IN-HOUSE OR OUTSIDE VISIT OR TRAINING PROGRAMS)

TABLE 1 : INDUSTRIAL VISITS

| S.No. | Type of industry | Nature of industry | Date of visit | No. of students participated | Year/branch | Remarks |
|-------|------------------|--------------------|---------------|------------------------------|-------------|---------|
| | | | | | | |
| | | | | | | |

**TABLE 2: INDUSTRIAL TRAINING (Shadow Engineering)
(Career Visit Approval)**

| S.No. | Name of the Course | Nature of industry | Duration of Training | Authority | Date of Training/Certificate No. | Remarks |
|-------|--------------------|--------------------|----------------------|-----------|----------------------------------|---------|
| | | | | | | |
| | | | | | | |

11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS: (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM):

OBJECTIVES OF SHADOW ENGINEERING:

1. The program uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future assignment.
4. This program leads to matrixing with the students.

12. ACTIVITIES IN LIFT PROGRAM:

i. CALIBRATION/INSTALLATION AND TESTING:

Calibration: Aim of this concept is to check:

- i. whether all the equipment is functioning correctly as per the standards
- ii. To bring correctness in the errors of instrument or equipment.
- iii. To rectify the errors if any

Installation: Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment

Testing: Aim of this concept is to test the equipment after installation whether it meets the existing standards.

The list of equipments (hardware/software):

- i. Necessity of tools for development and testing.
- ii. Equipment to be calibrated.
- iii. Installation of supporting equipment if any.

After calibration the details of equipment should be submitted in following format.

| S.No. | Type of equipment | Certificate no. | Certificate issued by | Date of calibration | Date of calibration due | Remarks |
|-------|-------------------|-----------------|-----------------------|---------------------|-------------------------|---------|
| | | | | | | |
| | | | | | | |

Calibration, Testing and Installation details equipment wise are mentioned as follows:

Case 1: Calibration of Equipment ----- if any

Case 2: Installation of Equipment ----- if any

Case 3: Testing of Equipment ----- if any

13. PREVENTIVE MAINTENANCE SCHEDULES

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following format.

| S.No. | Name of the Equipment | Date of Maintenance | Type of Activity | Remarks |
|-------|-----------------------|---------------------|------------------|---------|
| | | | | |

(2) Maintenance Reports should be duly signed by in charges as well as HODs and duly approved by Principal periodically.

14. TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S.No., Equipment Name , Type of Problem (Too much Noise, Abnormal Sound, Corrupt Software, Anti Virus Problem, Missing of Display, CRT not working, Motor is not giving signal, Digital display is not working, Break of tools, Misalignment of machine elements, PLC is not properly working), Expected Reasons (Bearing failure, Improper alignment of machine centres, Missing of vibration pads etc) Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Equipment Name | Type of Trouble | Remedial Activity | Remarks |
|-------|----------------------------|----------------|-----------------|-------------------|---------|
| | | | | | |

NAME OF THE LABORATORIES

| S.No. | YEAR-SEM | NAME OF THE LAB |
|--------------|------------------------|-------------------------------------|
| 1 | II B.TECH-ISEM | DATA STRUCTURES |
| 2 | II B.TECH-ISEM | ELECTRICAL AND ELECTRONICS |
| 3 | III B.TECH-ISEM | OPERATING SYSTEMS |
| 4 | III B.TECH-ISEM | COMPILER DESIGN |
| 5 | IV B.TECH-ISEM | LINUX ROGRAMMING |
| 6 | IV B.TECH-ISEM | DATA WARE HOUSING AND MINING |

SUBJECTWISE LAB PLANNER

Name of the Subject: DATA STRUCTURES LAB

CONTENTS:

- 1. OBJECTIVES AND RELEVANCE**
- 2. SCOPE**
- 3. PREREQUISITES**
- 4. SYLLABUS AS PER JNTUH**
- 5. LAB SCHEDULE**
- 6. SUGGESTED BOOKS**
- 7. WEBSITES (USEFUL LINKS)**
- 8. EXPERT DETAILS**
- 9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D**
- 10. PROPOSALS**
- 11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
- 12. CALIBRATION, TESTING AND INSPECTION**
- 13. PREVENTIVE MAINTENANCE SCHEDULES**
- 14. TROUBLESHOOTING**

1. OBJECTIVES AND RELEVANCE

To develop skills to design and analyze simple linear and non linear data structures, to strengthen the ability to identify and apply the suitable data structure for the given real world problems and to gain knowledge in practical applications of data structures.

2. SCOPE

Assess how the choice of data structure design methods impacts the performance of programs. Choose the appropriate data structure design method for a specified application. Solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps, binary search trees, and graphs and writing programs for these solutions.

3. PREREQUISITES

This subject recommends continuous practice of powerful modern language that combines the power, elegance and flexibility of C programming.

4. JNTUH Syllabus

Week1: Write a C program that uses functions to perform the following:

- a) Create a singly linked list of integers.
- b) Delete a given integer from the above linked list.
- c) Display the contents of the above list after deletion.

Week2: Write a C program that uses functions to perform the following:

- a) Create a doubly linked list of integers.
- b) Delete a given integer from the above doubly linked list.
- c) Display the contents of the above list after deletion.

Week3: Write a C program that uses stack operations to convert a given infix expression into its postfix equivalent. Implement the stack using an array.

Week 4: Write C programs to implement a double ended queue ADT using

- i) array and ii) doubly linked list respectively.

Week 5: Write a C program that uses functions to perform the following:

- a) Create a binary search tree of characters.
- b) Traverse the above Binary search tree recursively in Post order.

Week 6: Write a C program that uses functions to perform the following:

- a) Create a binary search tree of integers.
- b) Traverse the above Binary search tree non recursively in order.

Week 7: Write C programs for implementing the following sorting methods to arrange a list of integers in Ascending order:

- a) Insertion sort
- b) Merge sort

Week 8: Write C programs for implementing the following sorting methods to arrange a list of integers in ascending order:

- a) Quick sort
- b) Selection sort

Week 9: i) Write a C program to perform the following operation:

- a) Insertion into a B-tree.
- ii) Write a C program for implementing Heap sort algorithm for sorting a given list of integers in ascending order.

Week 10: Write a C program to implement all the functions of a dictionary (ADT) using hashing.

Week 11: Write a C program for implementing Knuth-Morris- Pratt pattern matching algorithm.

Week 12: Write C programs for implementing the following graph traversal algorithms:

- a) Depth first traversal
- b) Breadth first traversal

LEAD PROGRAMS

1. Write a C program to perform operations of circular queue.
2. Write a C program to check whether the given matrix is sparse matrix or not.

MAIN LINKAGE OF DATA STRUCTURES THEORY WITH LAB EXPERIMENTS:

UNIT-I

EXPERIMENT NO.1

Write a C program that uses functions to perform the following:

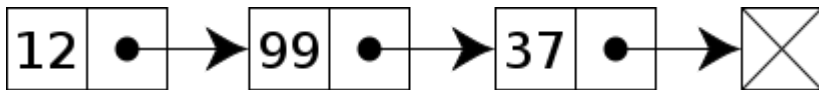
- a) Create a singly linked list of integers.
- b) Delete a given integer from the above linked list.
- c) Display the contents of the above list after deletion.

OBJECTIVE: The main objective is to understand singly linked lists and their operations

PRE REQUISITES: Basic understanding of nodes and pointers

DESCRIPTION:

Singly Linked Lists are a type of data structure. It is a type of list. In a singly linked list each node in the list stores the contents of the node and a pointer or the reference to the next node in the list. It does not store any pointer or reference to the previous node. It is called a singly linked list because each node only has a single link to another node. To store a singly linked list, you only need to store a reference or pointer to the first node in that list. The last node has a pointer to nothingness to indicate that it is the last node.



A singly linked list whose nodes contain two fields: an integer value and a link to the next node

APPLICATIONS:

- 1) A singly-linked list is ideally suited to stacks, queues
- 2) Reduces access time and may expand in real time without memory overhead

EXPERIMENT NO.2

Write a C program that uses functions to perform the following:

- a) Create a doubly linked list of integers.
- b) Delete a given integer from the above doubly linked list.
- c) Display the contents of the above list after deletion.

OBJECTIVE:

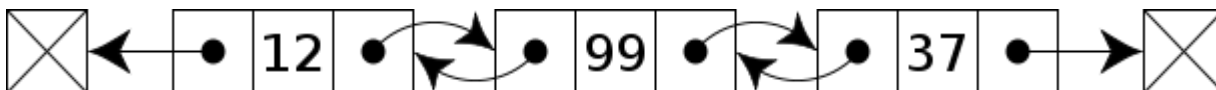
The main objective is to understand doubly linked list and their operations

PRE REQUISITES:

Basic understanding of singly linked lists and their operations

DESCRIPTION:

In a **doubly linked list**, each node contains, besides the next-node link, a second link field pointing to the *previous* node in the sequence. The two links may be called **forward(s)** and **backwards**, or **next** and **prev(previous)**.



A doubly linked list whose nodes contain three fields: an integer value, the link forward to the next node, and the link backward to the previous node

A technique known as [XOR-linking](#) allows a doubly linked list to be implemented using a single link field in each node.

APPLICATIONS:

- 1) The cache in your browser allows you to hit the BACK button (a linked list of URLs)
- 2) Undo functionality in Photoshop or Word (a linked list of state)
- 3) Allows you to hit the BACK button (a linked list of URLs)
- 4) A great way to represent a deck of cards in a game.

UNIT-II

EXPERIMENT NOS. 3 & 4

- 3) Write a C program that uses stack operations to convert a given infix expression into its postfix equivalent. Implement the stack using an array.
- 4) Write C programs to implement a double ended queue ADT using
 - i) array and
 - ii) doubly linked list respectively.

OBJECTIVE:

The main objective is to understand stack and its operations, de queue and its operations

PRE REQUISITES:

Basics of arrays and linked list concepts

Basic introduction of ADT

Knowledge of different types of expressions

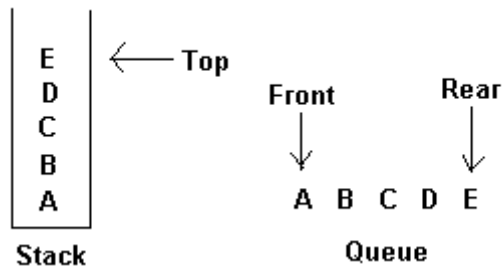
DESCRIPTION:

A *stack* is an ordered list in which all insertions and deletions are made at one end, called the *top*.

A *queue* is an ordered list in which all insertions take place at one end, the *rear*, while all deletions take place at the other end, the *front*.

Given a stack $S=(a[1],a[2],\dots,a[n])$ then we say that a_1 is the bottommost element and element $a[i]$ is on top of element $a[i-1]$, $1 < i \leq n$.

When viewed as a queue with $a[n]$ as the rear element, one says that $a[i+1]$ is behind $a[i]$, $1 < i < n$.



STACK APPLICATIONS:

- 1) The simplest application of a stack is to reverse a word
- 2) Another application is an "undo" mechanism in text editors; this operation is accomplished by keeping all text changes in a stack.
- 2) Compiler's syntax check for matching braces is implemented by using stack.
Arithmetic expression evaluation

QUEUE APPLICATIONS:

- 1) Used by operating system in scheduling jobs and spooling (queue of print jobs)
- 2) Disk I/O calls

UNIT III

EXPERIMENT NO. 12

Write C programs for implementing the following graph traversal algorithms:

- a) Depth first traversal
- b) Breadth first traversal

OBJECTIVE: The main objective is to understand graph traversal algorithms.

PRE REQUISITES: Basic understanding of nodes, vertices and graphs.

DESCRIPTION:

Breadth-first traversal

In [graph theory](#), breadth-first traversal is a [strategy for searching in a graph](#) when search is limited to essentially two operations: (a) visit and inspect a node of a graph; (b) gain access to visit the nodes that neighbor the currently visited node. The BFS begins at a root node and inspects all the neighboring nodes. Then for each of those neighbor nodes in turn, it inspects their neighbor nodes which were unvisited, and so on.

Depth first traversal

Depth-first search (DFS) is a general technique for traversing a graph

A DFS traversal of a graph G

Visits all the vertices and edges of G

Determines whether G is connected

Computes the connected components of G

Computes a spanning forest of G

APPLICATIONS

- 1) Finds the [shortest path](#) between two nodes u and v
- 2) To solve complex routing problems such as routing airlines among the airports
- 3) To route messages over computer networks

UNIT IV

EXPERIMENT NOS. 7, 8 & 9

7) Write C programs for implementing the following sorting methods to arrange a list of integers in ascending order

- a) Insertion sort b) Merge sort

8) Write C programs for implementing the following sorting methods to arrange a list of integers in ascending order

- a) Quick sort b) Selection sort

9) ii) Write a C program for implementing Heap sort algorithm for sorting a given list of integers in ascending order.

OBJECTIVE:

To arrange the list of integers in an ascending order by using various sorting techniques.

PREREQUISITES:

It requires the knowledge of arrays

DESCRIPTION:

Sorting is the process of placing elements from a collection in some kind of order. Sorting makes searching process easy. Various sorting techniques are available like Insertion sort, Merge sort, Quick sort, Selection sort, Heap sort, etc. to arrange elements in particular order.

APPLICATIONS:

- 1) A list of words could be sorted alphabetically or by length
- 2) A list of cities could be sorted by population, by area, or by zip code

EXPERIMENT NO. 10

Write a C program to implement all the functions of a dictionary (ADT) using hashing.

OBJECTIVE:

The main objective is to perform the main operations of a dictionary using hashing.

PREREQUISITES:

Student has to know about ADT and hashing technique.

DESCRIPTION:

The dictionary ADT models a searchable collection of key-element entries. The main operations of a dictionary are searching, inserting, and deleting items. Multiple items with the same key are allowed.

APPLICATIONS:

- 1) word-definition pairs
- 2) credit card authorizations
- 3) DNS mapping of host names

UNIT-V**EXPERIMENT NOS. 5, 6 & 9**

- 5) Write a C program that uses functions to perform the following:
 - a) Create a binary search tree of characters.
 - b) Traverse the above Binary search tree recursively in Post order.
- 6) Write a C program that uses functions to perform the following:
 - a) Create a binary search tree of integers.
 - b) Traverse the above Binary search tree non recursively in order.
- 9) i) Write a C program to perform Insertion into a B-tree.

OBJECTIVE:

To represent and access the elements in a binary search tree, B-tree

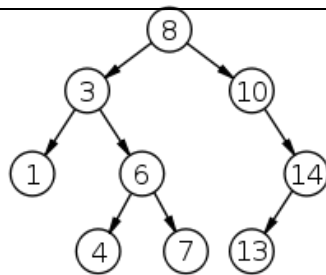
PREREQUISITES:

Basic knowledge of tree data structure and its traversal operations.

DESCRIPTION:

Binary search tree (BST)

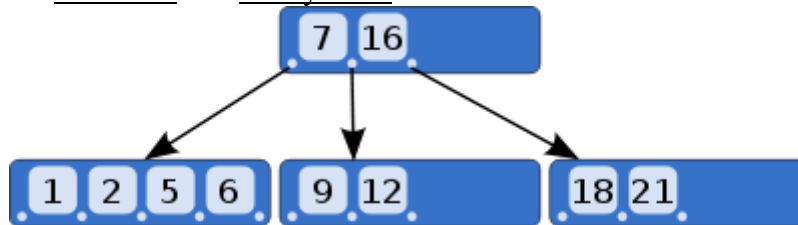
A special type of binary tree called a binary search tree (BST) is often useful. A BST (with no duplicate elements) has the property that, for every node in the tree, the value of any node in its left sub tree is less than the value of the node, and the value of any node in its right sub tree is greater than the value of the node.



A binary search tree of size 9 and depth 3, with root 8 and leaves 1, 4, 7 and 13

B-tree

A B-tree is a tree data structure that keeps data sorted and allows searches, sequential access, insertions, and deletions in logarithmic time. The B-tree is a generalization of a binary search tree in that a node can have more than two children (Comer 1979, p. 123). Unlike self-balancing binary search trees, the B-tree is optimized for systems that read and write large blocks of data. It is commonly used in databases and file systems.



A B-tree of order 2

APPLICATIONS:

1. Used in *many* search applications where data is constantly entering/leaving,
2. To maintain ordered collection of data

EXPERIMENT NO: 11 Write a C program for implementing Knuth-Morris- Pratt pattern matching algorithm.

OBJECTIVE:

To search for occurrences of a "word" W within a main "text string" S

PREREQUISITES:

Basic knowledge of searching algorithms

DESCRIPTION:

In computer science, the **Knuth–Morris–Pratt string searching algorithm** (or **KMP algorithm**) searches for occurrences of a "word" W within a main "text string" S by employing the observation that when a mismatch occurs, the word itself embodies sufficient information to determine where the next match could begin, thus bypassing re-examination of previously matched characters.

APPLICATIONS:

Searches for occurrences of a "word" W within a main "text string" S

LEAD:

1. Write a program to perform operations of circular queue.
2. Write a program to check the given matrix is sparse matrix or not using C.

5(A) LAB SCHEDULE:**CYCLE 1:**

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| B1(501 to 512) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B2(513 to 524) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B3(525 to 536) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B4(537 to 548) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B5(549 to 560) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week-16 |
|----------------|---------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| B1(501 to 512) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | Test |
| B2(513 to 524) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | Test |
| B3(525 to 536) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp11 | Exp 12 | LEAD1 &2 | Test |
| B4(537 to 548) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | Test |
| B5(549 to 560) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | Test |

CYCLE 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| B6(561 to 572) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B7(573 to 584) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B8(585 to 596) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B9(597 to 5A8) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B10(5A9 to 5C0) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week 16 |
|-----------------|---------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| B6(561 to 572) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp.11 | Exp 12 | LEAD1 &2 | test |
| B7(573 to 584) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp.12 | LEAD1 &2 | test |
| B8(585 to 596) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | test |
| B9(597 to 5A8) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | test |
| B10(5A9 to 5C0) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | test |

CYCLE 3

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| B11(5C1 to 5D2) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B12(5D3 to 5E4) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B13(5E5 to 5F6) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B14(5F7 to 5G8) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |
| B15(5G9 to 5H8) | Demo | Exp.1 | Exp.2 | Exp.3 | Exp.4 | Exp.5 | test | Exp.6 |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week 16 |
|-----------------|---------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| B11(5C1 to 5D2) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp.11 | Exp 12 | LEAD1 &2 | test |
| B12(5D3 to 5E4) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp.12 | LEAD1 &2 | test |
| B13(5E5 to 5F6) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | test |
| B14(5F7 to 5G8) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | test |
| B15(5G9 to 5H8) | Exp.7 | Exp.8 | Exp.9 | Exp.10 | Exp 11 | Exp 12 | LEAD1 &2 | test |

(B) VIVA SCHEDULE: The viva schedule should be planned prior to the lab experiment.

CYCLE 1:

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|
| B1,B2,B3 | viva | | | | | viva | | | | | Viva | | | |
| B4,B5,B1 | | viva | | | | | viva | | | | | viva | | |
| B2,B3,B4 | | | viva | | | | | viva | | | | | viva | |
| B5,B1,B2 | | | | Viva | | | | | Viva | | | | | viva |
| B2,B3,B4 | | | | | viva | | | | | viva | | | | |

CYCLE 2:

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|-----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|
| B6,B7,B8 | viva | | | | | viva | | | | | viva | | | |
| B9,B10,B6 | | viva | | | | | viva | | | | | viva | | |
| B7,B8,B9 | | | viva | | | | | viva | | | | | viva | |
| B10,B6,B7 | | | | Viva | | | | | viva | | | | | viva |
| B8,B9,B10 | | | | | viva | | | | | viva | | | | |

CYCLE 3:

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|-------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|
| B11,B12,B13 | viva | | | | | viva | | | | | viva | | | |
| B14,B15,B11 | | viva | | | | | viva | | | | | viva | | |
| B12,B13,B14 | | | viva | | | | | viva | | | | | viva | |
| B15,B11,B12 | | | | viva | | | | | viva | | | | | viva |
| B13,B14,B15 | | | | | viva | | | | | viva | | | | |

ROUND - 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|
| SG1 | viva | | | | | | | | | |
| SG2 | | | viva | | | | | | | |
| SG3 | | | | | viva | | | | | |
| SG4 | | | | | | | viva | | | |
| SG5 | | | | | | | | | viva | |

*SG: Selected Group with a maximum of 6 or 12 students

(C) Scheme of Evaluation:

The scheme of evaluation for internal and external exams is as follows:

| LAB INTERNAL: | | | | | | | |
|---------------------------------|---------------------------------|--------------------------------------|----------------|----------------|-------------------------|--------------------------------|----------------|
| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-4 | Marks-3 | Marks-3 |
| Total Marks-25 | | | | | | | |

6. SUGGESTED BOOKS:

1. C and Data Structures, Third Edition, P.Padmanabham, BS Publications.
2. C and Data Structures, Prof. P.S.Deshpande and Prof. O.G. Kakde, Dreamtech Press.
3. Data structures using C, A.K.Sharma, 2nd edition, Pearson.
4. Data Structures using C, R.Thareja, Oxford University Press.
5. C and Data Structures, N.B.Venkateswarlu and E.V.Prasad, S.Chand.
6. C Programming and Data Structures, P.Radha Krishna, Hi-Tech Publishers.

| LAB EXTERNAL: | | | |
|-----------------------|---|--|---|
| S.NO | Write up | Final Evaluation | Viva |
| 1 | 1. Aim 2. Procedure 3. Program 4. Expected output. | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

7. WEBSITES

<http://www.c4learn.com/data-structure>

http://www-gs.informatik.tu-cottbus.de/cs1_v06.pdf

<http://www.users.pjwstk.edu.pl/~msyd/wyka-eng/complexity2.pdf>

<http://www.cs.cmu.edu/~adamchik/15->

<121/lectures/Stacks%20and%20Queues/Stacks%20and%20Queues.html>

<http://www.cprogramming.com/>

ftp://ftp.awl.com/ceng/authors/gray/Graphs%20Chapter/Gray_CH13_rev.pdf

<http://www.sanfoundry.com/c-programming-examples-searching-sorting/>

<https://www.cs.auckland.ac.nz/software/AlgAnim/AVL.html>

8. EXPERTS DETAILS

The Expert Details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally. There are a few others known as well.

INTERNATIONAL

Adam Drozdek, Associate Professor of Computer Science

M.S., Wright State University
B.S., M.A., Wroclaw University
Ph.D., Jagiellonian University

NATIONAL

Prof E. Balagurusamy,
UPSC Member,
Vice- Chancellor of Anna University,
Chennai.

REGIONAL:

- 1) **Dr. T. Ramesh**, Ph.D,e-mail: rmesht@nitw.ac.in,phone: 0870-2462702(O)
- 2) **Dr.Ch. Sobhan Babu** , Email :[sobhan \[at\] iith.ac.in](mailto:sobhan[at]iith.ac.in)

9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D:

PROPOSAL 1: PROPOSAL FOR PROJECT ACTIVITY

Contacts Manager is a simple project in C. This is similar to Contact Manager in Mobile phones. You can add contacts, remove contacts, edit contacts, search contacts and view all contacts. Graphics is not used, just a console application suitable for learning purpose for the beginner can be used as a reference for college mini project. In this project you learn file handling, how to write data, delete data and search data from file.

PROPOSAL 2 : R & D ACTIVITY

Performance Analysis of BSTs in System Software*

Ben Pfaff,Stanford University Department of Computer Science blp@cs.stanford.edu

Abstract

Binary search tree (BST) based data structures, such as AVL trees, red-black trees, and splay trees, are often used in system software, such as operating system kernels. Choosing the right kind of tree can impact performance significantly, but the literature offers few empirical studies for guidance. We compare 20 BST variants using three experiments in real world scenarios with real and artificial workloads. The results indicate that when input is expected to be randomly ordered with occasional runs of sorted order, red-black trees are preferred; when insertions often occur in sorted order, AVL trees excel for later random access, whereas splay trees perform best for later sequential or clustered access. For node representations, use of parent pointers is shown to be the fastest choice, with threaded nodes a close second choice that saves memory; nodes without parent pointers or threads suffer when traversal and modification are combined; maintaining an in-order doubly linked list is advantageous when traversal is very common; and right-threaded nodes perform poorly.

Proposal 3: Consultancy Task / Project Design & Development

The above project will be able to complete and it will be used as consultancy for better corporate relations environment for revenue generation.

10. PROPOSALS

Relevant IV (Industrial Visit) (VIP= Visit of Industrial Proposals)

If students visited different software development companies like Electronic arts, Game Loft, prokarma.

| S.No | Type of industry | Nature of industry | Date of visit | No. of students participated | Year/branch | remarks |
|------|------------------|--------------------|---------------|------------------------------|-------------|---------|
| 1 | MNC | software | | 60 | CSE-A | |
| 2 | MNC | software | | 60 | CSE-B | |
| 3 | Level-3 | software | | 60 | CSE-C | |

11. SHADOW ENGINEERING

GUIDELINES FOR SHADOW ENGINEERING (VIP)

INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)

OBJECTIVES OF SHADOW ENGINEERING:

1. To uplift the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

**TABLE 2: INDUSTRIAL TRAINING (Shadow Engg)
(Career Visit Approval)**

| S.No | Name of the Course | Nature of industry | Duration of Training | Authority | Date of Training/Certificate No. | remarks |
|------|--------------------|--------------------|----------------------|-----------|----------------------------------|---------|
| 1 | Data Structures | Software | 1 WEEK | | | |
| | | | | | | |

12. CALIBRATION, TESTING & INSPECTION

Calibration

There is no need of any equipment to be calibrated.

Testing & Inspection:

Black Box Testing

The technique of testing without having any knowledge of the interior workings of the application is Black Box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, when performing a black box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

White Box Testing

White box testing is the detailed investigation of internal logic and structure of the code. White box testing is also called glass testing or open box testing. In order to perform white box testing on an application, the tester needs to possess knowledge of the internal working of the code. The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

Grey Box Testing

Grey Box testing is a technique to test the application with limited knowledge of the internal workings of an application. In software testing, the term *the more you know the better* carries a lot of weight when testing an application. Mastering the domain of a system always gives the tester an edge over someone with limited domain knowledge. Unlike black box testing, where the tester only tests the application's user interface, in grey box testing, the tester has access to design documents and the database. Having this knowledge, the tester is able to prepare better test data and test scenarios when making the test plan. In case if the software is lost we follow the following procedure to install.

Installation: Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment

Linux OS Installation steps :(Red Hat Linux GUI)

Step 1 – Insert the Red Hat Linux DVD into the DVD-drive of your computer. As soon as the following screen pops up, press ‘Enter’ to install Red Hat Enterprise Linux (RHEL) through GUI mode.

Step 2- RHEL installer would then prompt you to conduct a check as to whether the CD media from which you’re installing is functioning correctly or not. Choose ‘Skip’, press enter and the installation would begin.

Step 3- Next, we need to select the language – English or any other language as per your preference, and then press ‘Next’.

Step 4- In this step, the RHEL installer would ask you about the appropriate type of keyboard for the system. We take the ‘US English’ keyboard; you can pick any other option depending on the type of your keyboard. Then press ‘Next’ to move to the next step.

Step 5- Next, the installer would ask for an ‘installation number’ if you wish to install full set of Red Hat functionalities. Enter the installation number and press ‘OK’ if you have an officially licensed installation number (for corporate clients that buy Red Hat’s backup support and full features).

Step 6- The Red Hat installer would then require you to create partitions in your computer’s hard disk for the installation. You can do it in four ways but the simplest way is to select ‘Use free space on selected drives and create default layout’ as this option will not affect any other OS residing in your system.

Step 7- In this step you must create the required system partitions and mount points such as ‘/boot’, ‘/home’, ‘swap’ etc which are required for the Linux’s proper functioning.

Step 8- This step pertains to the default OS that will be loaded by the GRUB loader.

Step 9- This step pertains to the network settings of the Linux system that you are going to install. You can select the Ethernet devices through which the system would communicate with other devices in the network.

Step 10- The next step is to adjust the system clock to your particular time zone. Select your time zone and then click ‘Next’.

Step 11 – This is a very important step that deals with the root (super-user) password for the system. Type the password and confirm it and then click next.

Step 12 – The RHEL installer would then prompt you about if you wish to install some extra ‘Software Development’ or ‘Web Server’ features. By default, keep it at ‘Customize later’ and press ‘Next’.

Step 13- This next step will initiate the installation of Red Hat Linux, press 'Next' to begin the process.

Step 14- Upon the completion of installation you should see the following screen. Press Reboot and you'd be ready to use your newly installed Red Hat Linux OS.

Commands in Linux for creating , running , testing C programs

Creating program:

vi program name .c

vi commands :

for saving the commands is: esc:wq

for editing the command is : esc:i

for compiling the command is :gcc programname.c -o programname

for running the command is : ./programname

13. PREVENTIVE MAINTENANCE SCHEDULE

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

| S.No. | Software | Date of Maintenance | Type of Activity | Remarks |
|-------|------------------|---------------------|---------------------|---------|
| 1 | Operating System | | Softwares installed | |
| | | | | |

(2) Maintenance Reports should be duly signed by in charges as well as HODs and duly approved by Principal periodically.

14. TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific lab.

S.No., Software Tool, Type of Problem (insufficient memory, type of version, security issues, authentication issues, Corrupt Software, proper configuration, Missing of Display, CRT not working), Expected Reasons .Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Software Tool | Type of Trouble | Remedial Activity | Remarks |
|-------|----------------------------|---------------|-----------------|-------------------|---------|
| | | | | | |

ELECTRICAL AND ELECTRONICS LAB

CONTENTS:

- 1. OBJECTIVES AND RELEVANCE**
- 2. SCOPE**
- 3. PREREQUISITES**
- 4. SYLLABUS AS PER JNTUH**
- 5. LAB SCHEDULE**
- 6. SUGGESTED BOOKS**
- 7. WEBSITES (USEFUL LINKS)**
- 8. EXPERT DETAILS**
- 9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D**
- 10. PROPOSALS**
- 11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
- 12. CALIBRATION, TESTING AND INSPECTION**
- 13. PREVENTIVE MAINTENANCE SCHEDULES**
- 14. TROUBLESHOOTING**

1. OBJECTIVES AND RELEVANCE

This lab introduces the basic concepts of Electrical and Electronics, which forms the core of the advanced concepts in the area of Electrical and Electronics. The emphasis is laid on the basic circuit designing to the solutions of complex circuits, also concepts regarding working of electrical machines and measuring equipments is explored.

Different Diodes are explored to have the basic idea of different characteristics of a diode. In this lab we get the idea of flow of current and voltage in simple and complex circuits practically.

2. SCOPE

The scope of this lab is to provide a thorough knowledge of the flow of voltage and current in electrical networks and circuits, to explore the applications of a diode and to find the different applications of machines and network reduction techniques. It also provides the insight of the working and applications of electrical machines and measuring instruments.

3. PREREQUISITES

Before the start of the experiments one should have the knowledge of voltage, current, basic electrical components, relation between V and I w.r.t electrical components, difference between ac and dc, different diode, transistor and rectifier characteristics.

4. SYLLABUS – JNTUH

PART-A

| S. No | Name of the Experiment |
|-------|---|
| 1 | Verification of Superposition and Reciprocity theorems. |
| 2 | Verification of Maximum Power Transfer Theorem. Verification on DC with Resistive load. |
| 3 | Experimental Determination of Thevenin's theorem. |
| 4 | Magnetization characteristics of D.C Shunt Generator. |
| 5 | Swinburne's Test on DC Shunt Machine (Predetermination of efficiency of a given DC shunt machine working as motor and generator). |
| 6 | Brake test on DC Shunt Motor. Determination of performance Characteristics. |
| 7 | OC & SC tests on Singles-phase Transformer (Predetermination of efficiency and regulation at given power factors). |
| 8 | Brake test on 3-phase Induction Motor (Performance characteristics). |

PART – B

| S. No | Name of the Experiment |
|-------|--|
| 1 | PN Junction Diode Characteristics (Forward bias, Reverse bias) |
| 2 | Zener Diode Characteristics. |
| 3 | Transistor CE Characteristics (Input and Output). |
| 4 | Rectifier without Filters (Full wave & Half wave) |
| 5 | Rectifier with Filters (Full wave & half wave) |

LAB SCHEDULE

| S.No | Experiment | Date of Completion |
|--|---|--------------------|
| 1 | Verification of Superposition and Reciprocity theorems. | |
| 2 | Verification of Maximum Power Transfer Theorem. Verification on DC with Resistive load. | |
| 3 | Experimental Determination of Thevenin's theorem. | |
| 4 | Swinburne's Test on DC Shunt Machine (Predetermination of efficiency of a given DC shunt machine working as motor and generator). | |
| 5 | OC & SC tests on Singles-phase Transformer (Predetermination of efficiency and regulation at given power factors). | |
| 6 | PN Junction Diode Characteristics (Forward bias, Reverse bias) | |
| 7 | Zener Diode Characteristics. | |
| 8 | Transistor CE Characteristics (Input and Output). | |
| 9 | Rectifier without Filters (Full wave & Half wave) | |
| 10 | Rectifier with Filters (Full wave & half wave) | |
| ADDITIONAL RESEARCH EXPERIMENTS | | |
| 11 | Norton's Theorem | |
| 12 | Led Flasher using an Oscillator | |

EXPERIMENT: SUPERPOSITION THEOREM & RECIPROCITY THEOREM:

OBJECTIVE: To verify Superposition theorem

PREREQUISITES

Basic circuit building, Understanding the meaning of voltage and current, Knowledge of nodal and mesh analysis.

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications.
- c. Experimental determination of voltage and current in the circuit.
- d. Verifying the practical values by comparing with theoretical calculations

APPLICATIONS

1. Current and Voltage values in closed loop circuits.
2. Describe the relation of values of currents that flow through a junction point, in an electrical circuit.
3. Describe the relation of values of voltages in an electrical circuit loop, in an electrical circuit.
4. Calculation of component values in an electric circuit.

EXPERIMENT: MAXIMUM POWER TRANSFER THEOREM

OBJECTIVE:

To verify the maximum power transfer theorem for D.C network considering the resistive load.

PREREQUISITES

Basic circuit building, Understanding the meaning of power, voltage and current

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications.
- c. Experimental determination of voltage and current in the circuit.
- d. Verifying the practical values by comparing with theoretical calculations

APPLICATIONS

1. power values in closed loop circuits.
2. to describe the relation of values of currents that flow through a junction point, in an electrical circuit.

3. to describe the relation of values of voltages in an electrical circuit loop, in an electrical circuit.

4. to calculate power values in an electric circuit.

EXPERIMENT: THEVENIN'S THEOREM

OBJECTIVE:

To find the Thevenin's Voltage and Thevenin's Resistance for the given network.

PREREQUISITES

Basic circuit building, Understanding the meaning of power voltage and current

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications.
- c. Experimental determination of voltage and current in the circuit.
- d. Verifying the practical values by comparing with theoretical calculations

APPLICATIONS

1. Current and Voltage values in closed loop circuits.
2. Describe the relation of values of currents that flow through a junction point, in an electrical circuit.
3. Describe the relation of values of voltages in an electrical circuit loop, in an electrical circuit.
4. Calculation of component values in an electric circuit.

EXPERIMENT: Swinburne's Test on DC shunt machine.

OBJECTIVE

Determination of critical field resistance and critical speed of a given DC shunt generator by obtaining Swinburne's test of a given machine practically.

PREREQUISITES

Basic laws of electromagnetism, principle of operation, types of DC generators and Motors. Basic knowledge about 3-point starter, field and armature speed control of DC motor, OCC characteristics, procedure for drawing OCC characteristics.

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications
- c. Experimental determination of magnetization characteristics.
- d. Graphical determination of critical field resistance and critical speed

APPLICATIONS

1. Design of DC generator
2. Design of excitation system for alternator.
3. Voltage stabilizer
4. Synchronous condenser.

EXPERIMENT: OC and SC tests on Single Phase Transformer

OBJECTIVE

Practical determination of single phase transformer (OC and SC Tests)

PREREQUISITES

Knowledge of the Principle of operation of transformer, the Procedure for conducting OC and SC tests and theoretical knowledge on calculations of losses and efficiency.

DESCRIPTION

- a. Introduction to experiment – 30 min
- b. Connection of experiment and its verification.
- c. Experimental determination of external and internal characteristics.

APPLICATIONS

1. Determination of efficiency of transformer.
2. Determination of regulation of transformer

EXPERIMENT: PN Junction Diode Characteristics (Forward bias, Reverse bias)

OBJECTIVE

To perform PN Junction Diode Characteristics under forward and reverse bias Conditions and also calculating static & dynamic resistances.

PREREQUISITES

Knowledge of semiconductors, PN junction diode operation and ohms law.

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications
- c. Experimental determination of PN Junction Diode characteristics.
- d. Graphical determination of Static resistance and Dynamic resistance

APPLICATIONS

1. PN Junction Diode can be used in Rectifiers
2. Design of Clipper Circuits
3. Design of Clamper Circuits
4. Used as Switch

EXPERIMENT: Zener Diode Characteristics

OBJECTIVE

To perform Zener Diode Characteristics under forward and Reverse bias Conditions and also Calculating Forward bias Resistance.

PREREQUISITES

Knowledge of the Principle of operation of Zener diode and its characteristics.

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications
- c. Experimental determination of Zener Diode characteristics.
- d. Graphical determination of Cut in Voltage and Break down Voltage

APPLICATIONS

1. Zener Diode is used as a regulating element in Voltage Regulators.
2. Used in various protection circuits.
3. Zener Limiters.

EXPERIMENT: Half wave Rectifier with and without filters

OBJECTIVE

Study of Half Wave Rectifier with and without capacitance filters and Determination of Regulation, Efficiency and Ripple factor.

PREREQUISITES

Basic Diode operation, Theory of Half Wave Rectifiers and Knowledge of the capacitance Filter.

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications
- c. Experimental determination of Half Wave Rectifier with & without filters.
- d. Comparison of Practical and Theoretical calculations of Regulation, Efficiency and Ripple factor.

APPLICATIONS

1. Used in Power Circuits to convert ac to dc.
2. Power Supply Circuits.

EXPERIMENT: Full wave Rectifier with and without filters

OBJECTIVE

Study of Full Wave Rectifier with and without capacitance filters and Determination of Regulation, Efficiency and Ripple factor.

PREREQUISITES

Knowledge of Basic Diode operation, Theory of Full Wave Rectifiers and Knowledge of the capacitance Filter.

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications
- c. Experimental determination of Full Wave Rectifier with & without filters.
- d. Comparison of Practical and Theoretical calculations of Regulation, Efficiency and Ripple factor.

APPLICATIONS

1. Used in Power Circuits to convert ac to dc.
2. Power Supply circuits.

EXPERIMENT: Transistor CE Characteristics**OBJECTIVE**

Determination of the transistor characteristics when it is operated in Common Emitter Configuration

PREREQUISITES

Knowledge of the Working principle of Transistor and common emitter configuration characteristics.

DESCRIPTION

- a. Introduction to experiment -30 min
- b. Connection of experiment and its verifications
- c. Experimental determination of Transistor characteristics in CE configuration.
- d. Graphical determination of Common Emitter Input and Output Characteristic.

APPLICATIONS

1. CE Configuration used in Audio Frequency Applications.
2. Coupling Device between various transistor stages.
3. Used as Amplifier

5 (A).LAB SCHEDULE

The lab schedule should be planned once in a week. The week wise scheduled experiment should be completed.

CYCLE 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 |
|---------|--------|--------|--------|--------|--------|--------|--------|
| B1 | Demo | Exp.1 | Exp.2 | Exp.10 | Exp.9 | Exp.8 | test |
| B2 | Demo | Exp.2 | Exp.10 | Exp.9 | Exp.8 | Exp.1 | test |
| B3 | Demo | Exp.10 | Exp.9 | Exp.8 | Exp.1 | Exp.2 | test |
| B4 | Demo | Exp.9 | Exp.8 | Exp.1 | Exp.2 | Exp.10 | test |
| B5 | Demo | Exp.8 | Exp.1 | Exp.2 | Exp.10 | Exp.9 | test |

CYCLE 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 |
|---------|--------|--------|--------|--------|--------|--------|--------|
| B1 | Exp.3 | Exp.4 | Exp.6 | Exp.11 | Exp.12 | Exp.5 | test |
| B2 | Exp.7 | Exp.6 | Exp.11 | Exp.12 | Exp.5 | Exp.4 | test |
| B3 | Exp.3 | Exp.11 | Exp.12 | Exp.5 | Exp.4 | Exp.6 | test |
| B4 | Exp.10 | Exp.12 | Exp.5 | Exp.4 | Exp.6 | Exp.11 | test |
| B5 | Exp.9 | Exp.5 | Exp.4 | Exp.6 | Exp.11 | Exp.12 | test |

(B) VIVA SCHEDULE:

The viva schedule should be planned prior to the lab experiment.

ROUND – 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 |
|----------|--------|--------|--------|--------|--------|
| B1,B2,B3 | Viva | | | | |
| B4,B5,B6 | | viva | | | |
| B1,B2,B3 | | | viva | | |
| B4,B5,B6 | | | | viva | |
| B1 to B6 | | | | | viva |

ROUND - 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 |
|---------|--------|--------|--------|--------|--------|
| SG1 | Viva | | | | |
| SG2 | | viva | | | |
| SG3 | | | viva | | |
| SG4 | | | | viva | |
| SG5 | | | | | viva |

*SG: Selected Group with a maximum of 6 or 12 students

(C) Scheme of Evaluation:

The scheme of evaluation for internal and external exams as follows:

| LAB INTERNAL: | | | | | | | |
|---------------------------------|---------------------------------|--------------------------------------|----------------|----------------|-------------------------|--------------------------------|----------------|
| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-4 | Marks-3 | Marks-3 |
| Total Marks-25 | | | | | | | |

| LAB EXTERNAL: | | | |
|-----------------------|---|--|---|
| S.NO | Write up | Final Evaluation | Viva |
| 1 | 1. Aim 2. Procedure 3. Program 4. Expected output. | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

6. SUGGESTED BOOKS

1. Electrical Machines, I.J. Nagrath and D. P.Kothari, TMH publishers, 3rd publishers
2. Electro Mechanics-I(D.C.Machines), S. Kamakshaiyah, Right Publishers
3. Electrical Machines, P.S Bhimbra, Khanna publications.
4. Fundamentals of Electronic Devices and Circuits, David A. Bell 5th Ed.
5. Electronic Devices and Circuits – K. Lal Kishore, 2ed. 2005, BSP
6. Millman’s Electronic Devices and Circuits – J.Millman, C.C Halkias and Satyabrata , 2ed. 1998, TMH.

7. USEFUL WEBSITES

- <http://www.svecw.edu.in/docs/eeemlab.pdf>
- <http://iitg.vlab.co.in/?sub=61&brch=168&sim=913&cnt=1698>
- www.vimicrosystems.com/vi/Web.../2_Electrical%20Machines%20lab
- <http://www.slideshare.net/sai55chaitanya/electrical-machines-2-lab-manual>
- <http://www.sjcetpalai.ac.in/eee-labs>
- http://www.squ.edu.om/Portals/67/Form&Downloads/ECCE4356_Lab_Manual.pdf
- <http://www.nepindia.com/>
- <http://electricallabs.lakeheadu.ca/yeartwo/eng2258/man2258.pdf>

8. EXPERTS DETAILS

ELECTRICAL

The expert details which have been mentioned below are only a few of the eminent one known Internationally, Nationally and Locally.

INTERNATIONAL

1. Mr. Clayton R Paul, BS, MS, PhD.
Professor of Electrical and computer engineering,
School of Engineering,
Mercer University, Macom, Georgia-31207,
www.faculty.mercer.edu.paul_cr.
2. Dr. Edward Wai-chau Lo, M.Phil.
Honorary Associate Professor
University of Hong Kong
E-mail: ewclo@poly.edu.hk

NATIONAL

1. Prof. D.P. Kothari
Dy. Director (Admin.), IIT - Delhi
Hauzkhas, New Delhi - 110016
E-mail: dpko71@yahoo.com / dkothari@ces.iitd.ernet.in
2. Dr. Sivaji Chakravorti
Professor, EEE Department
Jadavpur University
Kolkatta - 700032, India
E-mail: sivaji@dvu.a.c.in / s_chakravorti@ieee.org

REGIONAL

1. Prof. Dhanvanthri
Head of EEE Department
CVR Engineering College, Hyderabad
2. Prof. A.D. Rajkumar
Electrical Engineering Dept.
CVR Engineering College, Hyderabad

ELECTRONICS

INTERNATIONAL

1. J. Millman ph.d and C.C.Halkias ph.d-
Associate Professor of Electrical Engineering,
Columbia University
2. Prof. Trevor J.Trarnton,
Director of Center for Solid State Electronics Research,
Arizona State University, Tempe, USA
Email:khan.tarik@asu.edu,

NATIONAL

1. Dr. Balasubramanyam S K – Professor Department of Electronics Engineering IIT(BHU), Varanasi Email:bala@bhu.ac.in
2. Dr.Banerjee Swapna, Professor in Electronics & EC engineering, IIT Khragpur Email: swapna [at] ece.iitkgp.ernet.in

REGIONAL

1. P. John Paul, Dean Academics in Gurunanak Engineering College,Hyderabad
2. K.Lal Kishore, Vice-Chancellor, JNTU Anantapur.
E-mail: vc@jntua.ac.in

9 MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D

The Machine tools lab course should be designed in such a way that it should meet the requirements of research and development as well as consultancy projects. Also the Proposals of Project/R&D/Consultancy are as follows:

Proposal 1: Project Design & Execution

Proposal 2: R& D Level Project Design & Execution

Proposal 3: Consultancy Task / Project Design & Development

PROPOSAL FOR R & D ACTIVITY:

- 1.An exact paper from a National/International journal in this entitled area/subject/area (IEEE Format) AND/OR
- 2.An article/white paper from a magazine /journal/weekly/any periodical in the entitled Subject AND/OR
- 3.An Advanced technology development/ proposal/article publication from any source of information.

EXACT PAPER FROM A NATIONAL/INTERNATIONAL JOURNAL

- **Breakdown Characteristics of a Zener Diode with n+p+n-p+ Structure**

Breakdown Characteristics of a Zener Diode with n+p+n-p+ Structure

S. W. SONG, S. K. CHUNG* and Y. I. CHOI

Department of Electronics Engineering, Ajou University, Suwon 442-749

Breakdown characteristics of a Zener diode with n+p+n-p+ structure are analyzed with two-dimensional device simulation. Effects of base doping concentration and profile on the breakdown are presented. An analytical expression of a maximum base doping level for the punch-through breakdown is derived. The diode with a linearly graded base doping shows superior leakage current and capacitance is satisfactory for applications for low-voltage circuits.

PACS numbers: 85.30.Da

Keywords: Punch-through breakdown, Zener diode, Four layer structure

I. INTRODUCTION

With power supply voltage rapidly dropping, a transient voltage suppressor (TVS) operating at low voltage is needed to protect VLSI (Very Large Scale Integration) circuits from voltage spikes originating from lightning, inductive load switching, and ESD (Electro Static Discharge).

Conventional Zener diodes are designed to breakdown at a well-controlled voltage [1], requiring high doping concentrations to achieve a low breakdown voltage, which cause, however, both leakage current and capacitance to increase dramatically. Punch-through diodes utilizing punch-through breakdown of a bipolar junction transistor with light base doping have been studied for high-power and high voltage TVS (Transient Voltage Suppressor) [2,3]. The punch-through breakdown occurs at a lower voltage than the avalanche breakdown voltage. The avalanche breakdown voltage imposes an upper limit on the reverse bias for most diode, bipolar transistor, MOFET (Metal-Oxide Semiconductor Field-Effect Transistor) [4,5], and IGBT (Insulated Gate Bipolar Transistor) [6]. The punch-through diode with n+p+n-p+ structure has been reported for much lower leakage current and capacitance than the Zener diode [1]. However, the punch-through condition for the base doping concentration of the diode was not given. The limit of the base doping level of the punch-through diode is found to be of critical importance for the breakdown voltage.

In this work, we have shown breakdown characteristics as a function of the base doping level as well as the base doping profile and developed an analytical expression for the limit of the base doping concentration for the punch-through breakdown. Two-dimensional numerical simulations using ATLAS have been carried out. The results are shown to support the analytical results.

II. SIMULATION RESULTS AND DISCUSSION

Fig. 1 shows the doping profile of the diode with n+p+n-p+ structure used in simulation. Thicknesses of the p+ and p-layer are fixed as $W_{p+} = 1 \mu\text{m}$ and $W_p = 0.1 \mu\text{m}$. A thinner base is preferred to minimize the punch-through voltage.

The peak doping concentration for p+ layer is varied from 10^{15} cm^{-3} to 10^{17} cm^{-3} whereas the doping level for the p-layer is fixed at 10^{14} cm^{-3} or 10^{15} cm^{-3} . Both the uniform and linearly graded base doping are considered. A p+n+n-p+ structure with the same doping profile and thickness is also investigated.

The breakdown characteristics are shown in Fig. 2 for the proposed structure with a uniform base doping concentration of 10^{16} cm^{-3} , for a p-i-n diode with the concentration of 10^{19} cm^{-3} for both p+ and n+ regions and of 10^{14} cm^{-3} with the i-region, and for a Zener diode with the concentrations of 10^{15} cm^{-3} for both the p+ and n+ region. The Zener diode shows much higher

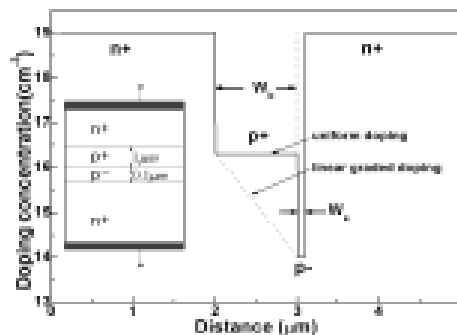


Fig. 1. Cross-sectional view and doping profile of the Zener diode with four layer structure.

*E-mail: skchung@madang.ajou.ac.kr

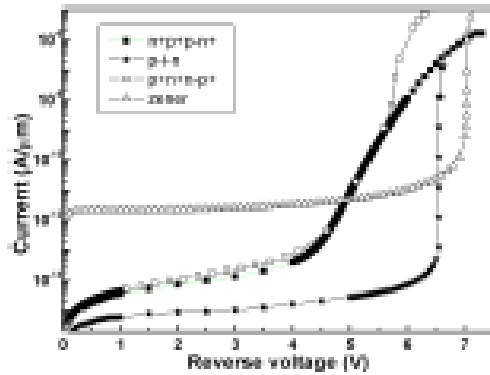


Fig. 2. The simulation results of I-V characteristics.

leakage current and the p-i-n diode shows a higher breakdown voltage to be an attractive low-voltage TVS. A lower breakdown voltage is hardly constructed with the p-i-n structure. Compared with a n+p+p-n+ structure, the p+n+n-p+ structure shows larger clamping voltage defined as the voltage at peak surge current because of lower hole mobility [1]. Fig. 3 shows the I-V curves for the diode with a n+p+p-n+ structure with the constant base doping concentration as a parameter. The n+p+p-n+ structure diode shows a weak snapback effect as doping concentration in the the p+ layer increases as reported [1]. The snapback effect allows reduction of the clamping voltage [1]. Electric field profile for the four layer structure is shown in Fig. 4 for a constant base doping concentration of 10^{16} cm^{-3} when the base is punched through and of 10^{17} cm^{-3} before the punch-through where the avalanche breakdown dominates.

The electric field distribution can be expressed by

$$\begin{aligned} E(x) &= E_c & \text{for } 0 \leq x \leq W_a \\ E(x) &= E_c - (qN_b/\epsilon_s)(x - W_a) & \text{for } x \geq W \end{aligned} \quad (1)$$

where N_b represents the uniform doping concentration of the base and ϵ_s is the dielectric constant for silicon.

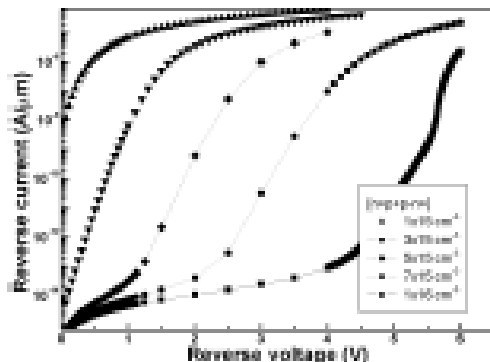


Fig. 3. The I-V curves of n+p+p-n+ structure with the constant base doping concentration as a parameter.

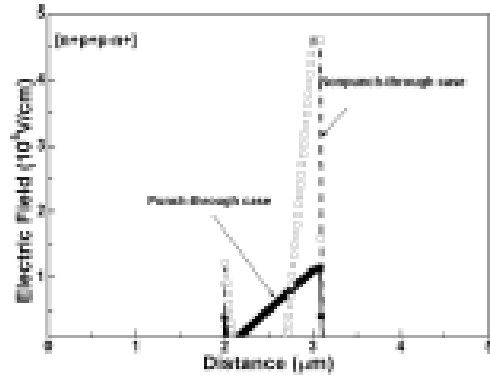


Fig. 4. Electric field distributions of the Zener diode at breakdown.

Using the avalanche breakdown condition for silicon [7] given by $A \int_0^W E(x)^2 dx = 1$ with $A = 1.8 \times 10^{-25}$ and W for the depletion width at which $E(W)=0$ holds, the breakdown condition for the layer structure is found as

$$(E_c)^2 [W_a + 0.125 E_c / (qN_b/\epsilon_s)] = 1/A \quad (2)$$

Equation (2) yields $E_c = 4.6 \times 10^5 \text{ V/cm}$ for $W_a = 0.1 \mu\text{m}$ and $N_b = 10^{17} \text{ cm}^{-3}$, which, in turn, gives $W = 0.4 \mu\text{m}$ determined from (1) with $E(W)=0$, whereas the punch-through for the base of the diode occurs at $E_c = (qN_b/\epsilon_s)W_b$, giving $E_c = 1.23 \times 10^5 \text{ V/cm}$ for $N_b = 10^{16} \text{ cm}^{-3}$ and $W_b = 0.8 \mu\text{m}$, which are in excellent agreement with the simulation results as shown in Fig. 4. Assuming that the avalanche breakdown dominates still in the case, equation (2) will reduce to

$$(E_c)^2 [W_a + 0.125W_b] = 1/A$$

The maximum base doping concentration for the punch-through breakdown is thus determined by

$$N_{bmax} = (\epsilon_s/qW_b) \times 10^5 / [1.8(W_a + 0.125W_b)]^{1/2} \quad (3)$$

Equation (3) gives $N_{bmax} \approx 2.8 \times 10^{16} \text{ cm}^{-3}$ for $W_b = 1 \mu\text{m}$ corresponding to $E_c = 4.24 \times 10^5 \text{ V/cm}$. The results are in fair agreement with the simulation results for non-punch-through case as shown in Fig. 4. Figure 5 shows variation of the breakdown voltage at two different current levels of 10 mA and 1 A as a function of the base doping concentration. When the breakdown voltage curves at two current levels intersect the breakdown curve is a straight vertical line between the two current levels at the corresponding base doping densities. The breakdown voltage for the n+p+p-n+ structure is always smaller than that for the p+n+n-p+ structure. The breakdown voltage remains almost unchanged for the base doping level beyond $N_{bmax} = 2.8 \times 10^{16} \text{ cm}^{-3}$, where the avalanche breakdown dominates as described earlier.

Figure 6 shows effects of the base doping profile on the breakdown characteristics when the peak doping level of

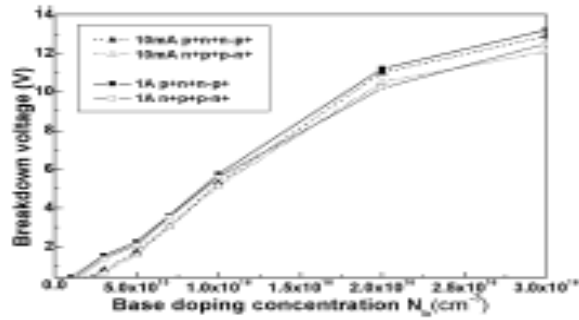


Fig. 5. Breakdown voltage as a function of the base doping concentration.

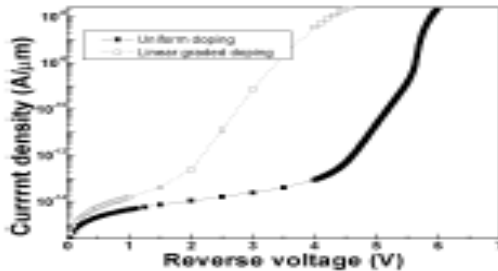


Fig. 6. Doping profile dependent I-V characteristics.

the base is fixed at 10^{16} cm^{-3} . The diode with a linearly graded base doping shows a much lower breakdown voltage than that for a uniform profile. The electric field distribution for the linearly graded base doping is given by

$$E(x) = E_e - [qN_b(W_b)/2\epsilon_s][(x - W_e)^2/W_b + 2R(x - W_e)] \quad (4)$$

for $x \geq W_e$

where $R = N_b(0)/N_b(W_b)$ is introduced. Equation (4) allows the punch-through voltage, $V_{PT,U}$

$$V_{PT,U} = [qN_b(W_b)2\epsilon_s][W_b^2(R + 2/3) + RW_e^2] \simeq [qN_b(W_b)/3\epsilon_s]W_b^2 \quad (5)$$

The punch-through voltage, $V_{PT,U}$ for the uniform base doping is obtained from (1) as

$$V_{PT,U} = (qN_b/2\epsilon_s)(W_b^2 + RW_e^2) \simeq (qN_b/2\epsilon_s)(W_b^2) \quad (6)$$

which yields 4.9 V whereas E_q (5) gives 3.26 V for the linearly graded base doping for the present structure when $W_b = 0.8 \mu\text{m}$ is taken to allow $0.2 \mu\text{m}$ for the depletion layer width of the forward biased junction. Thus, the diode with the linearly graded base is preferred over that with a uniform base doping for a low-voltage TVS.

III. CONCLUSION

We have shown that a Zener diode with an n+p+p-n-p+ structure has superior performance compared with both the p+n+n-p+ structure and the conventional one with a highly doped p+n+ structure. The diode with a linearly graded base doping profile is preferred over that with a uniform base doping in application for the low-voltage TVS devices. An analytical expression of the maximum base doping density for the punch-through breakdown presented in this work will be useful in the diode design for application for low-voltage electronics

ACKNOWLEDGMENTS

This work was supported by the Brain Korea 21 Project. The authors would like to thank Ajou University for providing the research equipment.

REFERENCES

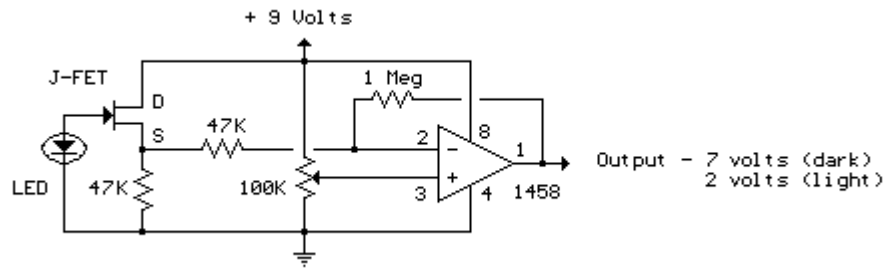
- [1] Y. King, B. Yu, J. Pohlman and C. Hu, *IEEE Electron Device Lett.* **16**, 303 (1995).
- [2] O. M. Clark, *IEEE Trans. Power Electron.* **5**, 20 (1990).
- [3] P. J. Kavan, *IEEE Trans. Electron Devices* **23**, 879 (1976).
- [4] T. M Roh, *J. Korean Phys. Soc.* **37**, 889 (2000).
- [5] S. H. Kim, I. Y. Park and Y. I. Choi and S. K. Chung, *J. Korean Phys. Soc.* **39**, 839 (2001).
- [6] M. S. Kim, Y. I. Choi and S. K. Chung, *J. Korean Phys. Soc.* **39**, 874 (2001).
- [7] W. Palop, *Solid-State Electron.* **10**, 39 (1967).

9. PROPOSAL FOR PROJECT ACTIVITY

A Proposal of a hobby/mini/proto/general/model/proto type project with extended abstract, Block Diagram/Circuit/Flow diagram and clear references may be presented and executed.

ABSTRACT ON HOBBY PROJECT for LED PHOTO SENSOR

Here's a circuit that takes advantage of the photo-voltaic voltage of an ordinary LED. The LED voltage is buffered by a junction FET transistor and then applied to the inverting input of an op-amp with a gain of about 20. This produces a change of about 5 volts at the output from darkness to bright light. The 100K potentiometer can be set so that the output is around 7 volts in darkness and falls to about 2 volts in bright light.



PROPOSAL FOR CONSULTANCY

OBJECTIVE: A programme/machine/product of utility may be proposed to develop for in house usage/ Industrial requirements may be useful for any outside agency that can be marketable in order to generate revenue through consultancy.

TESTING OF 2KVA 220V/220V 1-PHASE TRANSFORMER FOR EFFICIENCY

The physical basis of the transformer is mutual induction between two circuits linked by a common magnetic field. A Transformer is required to pass electrical energy from one circuit to another, via the medium of the pulsating magnetic field, as efficiently and economically as possible. This could be achieved using either iron or steel which serves as a good permeable path for the mutual magnetic flux. An elementary linked circuit is shown in Fig.1. The principle of operation of this circuit can be explained as follows:

Let an alternating voltage v_1 be applied to a primary coil of N_1 turns linking a suitable iron core. A current flows in the coil, establishing a flux Φ_p in the core. This flux induces an emf e_1 in the coil to counterbalance the applied voltage v_1 . This e.m.f. is

$$e_1 = N_1 \frac{d\phi_p}{dt}.$$

Assuming sinusoidal time variation of the flux, let $\phi_p = \Phi_m \sin \omega t$. Then,

$$e_1 = N_1 \omega \Phi_m \cos \omega t, \quad \text{where} \quad \omega = 2\pi F$$

$$E_1 = 4.44 F N_1 \Phi_m$$

$$E_2 = 4.44 F N_2 \Phi'_m$$

From transformation ratio

$$\frac{e_1}{e_2} = \frac{\overline{E_2}}{E_1} = \frac{N_2}{N_1}$$

FUNDED/UNFUNDED PROPOSALS (if any):

OBJECTIVE:

The proposals for AICTE grants like (SDPs, RPS and MODROBES etc) UGC grants, DST CPRI and other funding agencies by giving Title and abstract/objective OR Self Funded program proposals may be submitted for Management approvals.

PROPOSALS (WEEK WISE INDUSTRIAL VISITS), (IN HOUSE OR OUTSIDE VISIT) / TRAININGPROGRAMMES:

TABLE 1 : INDUSTRIAL VISITS

| S.No | Type of industry | Nature of industry | Date of visit | No. of students participated | Year/branch | remark |
|------|------------------|--------------------|---------------|------------------------------|-------------|--------|
| | | | | | | |

11. GUIDELINES FOR SHADOW ENGINEERING (VIP)

INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM):

OBJECTIVES OF SHADOW ENGINEERING:

1. This program uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

TABLE 2: INDUSTRIAL TRAINING (Shadow Engg)

(Career Visit Approval)

| S.No | Name of the Course | Nature of industry | Duration of Training | Authority | Date of Training/Certificate No. | remarks |
|------|--------------------|--------------------|----------------------|-----------|----------------------------------|---------|
| | | | | | | |

12.CALIBRATION/INSTALLATION AND TESTING:

Calibration: Aim of this concept is to check:

- i. whether all the equipment is functioning correctly as per the standards
- ii. To bring correctness in the errors of instrument or equipment.
- iii. To rectify the errors if any

Installation: Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment

Testing: Aim of this concept is to test the equipment after installation whether it meets the existing standards.

The list of equipments (hardware/software):

Necessity of tools for development and testing:

Equipment to be calibrated:

Installation of supporting equipment if any:

PROCEDURE FOR CALIBRATION:

Any Equipment or Instrument or Gauge or Machine can be calibrated as per the standard guidelines mentioned below:

1. Identify the Equipment/Instrument/Gauge/Machine which is defective or to be calibrated or needs correction for an error.
2. Identify the type of error and estimate its frequency of variation.
3. Check with Master Standards or equipment/instrument/machine which is working correctly and meeting our requirements.
4. Estimate the frequency of deviations from normal mode.
5. If the equipment is under warranty, then inform to concerned supplier or agency that will carry out calibration.
6. If the equipment is out of warranty then we can compare the deviations and set the error rectification.
7. Generally as per the procedure, the equipment or instruments can be calibrated by the agencies and issue calibration certificate which consists of date of calibration, calibration next due date and remarks as mentioned in the following format.
8. Record and keep all the calibration certificates in safe custody.

After calibration the details of equipment should be submitted in following format.

| S.No | Type of equipment | Certificate no | Certificate issued by | Date of calibration | Date of calibration due | Remarks |
|------|---|----------------|-------------------------------|---------------------|-------------------------|---------|
| 1 | FT-3254 DC Ammeter (0-20)mA Digital type | 316 | Future Tech instruments P LTD | | | |
| 2 | FT-3264 Digital AC Ammeter (0-100)mA | 316 | Future Tech instruments P LTD | | | |
| 3 | FT-3213 DC Voltmeter (0-20)v Digital type | 316 | Future Tech instruments P LTD | | | |
| 4 | FT-3254 DC Ammeter (0-20)mA Digital type | 316 | Future Tech instruments P LTD | | | |
| 5 | FT-3264 Digital AC Ammeter (0-100)mA | 316 | Future Tech instruments P LTD | | | |

iv. Calibration, Testing and Installation details equipment wise are mentioned as follows:

Case 1: Calibration of Equipment ----- if any

Case 2: Installation of Equipment ----- if any

Case 3: Testing of Equipment ----- if any

Presently there is no new equipment is present for either testing or installations.

13.PREVENTIVE MAINTENANCE SCHEDULES:

MAINTENANCE:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

| S.No | Type of equipment | Date of Maintenance | Type of Activity | Remarks |
|------|---|---------------------|---------------------------------------|----------------------------------|
| 1 | Experimental Setup for Magnetization ckts of Dc Shunt Gen. for Science lab. | | Finding magnetization characteristics | Working in good condition |
| 2 | Experimental Setup for Swinburne's test on DC Shunt M/C for Science lab. | | Finding the efficiency | Working in good condition |
| 3 | Experimental Setup for brake test on DC Shunt M/r for Science lab. | | Finding the efficiency | Working in good condition |
| 4 | Oc & SC test on 1Ph.T/F for science lab | | For sc and oc test | Working in good condition |
| 5 | Experimental Setup for brake test on 3Ph. Indution M/r for Science lab. | | Finding the efficiency | Working in good condition |
| 6 | Experimental Setup for Regulation of alternator by Synch. for Science lab. | | Finding the regulation | Working in good condition |
| 7 | Rectifier 100A for science lab. | | To conver DC into AC supply | Working in good condition |
| 8 | 3Ph. Auto T/f 5KVA for science lab | | | Working in good condition |
| 9 | 1Ph. Auto T/f 15A for science lab | | | Working in good condition |
| 10 | DHH Tachometer for science lab | | Find the no. of revolution | Working in good condition |
| 11 | Meters and Instruments for | | Find voltage | Working in good condition |
| | | | | |

| | | | | |
|----|---|--|-----------------------------------|--------------------------------------|
| | science lab PMVM MC 150/300V-5No.s,25/50V PMAM 10/20A,1/2A-5NO's each | | and current | condition |
| 12 | Meters and Instruments for science lab PMAM MI 1A,5/10A- 2 No's each PMVM MI 150/300V, 300/600V-2No's each, 25/50V -1No | | Find voltage and current | Working in good condition |
| 13 | Meters and Instruments for science lab WM LPF 75/150/300V 1/2A,300/600V 5/10A-2No's each | | Find voltage and current | Working in good condition |
| 14 | Meters and Instruments for science lab WM UPF 75/150V 5/10A,300/600V 5/10A-2No's each | | Find voltage and current | Working in good condition |
| 15 | Rheostat 370 Ohm,1.7A | | To control the flow of current | Working in good condition |
| 16 | Rheostat 50 Ohm,5A | | To control the flow of current | Working in good condition |
| 17 | Rheostat 350 Ohm,2A | | To control the flow of current | Working in good condition |
| 19 | FT 2504D Dual (0-30)V/2 amp RPS Digital meters | | Supply the required voltage | Working in good condition |
| 20 | Decade resistances boxes 1Watt | | Variable resistance | Working in good condition |
| 21 | FT-1352 Decade capacitive box | | Variable capacitance | Working in good condition |
| 22 | FT-1342 Decade inductive box | | Variable inductance | Working in good condition |
| 23 | FT-2603 Function generator | | | Working in good condition |
| 24 | FT-3254 DC Ammeter (0-20)mA | | Find the current | |

| | | | | |
|----|--|--|-------------------------|----------------------------------|
| | Digital type | | | |
| 25 | FT-3264 Digital AC Ammeter (0-100)mA | | Find the current | Working in good condition |
| 26 | FT-3213 DC Voltmeter (0-20)v Digital type | | Find the current | Working in good condition |
| 27 | FT-1726 Two port networks parameter | | Find the n/w parameters | Working in good condition |

(2) Maintenance Reports duly signed by in charges as well as HODs and duly approved by Principal periodically.

14.TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S.No., Equipment Name , Type of Problem (Too much Noise, Abnormal Sound, Corrupt Software, Anti Virus Problem, Missing of Display, CRT not working, Motor is not giving signal, Digital display is not working, Break of tools, Misalignment of machine elements, PLC is not properly working), Expected Reasons (Bearing failure, Improper alignment of machine centres, Missing of vibration pads etc)

Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| | Date of recording activity | Equipment Name | Type of Trouble | Remedial Activity | Remarks |
|--|-----------------------------------|-----------------------|------------------------|--------------------------|----------------|
| | | | | | |

OPERATING SYSTEMS LAB

CONTENTS:

- 1. OBJECTIVES AND RELEVANCE**
- 2. SCOPE**
- 3. PREREQUISITES**
- 4. SYLLABUS AS PER JNTUH**
- 5. LAB SCHEDULE**
- 6. SUGGESTED BOOKS**
- 7. WEBSITES (USEFUL LINKS)**
- 8. EXPERT DETAILS**
- 9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D**
- 10. PROPOSALS**
- 11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
- 12. CALIBRATION, TESTING AND INSPECTION**
- 13. PREVENTIVE MAINTENANCE SCHEDULES**
- 14. TROUBLESHOOTING**

1. OBJECTIVES AND RELEVANCE

To provide students with an understanding of the standard problems and their solutions in the area of operating systems (including process management, storage management, I/O systems design, distributed systems, protection and security) and with practical experience with one or more modern operating systems (usually some versions of Unix and Windows).

2. SCOPE

The scope of this subject is to organize and control hardware's and software's operating systems so that the device operates in an adaptable and predictable manner. An operating system gives the capacity to give out a variety of purposes, interact with computer users in a less complicated manner and stay up to date with requirements that keep changing with time. The operating system is also responsible to manage the processor, memory, devices, storage, applications interfaces and user interfaces.

3. PREREQUISITES

This lab recommends complete practice of C Programming & basics of Disk Operating System commands and GUI of window based operating system.

4. JNTUH SYLLABUS

1. Simulate the following CPU scheduling algorithms
 - a) FCFS b) SJF c) Round Robin d) Priority
2. Simulate all file allocation strategies
 - a) Sequential b) Indexed c) Linked
3. Simulate MVT and MFT
4. Simulate all File Organization Techniques
 - a) Single level directory b) Two level c) Hierarchical d) DAG
5. Simulate Bankers Algorithm for Dead Lock Avoidance
6. Simulate Bankers Algorithm for Dead Lock Prevention
7. Simulate all page replacement algorithms
 - a) FIFO b) LRU c) LFU Etc.....
8. Simulate Paging Technique of memory management.

MAIN LINKAGE OF OS THEORY WITH LAB EXPERIMENTS:

UNIT-I

No experiments suggested as per the JNTU syllabus. Basic introduction concepts are discussed.

UNIT –II

EXPERIMENT NO.1

Introduction of CPU scheduling algorithms – FCFS, SJF, Round Robin, Priority.

OBJECTIVE:

The main objective is to understand clearly about all the process scheduling algorithms.

PRE REQUISITES:

Basic scheduling process technique.

DESCRIPTION:

1. Introduction to process techniques.
2. Introduction to scheduling techniques.
3. Implementation of scheduling algorithms.

APPLICATIONS:

1. These mechanisms can be practically applied in various Operating Systems.
2. These techniques are useful to improve the throughput.

UNIT-III

No experiments suggested as per the JNTU syllabus. Discuss synchronization techniques.

UNIT IV:

EXPERIMENT NO. 3, 7 & 8

Introduction of paging techniques.

OBJECTIVE:

The main objective is to understand clearly about all the page replacement algorithms and allocation techniques.

PRE REQUISITES:

Basic paging techniques.

DESCRIPTION:

1. Introduction to paging techniques.
2. Introduction to page replacement and allocation technique.
3. Implementation of page replacement algorithms.

APPLICATIONS:

1. These mechanisms can be practically applied in various secondary storage techniques such as disk storage.

EXPERIMENT NO.8

Simulate Paging Technique of memory management.

OBJECTIVE:

The main objective is to understand clearly about paging technique

PRE REQUISITES:

Basic paging techniques.

DESCRIPTION:

1. Introduction to paging techniques.
2. Implementation of paging technique.

APPLICATIONS:

1. These mechanisms can be practically applied in various secondary storage techniques such as disk storage.

UNIT-V**EXPERIMENT NOS. 5 & 6**

Deadlocks

OBJECTIVE:

The main objective is to understand clearly about Deadlocks technique

PREREQUISITES:

Basic knowledge of Deadlock characteristics.

DESCRIPTION:

1. Introduction to Deadlocks.
2. Deadlock detection
3. Deadlock Avoidance
4. Data Recovery from Deadlocks.
5. Deadlock prevention.

APPLICATION:

1. It gives the correct idea of Deadlocks.
2. It helps the student to know about Deadlock detection, Deadlock avoidance & prevention techniques.

UNIT-VI**EXPERIMENT NOS. 2 & 4****OBJECTIVE:**

The main objective is to understand clearly about File Organization & Implementation techniques.

PREREQUISITES:

Basic knowledge of File Accessing Procedure.

DESCRIPTION:

1. Introduction to File Organization techniques.
2. Introduction to File Implementation techniques.

APPLICATION:

1. It gives the correct idea of File Accessing technique.
2. It gives the complete knowledge to a student about the File Implementation techniques.

LEAD PROGRAM

Do a Lead Experiment on interprocess communication (IPC) using pipe and message queues using LINUX OS.

5 (A) LAB SCHEDULE: CYCLE 1:

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------------|--------|------------|------------|--------|-----------|----------|--------|---------|
| B1(501 to 512) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |
| B2(513 to 524) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |
| B3(525 to 536) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |
| B4(537 to 548) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |
| B5(549 to 560) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |

CYCLE 2:

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------------|--------|------------|------------|--------|-----------|-----------|--------|-----------|
| B6(561 to 572) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5 & 6 | test | Exp.2 & 4 |
| B7(573 to 584) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5 & 6 | test | Exp.2 & 4 |
| B8(585 to 596) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5 & 6 | test | Exp.2 & 4 |
| B9(597 to 5A8) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5 & 6 | test | Exp.2 & 4 |
| B10(5A9 to 5C0) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5 & 6 | test | Exp.2 & 4 |

CYCLE 3:

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------------|--------|------------|------------|--------|-----------|----------|--------|---------|
| B11(5C1 to 5D2) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |
| B12(5D3 to 5E4) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |
| B13(5E5 to 5F6) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |
| B14(5F7 to 5G8) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |
| B15(5G9 to 5H0) | Demo | Exp.1(a,b) | Exp.1(c,d) | Exp.7 | Exp.3 & 8 | Exp.5& 6 | test | Exp.2&4 |

(B) VIVA SCHEDULE: The viva schedule should be planned prior to the lab experiment.

CYCLE 1:

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|
| B1,B2,B3 | viva | | | | | viva | | |
| B4,B5,B1 | | viva | | | | | viva | |
| B2,B3,B4 | | | viva | | | | | viva |
| B5,B1,B2 | | | | viva | | | | |
| B2,B3,B4 | | | | | viva | | | |

CYCLE 2:

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------|--------|--------|--------|--------|--------|--------|--------|--------|
| B6,B7,B8 | viva | | | | | viva | | |
| B9,B10,B6 | | viva | | | | | viva | |
| B7,B8,B9 | | | viva | | | | | viva |
| B10,B6,B7 | | | | viva | | | | viva |
| B8,B9,B10 | | | | | viva | | viva | |

CYCLE 3:

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-------------|--------|--------|--------|--------|--------|--------|--------|--------|
| B11,B12,B13 | viva | | | | | viva | | |
| B14,B15,B11 | | viva | | | | | viva | |
| B12,B13,B14 | | | viva | | | | | viva |
| B15,B11,B12 | | | | viva | | | | viva |
| B13,B14,B15 | | | | | viva | | viva | |

ROUND - 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|
| SG1 | viva | | | | | | | |
| SG2 | | | viva | | | | | |
| SG3 | | | | | viva | | | |
| SG4 | | | | | | | viva | |
| SG5 | | | | | | | | viva |

*SG: Selected Group with a maximum of 6 or 12 students

(C) Scheme of Evaluation:

The scheme of evaluation for internal and external exams as follows:

| <u>LAB INTERNAL:</u> | | | | | | | |
|---------------------------------|---------------------------------|--------------------------------------|----------------|----------------|-------------------------|--------------------------------|----------------|
| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-4 | Marks-3 | Marks-3 |
| Total Marks-25 | | | | | | | |

| <u>LAB EXTERNAL:</u> | | | |
|-----------------------------|---|--|---|
| S.NO | Write up | Final Evaluation | Viva |
| 1 | 5. Aim 6. Procedure 7. Program 8. Expected output. | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

6. SUGGESTED BOOKS:

1. C programming and Data Structures, P. Padmanabham, Third Edition, BS Publications
2. Mastering C, K.R. Venugopal and S.R. Prasad, TMH Publications.
3. C Programming & Data Structures, B.A.Forouzan and R.F. Gilberg, Third Edition, Cengage Learning.
4. C Programming & Data Structures, E. Balagurusamy, TMH.
5. Operating System Concepts- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, JohnWiley.

7. WEBSITES(USEFUL LINKS)

<http://www.rajalakshmi.org/dept/it/cs2257-lm.pdf>

<http://www.mamcet.com/it/e-learning/4sem/oslabmanual.pdf>

<http://www.sriengg.com/wp-content/uploads/2013/12/final-copy-oslab-manual.-iiiiiiiiiii-2.pdf>

<http://www.drnnce.ac.in/files/OS-Lab%20Manual%20nisha.pdf>

http://jwfiles.net/gv6paxaa86uu/oslab_2.doc.html

<http://jwfiles.net/pbcsioo5xyyv/Computer Networks Operating System Lab Manual.pdf.html>

http://jwfiles.net/zv07b56acrqo/cn-os-lab_manual.pdf.html

<http://jwfiles.net/dg7pr06fhisw/Computer Networks Operating System Lab Manual I II B.Tech I Sem - R09 www.jntuworld.com JWFILES.zip.html>

<http://www.jbiet.edu.in/coursefiles/it/CN-OS-LP%20LAB%20MANUAL%20K.ROSHAN.pdf>

8. EXPERTS DETAILS

International

Martin C. Rinard, MIT Computer Science and Artificial Intelligence Laboratory ,The Stata Center,

Building 32-G82832 Vassar Street, Cambridge, MA 02139 Email: rinard@lcs.mit.edu

Abraham Silberschatz , Professor and Chair of Computer Science at Yale University.

Andrew S. Tanenbaum, American computer scientist and professor of computer science, Amsterdam in the Netherlands.

Ruth E. Anderson Department of Computer Science & Engineering University of Washington.

National



1. Matthew Jacob T. Ph.D. (SERC) Professor. Performance aspects of Computer Architecture, Operating Systems, Parallel Processing. Email: mjt@serc.iisc.ernet.in

2. Mrs. A. Kalaivani MCA., M.Phil Asst. Professor, NIITM, NEW DELHI.

REGIONAL

1. G.Venkata Rami Reddy ,Associate professor School of Information Technology , JNT University Hyderabad Email: gvr_reddi@yahoo.co.in

2. Dr. B. Vishnu Vardhan, M.Tech, PhD JNTUH COLLEGE OF ENGINEERING,Email:vishnubulusu@yahoo.com

3. Dr. Sanikommu vasavi, Deptt. of Computer Sc. & Engg. VELAGAPUDI RAMAKRISHNA SIDDHARTHA ENGINEERING COLLEGE E-mail:vasavi_movva@gmail.com

4. K. Priyadarshini, Asst professor, Dept of CSE, CMREC .E-mail: chinnupriya6@gmail.com

9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D:

The OS lab course should be designed in such a way that it should meet the requirements of research and development as well as consultancy projects. Also the Proposals of Project/R&D/Consultancy are as follows:

❖ The Android Application Development College Challenge

Abstract – Since mobile devices have become more and more powerful and distributive, mobile computing has greatly changed our daily life. As one of the most popular mobile operating systems, Android provides the tools and API for Android developer to develop Android applications. The Android Application Development College Challenge is an influential Android developer contest for college students in China. This contest has been held for two times since 2010. It encourages college students to design and implement their applications on the Android platform. In this way, it gives the students an opportunity to show their creativity and learn about the development of Android applications. The influence of the contest is going on increasing, and students from more universities and regions are taking part in the contest. In this paper we introduce an organization of such a contest.

An Advanced technology development/ proposal/article publication from any source of Information

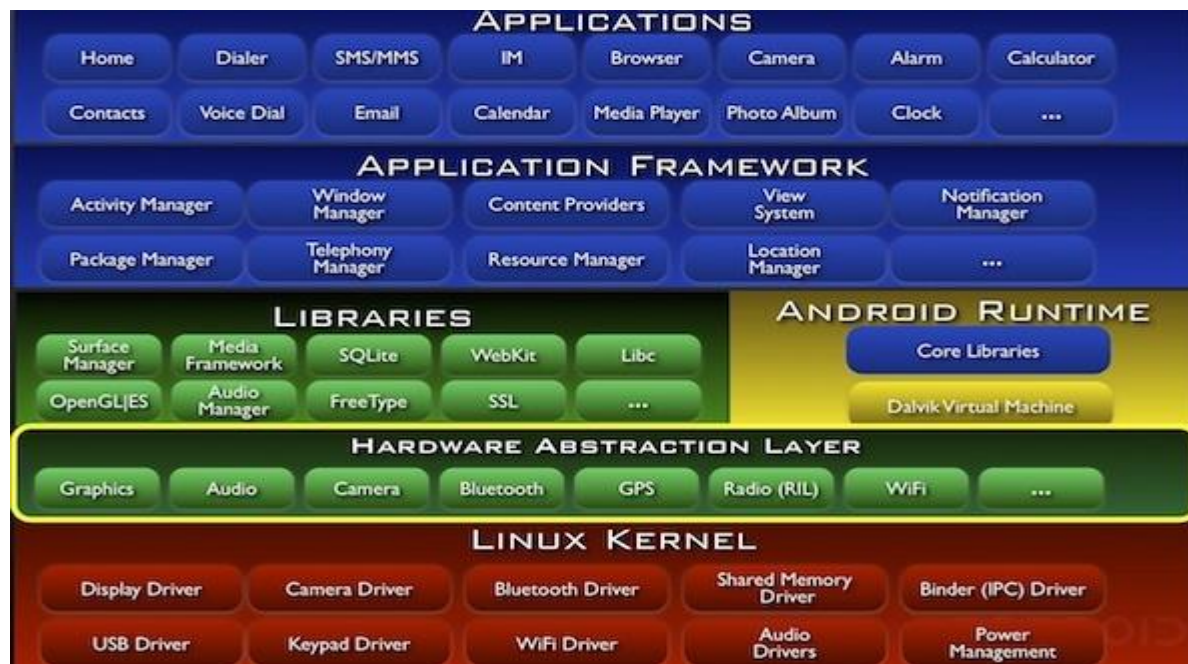
Android software development

Android software development is the process by which new applications are created for the [Android operating system](#). Applications are usually developed in the [Java](#) programming language using the [Android Software Development Kit](#), but other development tools are also available.

What is Android?

- A free, open source mobile platform
- A Linux-based, multiprocess, multithreaded OS
- Android is not a device or a product
- It's not even limited to phones - you could build a DVR, a handheld GPS, an MP3 player, etc.

System architecture



Linux kernel

At the bottom of the layers is Linux - Linux 2.6 with approximately 115 patches. This provides basic system functionality like process management, memory management and device management like camera, keypad, display etc. Also, the kernel handles all the things that Linux is really good at such as networking and a vast array of device drivers, which take the pain out of interfacing to peripheral hardware.

Libraries

On top of Linux kernel there is a set of libraries including open-source Web browser engine WebKit, well known library libc, SQLite database which is a useful repository for storage and sharing of application data, libraries to play and record audio and video, SSL libraries responsible for Internet security etc.

Android Runtime

This is the third section of the architecture and available on the second layer from the bottom. This section provides a key component called **Dalvik Virtual Machine** which is a kind of Java Virtual Machine specially designed and optimized for Android.

The Dalvik VM makes use of Linux core features like memory management and multi-threading, which is intrinsic in the Java language. The Dalvik VM enables every Android application to [run](#) in its own process, with its own instance of the Dalvik virtual machine.

The Android runtime also provides a set of core libraries which enable Android application developers to write Android applications using standard Java programming language.

Application Framework

The Application Framework layer provides many higher-level services to applications in the form of Java classes. Application developers are allowed to make use of these services in their applications.

10 PROPOSAL 1 : FOR PROJECT ACTIVITY :

1.A Proposal of a hobby/mini/proto/general/model/proto type project with extended abstract, Block Diagram/Circuit/Flow diagram and clear references may be presented and executed.

ABSTRACT ON HOBBY PROJECT RELATED TO ANDROID APP

Android Free SMS App

Abstract:

Since mobile devices have become more and more powerful and distributive, mobile computing has greatly changed our daily life. As one of the most popular mobile operating systems, Android provides the tools and API for Android developer to develop Android applications. In this way, it gives the students an opportunity to show their creativity and learn about the development of Android applications. This application is mainly intended to send sms free of cost from Android mobile. Here user can send any no. of sms for free and to n number of users at a time with a maximum character constraint. Here user doesn't need to activate any sms package provided by service provider. And the main advantage is

we can send sms without SIM card using Wi-Fi. As we are sending sms through GPRS or Wi-Fi, there's no need to bother about service provider cost. This application has facilities like sending group sms, customized sms and viewing history.

Proposed System

The beauty of the application is if you are in accessible Wi-Fi zone, then you can connect to Wi-Fi and send sms for free. It will not cost on the present balance. We have powered this application with single and multiple users' sms sending option. The user can see the history of his usage. This application uses SQLite 3.0 as back end to store all sms history without relying on third party tools. The user can create customized groups and can save and send sms (Bulk sms).

PROPOSAL3: FOR CONSULTANCY:

OBJECTIVE: A programme/machine/product of utility may be proposed to develop for in house usage/ Industrial requirements may be useful for any outside agency that can be marketable in order to generate revenue through consultancy

Proposal to develop Android Free SMS App for various communication purposes like Placement & Training, Attendance Status to the parents, results etc.

A proposal is made to develop the Android App Using laboratory Facilities.

Proposed Equipment for usage: Lab Facility with Internet

INTRODUCTION TO THE CONCEPT & SCOPE OF THE ANDROID APP

Proposal to develop Android App: This application is mainly intended to send sms for free of cost from Android mobile. Here user can send any no. of sms for free and to n number of users at a time with a maximum character constraint.

Android App Scope: There are many competitors in this area who are doing research and development on new platforms & user experience. One such technology is Android from Google which supports Google phones. These phones are described as next Generation mobiles [As described by Google]. Developing application for such mobile phones using the open source android SDK is quite interesting. This makes the sms application quite easy, efficient, flexible and economic.

The Android App should have the following properties:

- It can be used as a free source.
- It should be easy to communicate with smart phones.
- Low Cost.

9. PROPOSALS

Relevant IV (Industrial Visit) (VIP= Visit of Industrial Proposals)

Students visit different software development companies like Data analytixds, Red hat Inc, fedora inc.

TABLE 1 : INDUSTRIAL VISITS

| S.no | Type of industry | Nature of industry | Date of visit | No. of students participated | Year/branch | remarks |
|------|------------------|----------------------|---------------|------------------------------|-------------|---------|
| 1 | MNC | S/W Industry Infosys | | 15(70% above) students | III CSE | |
| 2 | MNC | S/W Industry | | 15 | III CSE | |
| 3 | | | | | | |

**TABLE 2: INDUSTRIAL TRAINING (Shadow Engg)
(Career Visit Approval)**

| S.no | Name of the Course | Nature of industry | Duration of Training | Authority | Date of Training/Certificate No. | remarks |
|------|--------------------|--------------------|----------------------|-----------|----------------------------------|---------|
| 1 | Android | COIGN | 2 WEEK | Director | | |
| | | | | | | |

11. GUIDELINES FOR SHADOW ENGINEERING (VIP)

INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM):

OBJECTIVES OF SHADOW ENGINEERING:

1. To uplift the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignments.
4. This program leads to matrixing with the students.

12. Calibration & Testing (Method & Tools to be used)

Calibration: There is no need to calibrate any equipment for this concerned lab.

Installation: Steps for downloading and installation of an **Android File Manager**.

Installation & Testing: Aim of this concept is to make and maintain installation procedure for a new App.

Keywords:

Eclipse: Eclipse is a platform that has been designed from the ground up for building integrated web and application development tooling.

IDE: It is a software application that provides comprehensive facilities to computer programmers for software development.

SDK(S/w Development Kit): It is typically a set of software development tools that allows for the creation of applications for a certain software package

Android Development Tools (ADT): It is a plugin for eclipse that extends the Eclipse IDE by providing more tools to develop Android Applications.

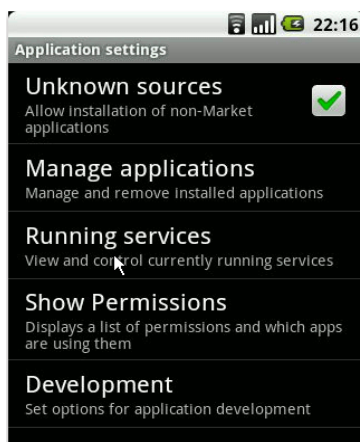
Java SE Development Kit (JDK): A popular Java SDK that is used to program Android applications.

Android Virtual Device (AVD): An Android emulator that allows you to simulate how the application will run on an actual Android device.

Enable Unknown Sources

Before attempting a manual installation of apps using the .apk files, you must first allow your phone to install from “*Unknown Sources*” (i.e. non-Market apps).

To do this, navigate to *Menu -> Settings -> Applications* and check the box marked “*Unknown Sources*”.



1. Installing app Using The Conventional Method

Step 1: Install File Manager

Android does not natively come with any method of browsing the data on your SD card, so you will need to install a file manager from a market. There are a large variety of file managers available on Android, but my personal favourite is [ASTRO File Manager](#).



Step 2: Copy .apk file to SD card

Once you have ASTRO File Manager installed, connect your Android device to your PC using your USB cable. Mount the SD card and copy over the .apk file you would like to install.

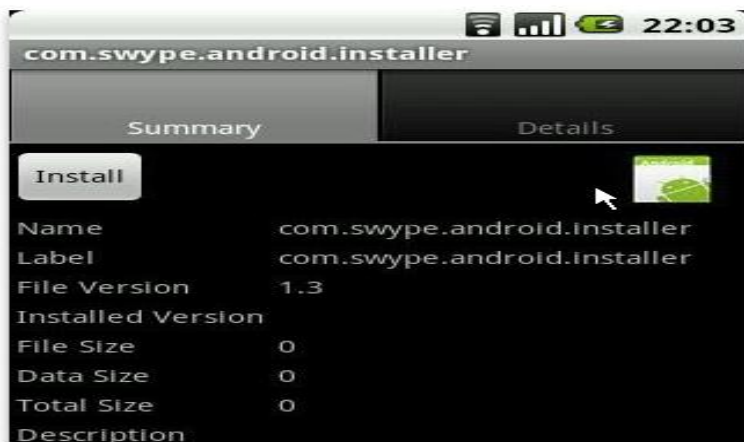
Step 3: Install .apk

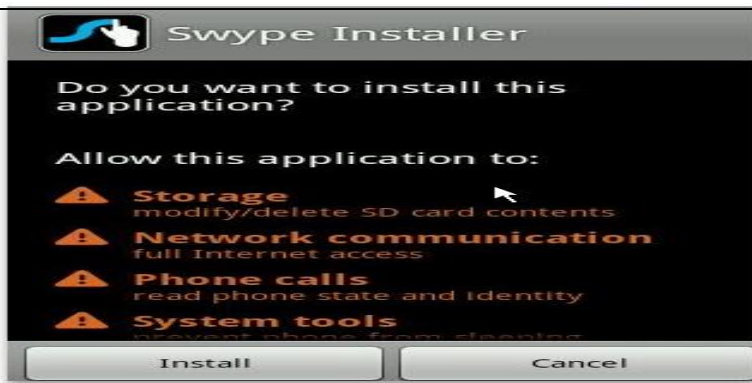
On your Android device, navigate to the .apk file using ASTRO File Manager and select it.

This will open a dialog box allowing you to install the app. Select “*Open App Manager*”.



On the next two pages, select “*Install*” and “*Install*” again to install the .apk.





Your new app is now installed.

2. Installing app using Dropbox

Dropbox is really a versatile app and is a waste if we don't fully utilize it. The method is simple. In your computer, download the apk file to your Dropbox folder. Let it finish syncing. In your phone, open the Dropbox app, navigate to the folder where you keep the apk file, click on it. Dropbox will then download the apk file. The usual installation follows.

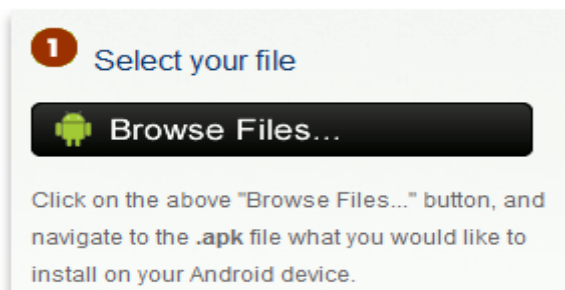
3. Installing app Using the Online Apk Installer

The online apk installer is a web app created by a helpful XDA-Developers forum member [htc-hd2](#), with the aim to make manual installation of an .apk file much easier. It is useful if you want to share an apk file with a friend.

Step 1: Upload File

First, navigate to www.apkinstall.com.

Here you will see a black "Browse Files..." button. Select this and choose the apk file on your PC.



The apk file will remain active on the website for 15 minutes.

Step 2: Scan QR Code

Once you have uploaded the file, a QR code will appear on the website.



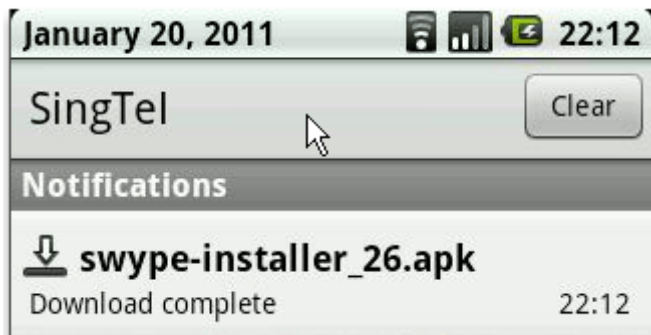
You must scan this QR code with a Barcode Scanner. The website suggests using [Barcode Scanner](#), however I personally prefer using [Google Goggles](#).



Once the QR code has scanned, you can click on the link to download the .apk file straight to your device.

Step 3: Install .apk

After the .apk file is downloaded to your Android device, you can install it by simply clicking on it and navigating through the installation pages.



Testing : Aim of this concept is to test the software after installation whether it meets the existing standards.

Step1. Accessibility is a usability requirement

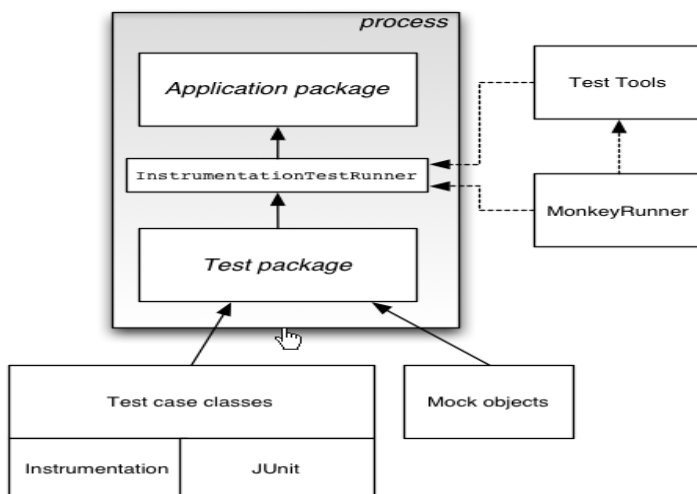
The Android testing framework, an integral part of the development environment, provides the architecture and powerful tools that help you test every aspect of your application at every level from unit to framework.

The testing framework has these key features:

- Android test suites are based on JUnit. You can use plain JUnit to test a class that doesn't call the Android API, or Android's JUnit extensions to test Android components. If you're new to Android testing, you can start with general-purpose test case classes such as Android Test Case and then go on to use more sophisticated classes.
- The Android JUnit extensions provide component-specific test case classes. These classes provide helper methods for creating mock objects and methods that help you control the lifecycle of a component.
- Test suites are contained in test packages that are similar to main application packages, so you don't need to learn a new set of tools or techniques for designing and building tests.
- The SDK tools for building and tests are available in Eclipse with ADT, and also in command-line form for use with other IDEs. These tools get information from the project of the application under test and use this information to automatically create the build files, manifest file, and directory structure for the test package.
- The SDK also provides monkeyrunner, an API for testing devices with Python programs, and UI/Application Exerciser Monkey, a command-line tool for stress-testing UIs by sending pseudo-random events to a device.

This document describes the fundamentals of the Android testing framework, including the structure of tests, the APIs that you use to develop tests, and the tools that you use to run tests and view results. The document assumes you have a basic knowledge of Android application programming and JUnit testing methodology.

The following diagram summarizes the testing framework:



13. PREVENTIVE MAINTAINANCE SCHEDULES

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

| S.No. | Name of the Equipment | Date of Maintenance | Type of Activity | Remarks |
|-------|-----------------------|---------------------|------------------|---------|
| 1 | System | | App Deployment | |
| | | | | |
| | | | | |
| | | | | |

(2) Maintenance Reports duly signed by in charges as well as HODs and duly approved by Principal periodically.

14. TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S/w Name: Android

Type of Problem (Version Compatibility, Proper Accessing from UI, virus problem), Expected Reasons, Trouble Shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Equipment /Software Name | Type of Trouble | Remedial Activity | Remarks |
|-------|----------------------------|--------------------------|-----------------------|-------------------|---------|
| 1 | | Android | Version Compatibility | Updation | |

COMPILER DESIGN LAB

CONTENTS:

- 1. OBJECTIVES AND RELEVANCE**
- 2. SCOPE**
- 3. PREREQUISITES**
- 4. SYLLABUS AS PER JNTUH**
- 5. LAB SCHEDULE**
- 6. SUGGESTED BOOKS**
- 7. WEBSITES (USEFUL LINKS)**
- 8. EXPERT DETAILS**
- 9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D**
- 10. PROPOSALS**
- 11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
- 12. CALIBRATION, TESTING AND INSPECTION**
- 13. PREVENTIVE MAINTENANCE SCHEDULES**
- 14. TROUBLESHOOTING**

1. OBJECTIVES AND RELEVANCE

The main objective of the lab course is to gain practical knowledge of the language translation peculiarities by designing a complete translator for a mini language.

2. SCOPE

The scope of this lab is to make students to acquire knowledge in different phases and passes of compiler, and specifying different types of tokens by lexical analyzer, and also able to use the compiler tools

3. PREREQUISITES

Students should have foundation in formal language .The student should have programming experience (in C/C++).

4. JNTUH Syllabus:

PREAMBLE:

This lab covers the experiments in Compiler Design lab. The JNTU has given 6 experiments in the syllabus out of which first two experiments are based on the lexical Analyzer, following four experiments deals with predictive parser, LALR bottom up parser for given language, and convert the BNF rules into Yacc form, and Program to generate machine code from the abstract syntax tree generated by the parser

1. Design a Lexical analyzer for the above language. The lexical analyzer should ignore redundant spaces, tabs and newlines. It should also ignore comments. Although the syntax specification

(Unit-1)

Objective: The main objective of this program states that identifiers can be arbitrarily long; you may restrict the length to some reasonable value.

Prerequisites: Require knowledge in different phases and passes of Compiler, and specifying different types of tokens by lexical analyzer.

Applications: A lexer forms the first phase of a compiler frontend in modern processing, and is generally done in a single pass.

3. Implement the lexical analyzer using JLex, flex or lex or other lexical analyzer generating tools.

Objective: The main objective of the programs is student able to learn the computer language tools (preprocessor Directives)

Prerequisites: Require knowledge of data structures in compilation, regular expressions and c code called rules

Applications: Flex is a tool for generating scanners. A scanner, sometimes called a tokenizer, is a program which recognizes lexical patterns in text.

3. Design Predictive parser for the given language

Objective: The main objective of the programs is student able to learn the Recursive Predictive Parsing and Non-Recursive predictive parsing

Prerequisites: Require knowledge of constructing LL Parsing table and top-down parsing

Applications: Able to recover errors in predictive parsing.

4. Design LALR bottom up parser for the above language.

Objective: The main objective of the programs is student able to learn the operator precedence parsing algorithm, shift reduce parsing

Prerequisites: Require knowledge of constructing LR Parsing table and Bottom-up parsing

Applications: Able to recover errors phrase-level error recovery and panic mode error recovery in LALR bottom up parsing

5. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree.

6. Write program to generate machine code from the abstract syntax tree generated by the parser. The following instruction set may be considered as target code. The following is a simple register-based machine, supporting a total of 17 instructions. It has three distinct internal storage areas. The first is the set of 8 registers, used by the individual instructions as detailed below, the second is an area used for the storage of variables and the third is an area used for the storage of program. The instructions can be preceded by a label. This consists of an integer in the range 1 to 9999 and the label is followed by a colon to separate it from the rest of the instruction. The numerical label can be used as the argument to a jump instruction, as detailed below.

In the description of the individual instructions below, instruction argument types are specified as follows:

R

specifies a register in the form R0, R1, R2, R3, R4, R5, R6 or R7 (or r0, r1, etc.).

L

specifies a numerical label (in the range 1 to 9999).

V

specifies a “variable location” (a variable number or a variable location pointed to by a register - see below).

A

specifies a constant value, a variable location, a register or a variable location pointed to by a register (an indirect address). Constant values are specified as an integer value, optionally preceded by a minus sign, preceded by a # symbol. An indirect address is specified by an @ followed by a register.

So, for example, an A-type argument could have the form 4 (variable number 4), #4 (the constant value 4), r4 (register 4) or @r4 (the contents of register 4 identifies the variable location to be accessed).

The instruction set is defined as follows:

LOAD A,R

loads the integer value specified by A into register R.

STORE R,V

stores the value in register R to variable V.

Objective: The main objective of the two programs is student able to learn formal mathematical way to describe the language. Translate each sub-tree of the decorated AST (Abstract syntax Trees) into intermediate code.

Prerequisites: Require knowledge of linked lists and trees

Applications: Able to use the formally define the grammar of a language, so that there is no disagreement or ambiguity BNF is so unambiguous that there is a lot of mathematical theory around these kinds of grammars

LEAD: 1. Configure a CFG which will accept the user input string following the given CFG.

S -> aBb/ccA

A -> b/c

B -> a/b

2. Check whether the given program contains comments.

5 (A) LAB SCHEDULE:

CYCLE 1:

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------------|---------------------|--------|--------|--------|--------|--------|--------|--------|
| B1(501 to 512) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |
| B2(513 to 524) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |
| B3(525 to 536) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |
| B4(537 to 548) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |
| B5(549 to 560) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week-16 |
|----------------|--------|---------|---------|---------|---------|---------|----------|---------|
| B1(501 to 512) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |
| B2(513 to 524) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |
| B3(525 to 536) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |
| B4(537 to 548) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |
| B5(549 to 560) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |

CYCLE 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------------|---------------------|--------|--------|--------|--------|--------|--------|--------|
| B6(561 to 572) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |
| B7(573 to 584) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |

| | | | | | | | | |
|-----------------|---------------------|-------|-------|-------|-------|-------|-------|------|
| B8(585 to 596) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |
| B9(597 to 5A8) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |
| B10(5A9 to 5C0) | Demo & Introduction | Exp.1 | Exp.1 | Exp.2 | Exp.2 | Exp.2 | Exp.3 | test |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week-16 |
|-----------------|--------|---------|---------|---------|---------|---------|----------|---------|
| B6(561 to 572) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |
| B7(573 to 584) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |
| B8(585 to 596) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |
| B9(597 to 5A8) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |
| B10(5A9 to 5C0) | Exp.4 | Exp.4 | Exp.5 | Exp.5 | Exp.6 | Exp.6 | LEAD1 &2 | Test |

(B) VIVA SCHEDULE: The viva schedule should be planned prior to the lab experiment.

CYCLE 1:

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|
| B1,B2,B3 | viva | | | | | viva | | | | | viva | | | |
| B4,B5,B1 | | viva | | | | | viva | | | | | viva | | |
| B2,B3,B4 | | | viva | | | | | viva | | | | | viva | |
| B5,B1,B2 | | | | viva | | | | | viva | | | | | viva |
| B2,B3,B4 | | | | | viva | | | | | viva | | | | |

CYCLE 2:**ROUND - 1**

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|-----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|
| B6,B7,B8 | viva | | | | | viva | | | | | viva | | | |
| B9,B10,B6 | | viva | | | | | viva | | | | | viva | | |
| B7,B8,B9 | | | viva | | | | | viva | | | | | viva | |
| B10,B6,B7 | | | | viva | | | | | viva | | | | | viva |
| B8,B9,B10 | | | | | viva | | | | | viva | | | | |

ROUND - 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|
| SG1 | viva | | | | | | | | | |
| SG2 | | | viva | | | | | | | |
| SG3 | | | | | viva | | | | | |
| SG4 | | | | | | | viva | | | |
| SG5 | | | | | | | | | viva | |

*SG: Selected Group with a maximum of 6 or 12 students

(C) SCHEME OF EVALUATION OF LABS**LAB INTERNAL:**

| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
|--------------------------|----------------------|-------------------------------|----------|----------|------------------|---------------------|----------|
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Mark s-3 | Mar ks-3 | Marks-4 | Marks-3 | Mark s-3 |
| Total Marks-25 | | | | | | | |

LAB EXTERNAL:

| S.NO | Write up | Final Evaluation | Viva |
|-----------------------|---|--|---|
| 1 | 1. Aim 2. Procedure etc 3. Program 4. Result etc | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

6. SUGGESTED BOOKS:

T1: Principles of compiler design – A.V. Aho, J.D. Ullman; Pearson Education.
T2: Modern Compiler Implementation in C – Andrew N. Appel, Cambridge University Press.

7. WEBSITES

1. www.compilerdesign/...../compilerdesign.
2. fita.hua.edu.vn/pttien/Setups/.../beginning-compilerdesign.pdf
3. www.cse.hcmut.edu.vn/~hungnq/courses/nap/alp.pdf
4. <https://sites.google.com/site/cdlabjntu09/home>
5. <http://youtube.com/compilerdesign>
6. <http://www.nptel.com/computerscience/compilerdesign>
7. <http://nptel.iitm.ac.in/courses.php?disciplineId=106>

8 .EXPERTS DETAILS

The expert details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally.

INTERNATIONAL

1. Chris Bates, WILEY Dreamtech
2. Sebesta, Pearson Education
3. Marty Hall & Larry Brown ,Pearson Education
4. Dietel & Nieto ,PHI/Pearson Education Asia.
5. Knuckles, John Wiley

NATIONAL

6. G.Venkata Rami Reddy ,Associate professor,School of Information Technology ,JNT University Hyderabad

7. Dr. M. Srinivasa Rao ,Professor,Dean(MSIT), JNT University Hyderabad

REGIONAL

1. Mohammed Ahsan Siddiqui, Assoc. Prof, IT Dept. BIET.Hyderabad.

2. Mr.S.Madhu Assoc.Prof , CSE Dept.GNEC, Hydearabad

3. Prof.P. Viswapathi, Pro& HOD, Dept. of CSE, CMREC,Hyderabad.

9. **Mapping Lab with Project/Consultancy/R&D Proposals**

Proposal 1: Project Design & Execution

Proposal 2: R& D Level Project Design & Execution

Proposal 3: Consultancy Task / Project Design & Development

Compiler Construction

Aastha Singh^{*} , Sonam Sinha^{} , Archana Priyadarshi^{**}**

Abstract- Compiler construction is a widely used software engineering exercise, but because most students will not be compiler writers, care must be taken to make it relevant in a core curriculum. The course is suitable for advanced undergraduate and beginning graduate students. Auxiliary tools, such as generators and interpreters, often hinder the learning: students have to fight tool idiosyncrasies, mysterious errors, and other poorly educative issues. We introduce a set of tools especially designed or improved for compiler construction educative projects in C.We also provide suggestions about new approaches to compiler Construction. We draw guidelines from our experience to make tools suitable for education purposes. The final result of this paper is to provide a general knowledge about compiler design and implementation and to serve as a springboard to more advanced courses. Although this paper concentrates on the implementation of a compiler, an outline for an advanced topics course that builds upon the compiler is also presented by us.

10 (Industrial Visit) (VIP= Visit of Industrial Proposals)

If students are visited different software development companies like Data analytixds, Redhat Inc, fedora inc.

| S.no | Type of industry | Nature of industry | Name of the company | Date of visit | No. of students participated | Year/branch | remarks |
|------|------------------|----------------------|-----------------------------|---------------|------------------------------|-------------|---------|
| 1 | MNC | Software Development | Data analytixds | | | | |
| 2 | MNC | Net working | Cisco Systems India Pvt Ltd | | | | |
| 3 | MNC | Network Security | Redhat Inc | | | | |
| 4 | MNC | Network Security | Fedora Inc | | | | |

11. GUIDELINES FOR SHADOW ENGINEERING (VIP) INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM):

OBJECTIVES OF SHADOW ENGINEERING:

1. The program which uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students

**TABLE 2: INDUSTRIAL TRAINING (Shadow Engg)
(Career Visit Approval)**

| S.No | Name of the Course | Nature of industry | Duration of Training | Authority | Date of Training/Certificate No. | Remarks |
|------|--------------------------|--------------------|----------------------|------------|----------------------------------|---------|
| 1 | Linux socket programming | Linux Admin | 2days | Fedora Inc | | |
| 2 | Multithread engineering | Shell script | 2days | Data Quest | | |
| 3 | | | | | | |

12. CALIBRATION & TESTING

Calibration: There is no need to be calibrated any equipment for this laboratory.

Installation: Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment

13. PREVENTIVE MAINTAINANCE SCHEDULE

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

| S.No. | Name of the Equipment | Date of Maintenance | Type of Activity | Remarks |
|--------------|------------------------------|----------------------------|-------------------------|----------------|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

(2) Maintenance Reports duly signed by in charges as well as HODs and duly approved by Principal periodically.

14. TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S.No., Equipment Name , Type of Problem (Abnormal Sound, Corrupt Software, Anti Virus Problem, Missing of Display, CRT not working,), Expected Reasons .Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Equipment Name | Type of Trouble | Remedial Activity | Remarks |
|--------------|-----------------------------------|-----------------------|------------------------|--------------------------|----------------|
| | | | | | |

CASE TOOLS LAB

CONTENTS:

- 1. OBJECTIVES AND RELEVANCE**
- 2. SCOPE**
- 3. PREREQUISITES**
- 4. SYLLABUS AS PER JNTUH**
- 5. LAB SCHEDULE**
- 6. SUGGESTED BOOKS**
- 7. WEBSITES (USEFUL LINKS)**
- 8. EXPERT DETAILS**
- 9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D**
- 10. PROPOSALS**
- 11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
- 12. CALIBRATION, TESTING AND INSPECTION**
- 13. PREVENTIVE MAINTENANCE SCHEDULES**
- 14. TROUBLESHOOTING**

1. OBJECTIVES AND RELEVANCE

The main objective of the lab course Documenting user requirements using the UML notation and designing the software architecture before coding for provide good quality software.

2. SCOPE

The scope of this lab is to understand the how to designing the software architectures using the UML notations.

3. PREREQUISITES

This lab requires the basic knowledge about the basic building blocks of uml and uses of things and relationships in development of diagrams. Common uses of diagrams, cutting operations and basics of lubrication, different types of chips and different cutting operations.

4. SYLLABUS-JNTU

Main linkage of machine tools theory with lab experiments:

i. Introduction to Case Tools.

ii).Automatic Teller Machine Uml Diagrams

Description of ATM System

The software to be designed will control a simulated automated teller machine (ATM) having a magnetic stripe reader for reading an ATM card, a customer console (keyboard and display) for interaction with the customer, a slot for depositing envelopes, a dispenser for cash, a printer for printing customer receipts, and a key-operated switch to allow an operator to start or stop the machine. The ATM will communicate with the bank's computer over an appropriate communication link. (The software on the latter is not part of the requirements for this problem.)

The ATM will provide service to one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN) – both of which will be sent to the bank for validation as part of each transaction. The customer will then be able to perform one or more transactions. The card will be retained in the machine until the customer indicates that he/she desires no further transactions, at which point it will be returned – except as noted below.

The ATM must be able to provide the following services to the customer:

- A. A customer must be able to make a cash withdrawal from any suitable account linked to the card. Approval must be obtained from the bank before cash is dispensed.
- B. A customer must be able to make a deposit to any account linked to the card, consisting of cash and/or checks in an envelope. The customer will enter the amount of the deposit into the ATM, subject to manual verification when the envelope is removed from the

LIFT MANUAL

machine by an operator. Approval must be obtained from the bank before physically accepting the envelope.

- C. A customer must be able to make a transfer of money between any two accounts linked to the card.
- D. A customer must be able to make a balance inquiry of any account linked to the card.
- E. A customer must be able to abort a transaction in progress by pressing the Cancel key instead of responding to a request from the machine.

The ATM will communicate each transaction to the bank and obtain verification that it was allowed by the bank. Ordinarily, a transaction will be considered complete by the bank once it has been approved. In the case of a deposit, a second message will be sent to the bank indicating that the customer has deposited the envelope. (If the customer fails to deposit the envelope within the timeout period, or presses cancel instead, no second message will be sent to the bank and the deposit will not be credited to the customer.)

If the bank determines that the customer's PIN is invalid, the customer will be required to re-enter the PIN before a transaction can proceed. If the customer is unable to successfully enter the PIN after three tries, the card will be permanently retained by the machine, and the customer will have to contact the bank to get it back.

If a transaction fails for any reason other than an invalid PIN, the ATM will display an explanation of the problem, and will then ask the customer whether he/she wants to do another transaction.

The ATM will provide the customer with a printed receipt for each successful transaction, showing the date, time, machine location, type of transaction, account(s), amount, and ending and available balance(s) of the affected account ("to" account for transfers).

The ATM will have a key-operated switch that will allow an operator to start and stop the servicing of customers. After turning the switch to the "on" position, the operator will be required to verify and enter the total cash on hand. The machine can only be turned off when it is not providing any service to a customer. When the switch is moved to the "off" position, the machine will shut down, so that the operator may remove deposit envelopes and reload the machine with cash, blank receipts, etc.

EXPERIMENT: 1

Name of the experiment: Class diagram for ATM System

AIM: To design and implement class diagram for ATM system

Applications:

The purpose of the class diagram is to model the static view of an application

EXPERIMENT: 2

NAME OF EXPERIMENT: Use case diagram for ATM System.

AIM: To design and implement Use case diagram for ATM System.

Applications:

The purpose of use case diagram is to capture the dynamic aspect of a system
Used to gather requirements of a system. Used to get an outside view of a system

To identify external and internal factors influencing the system.

Shows the interacting among the requirements are actors

EXPERIMENT: 3

Name of the experiment: Sequence diagram for ATM System

AIM: To design and implement Sequence diagram for ATM System

Applications:

A Sequence diagram is an interaction diagram that emphasizes the time ordering of messages.

This diagram is used to show the dynamic view of a system.

Object organization

EXPERIMENT: 4

Name of the experiment: collaboration diagram for ATM System

AIM: To design and implement collaboration diagram for ATM System

Applications:

3.1 To capture dynamic behaviour of a system.

3.2 To describe the message flow in the system

EXPERIMENT: 5

NAME OF EXPERIMENT: State chart diagram for ATM System.

AIM: To design and implement State chart diagram for ATM System

Applications:

1. State chart diagram is used to model dynamic nature of a system
2. They define different states of an object during its lifetime

EXPERIMENT: 6

NAME OF EXPERIMENT: Activity diagram for ATM System.

AIM: To design and implement Activity diagram for ATM System.

Applications:

Activity diagram is basically a flow chart to represent the flow from one activity to another.
The activity can be described as an operation of the system.

EXPERIMENT: 7

NAME OF EXPERIMENT: Component diagram for ATM System.

AIM: To design and implement Component diagram for ATM System

Objectives:

Component diagrams can be described as a static implementation view of a system.

Static implementation represents the organization of the components at a particular moment

EXPERIMENT: 8

NAME OF EXPERIMENT: Deployment diagram for ATM System.

AIM: To design and implement Deployment diagram for ATM System.

Applications:

Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed

LEAD: 1. CASE STUDY OF LIBRARY MANAGEMENT SYSTEM

5(A). LAB SCHEDULE: The lab schedule should be planned once in a week.

The week wise scheduled experiment should be completed.

CYCLE 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 |
|----------------|--------|--------|--------|--------|--------|--------|
| B1(501 to 512) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |
| B2(513 to 524) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |
| B3(525 to 536) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |
| B4(537 to 548) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |
| B5(549 to 560) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |

| Batches | week-7 | week-8 | Week-9 | week-10 | week-11 | week-12 |
|----------------|--------|--------|--------|---------|---------|----------|
| B1(501 to 512) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |
| B2(513 to 524) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |
| B3(525 to 536) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |
| B4(537 to 548) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |
| B5(549 to 560) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |

CYCLE 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 |
|-----------------|--------|--------|--------|--------|--------|--------|
| B6(561 to 572) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |
| B7(573 to 584) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |
| B8(585 to 596) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |
| B9(597 to 5A8) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |
| B10(5A9 to 5C0) | Demo | Exp1 | Exp2 | Exp3 | Exp4 | test |

| Batches | week-7 | week-8 | Week-9 | week-10 | week-11 | week-12 |
|-----------------|--------|--------|--------|---------|---------|----------|
| B6(561 to 572) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |
| B7(573 to 584) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |
| B8(585 to 596) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |
| B9(597 to 5A8) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |
| B10(5A9 to 5C0) | Exp5 | Exp6 | Exp7 | Exp8 | test | Lead exp |

(B) VIVA SCHEDULE: The viva schedule should be planned prior to the lab experiment.

CYCLE 1:**ROUND - 1**

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|
| B1,B2,B3 | viva | | | | | viva | | | | | viva | |
| B4,B5,B1 | | viva | | | | | viva | | | | | viva |
| B2,B3,B4 | | | viva | | | | | viva | | | | |
| B5,B1,B2 | | | | viva | | | | | viva | | | |
| B2,B3,B4 | | | | | viva | | | | | viva | | |

CYCLE 2:**ROUND - 1**

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 |
|-----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|
| B6,B7,B8 | viva | | | | | viva | | | | | viva | |
| B9,B10,B6 | | viva | | | | | viva | | | | | viva |
| B7,B8,B9 | | | viva | | | | | viva | | | | |
| B10,B6,B7 | | | | viva | | | | | viva | | | |
| B8,B9,B10 | | | | | viva | | | | | Viva | | |

ROUND - 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|
| SG1 | viva | | | | | | | | | |
| SG2 | | | viva | | | | | | | |
| SG3 | | | | | viva | | | | | |
| SG4 | | | | | | | viva | | | |
| SG5 | | | | | | | | | viva | |

(C) SCHEME OF EVALUATION OF LABS:

The scheme of evaluation for the lab internal and lab external as follows

| LAB INTERNAL: | | | | | | | |
|---------------------------------|---------------------------------|--------------------------------------|----------------|----------------|-------------------------|--------------------------------|----------------|
| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-4 | Marks-3 | Marks-3 |
| Total Marks-25 | | | | | | | |

| LAB EXTERNAL: | | | |
|-----------------------|--|---|--|
| S.NO | Write up | Final Evaluation | Viva |
| 1 | 1. Aim 2. Procedure 3. Program 4. Expected output | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

5.

6.SUGGESTED BOOKS:

- Object-Oriented Analysis and Design with Applications (3rd Edition) [Kindle Edition]
Grady Booch (Author), Robert A. Maksimchuk (Author)

7.WEB SITES(USEFUL LINKS):

- <http://jntuhome.com/wp-content/uploads/2011/07/R09-CSE-Lab-3-4.pdf>
- <http://forum.jntuworld.com/showthread.php?22468-Case-Tools-and-Software-Testing-Lab-Manual>
- <http://www.scribd.com/doc/188516982/CT-and-ST-LAB-Manual>
- <http://forum.jntuworld.com/showthread.php?20471-Lab-Manuals-For-JNTUH-JNTUK-amp-JNTUA-%28For-All-Courses-All-Branches-amp-All-Regulations%29>
- <http://www.mamse.co.in/study/cse/7sem/CS1311.pdf>
- <http://sourcecodesonline.blogspot.in/2010/12/atm-system-using-case-tools.html>

8. EXPERTS' DETAILS

The expert details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally.

INTERNATIONAL

- Grady Booch ,IBM Fellow
gbooch@us.ibm.com +1-720-299-8292
- Ivar Jacobson

REGIONAL

- Dr. Govardhan Reddy Head of the Department, kottam tulasi reddy college,kurnool

2. Mr. D.Kondaiah, Assoc Professor , Srinidhi Institute of Science and Technology ,Hyderabad.

9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R&D:

The case tools lab course should be designed in such a way that it should meet the requirements of research and development as well as consultancy projects. Also the Proposals of Project/R&D/Consultancy are as follows:

❖ Measuring the Quality of Software through Analytical Design by OOAD Metrics

ABSTRACT

Software plays an important role in today's computerized world. The programmer can use different languages to develop the software. In order to develop software, it needs several phases such as Analysis, Design, Implementation, Testing and Maintenance. Of these several phases, the analysis and design becomes essential, since these are the most essential feature in the development of the software.

❖ An article/white paper from a magazine /journal/weekly/any periodical in the entitled Subject

In this article, we are going to learn about how to identify and design conceptual classes and different architectural classes from the application domain. This knowledge would help you in designing architectural classes during early construction stages of your application.

Introduction

Understanding the various entities from the problem domain, their relationships, and their interactions is a useful starting point of analysis, and is commonly known as domain modeling. Domain modeling helps in subsequent design of the systems.

A domain model consists of conceptual classes from the application domain. The conceptual classes may represent people, organizations, things, events, or even ideas - virtually anything that exists in the application domain which is of some interest to the system. In this article, we are going to learn about how to identify and design conceptual classes and different architectural classes from the application domain. The use case model generated during Requirements Determination stage becomes the basis of the analysis and subsequently design and implementation activities. A common starting point of analysis is to identify analysis classes that focus on handling functional requirements of the system.

The analysis classes identified in this stage represent an early conceptual model of the system. These classes are likely to be significantly refined as we proceed with further analysis and design, and evolve into a final mode of the system. The analysis classes would morph or evolve into design classes with more concrete structures. The design classes would then be implemented through source code, resulting in an executable release of the system.

Since the analysis classes are likely to change and evolve during subsequent stages, you should find the right balance of formal documentation of the same. In the earlier stages, it may be adequate to just identify these classes, name them, and make a brief description in a few sentences.

Small Classes

A single class doing too many things makes it harder to understand, modify and extend. If a class is doing too many things, it is better to split it into multiple classes. In general, it is better to have a large number of small classes than a small number of large classes. A large class knows too much of the system's behaviour is difficult to modify and may carry extra baggage in case of reuse. In general, it is better to have a large number of small classes than a small number of large classes. A large class knows too much of the system's behaviour is difficult to modify and may carry extra baggage in case of reuse. On the other hand, a small class encapsulates small level of system's behaviour, is easy to understand and modify, and easy to reuse without extra baggage.

Steps in Class Design

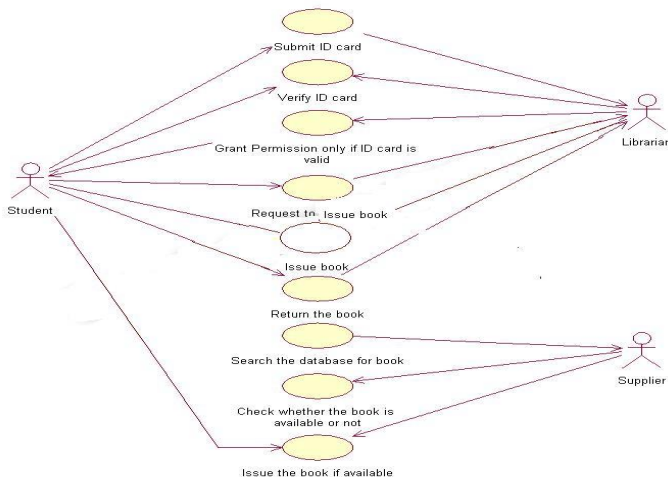
You should not jump to designing the full structure of a class straightaway. The correct approach should be as follows:

❖ Proposal for project activity:

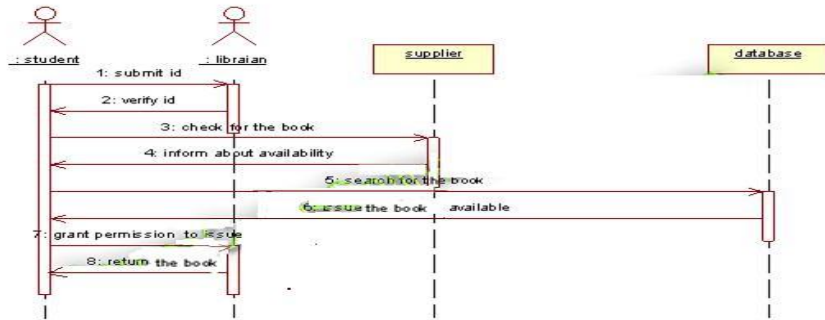
AIM: To model the "**Library Management System**" using the software Rational Rose with various UML (Unified Modeling Language) diagrams.

UML DIAGRAMS:

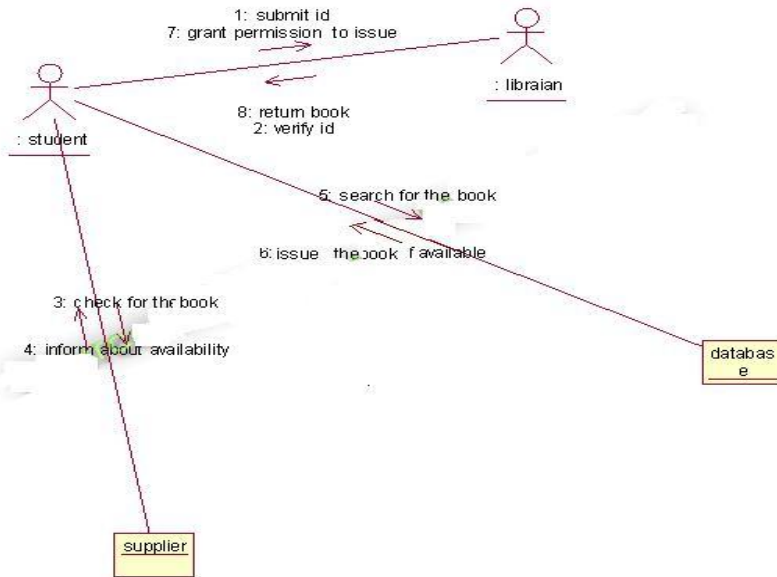
Use Case Diagram:



SEQUENCE DIAGRAM:

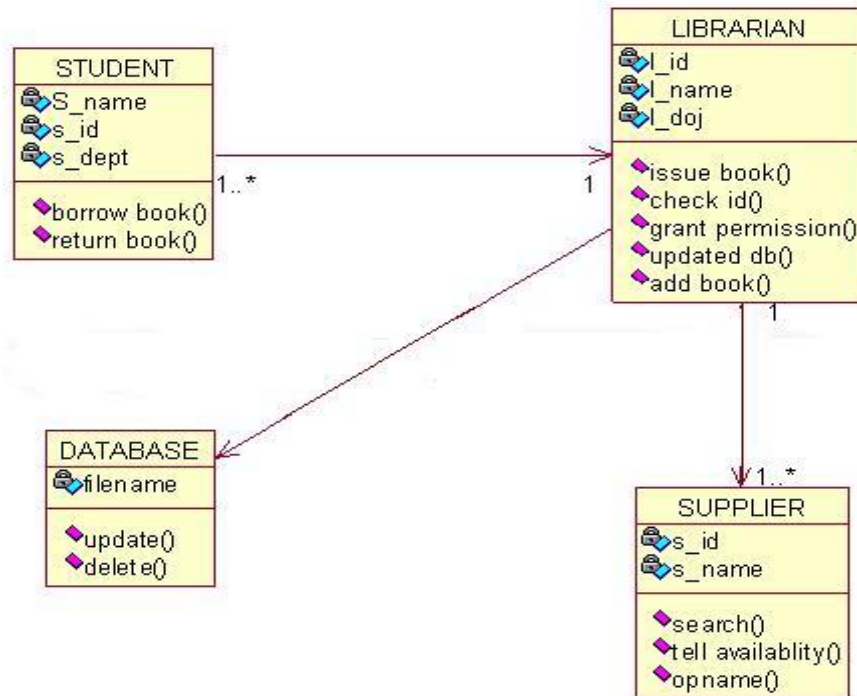


COLLABORATION DIAGRAM:

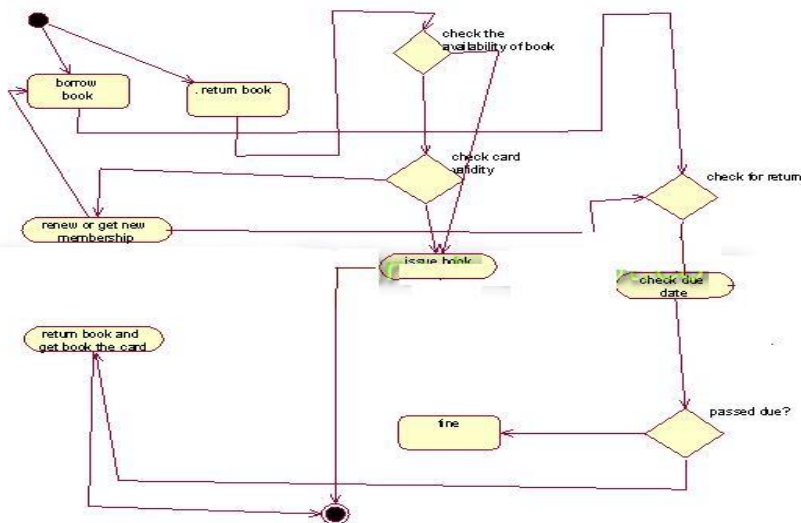


LIFT MANUAL

CLASS DIAGRAM

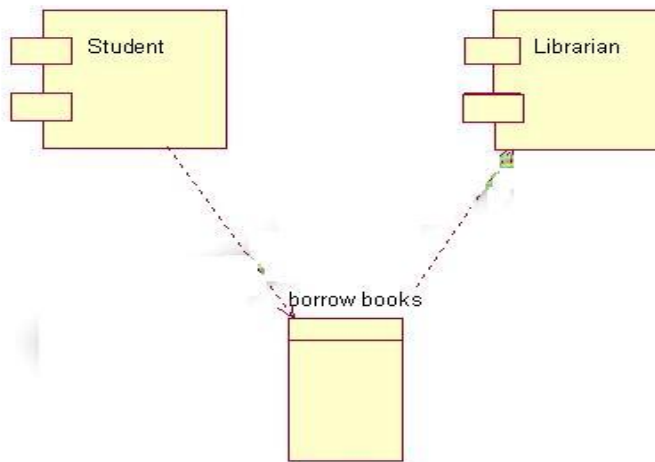


ACTIVITY DIAGRAM



LIFT MANUAL

COMPONENT DIAGRAM

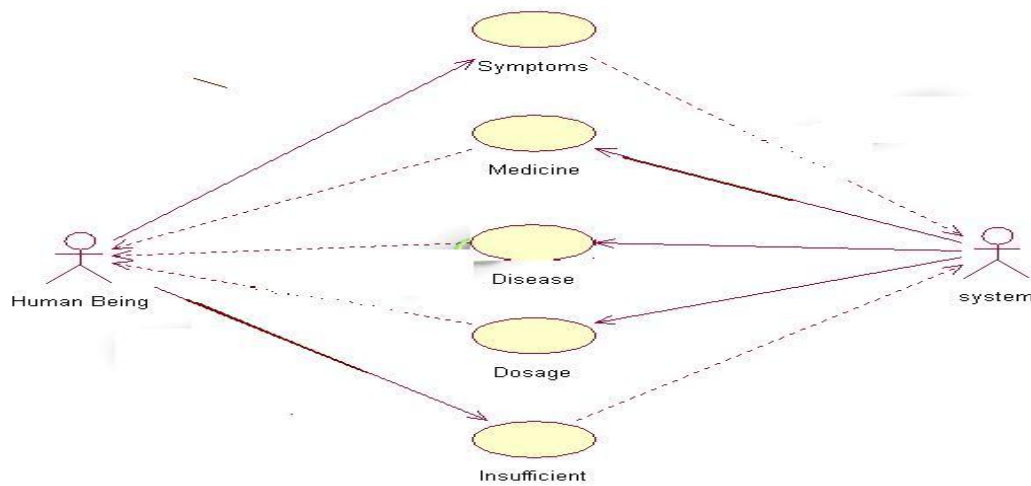


PROPOSAL FOR CONSULTANCY:

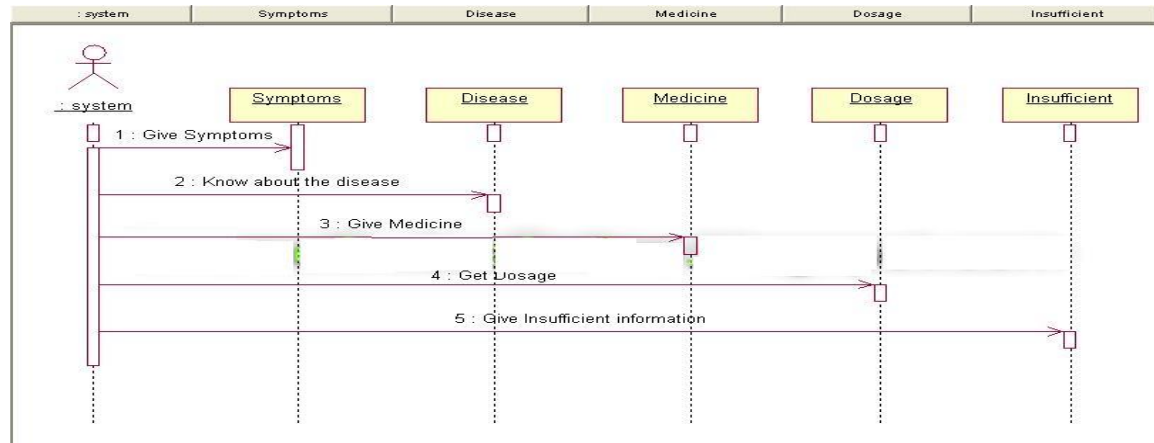
AIM:

To model the "**Medical Expert System**" using the software Rational Rose with various UML (Unified Modeling Language) diagrams.

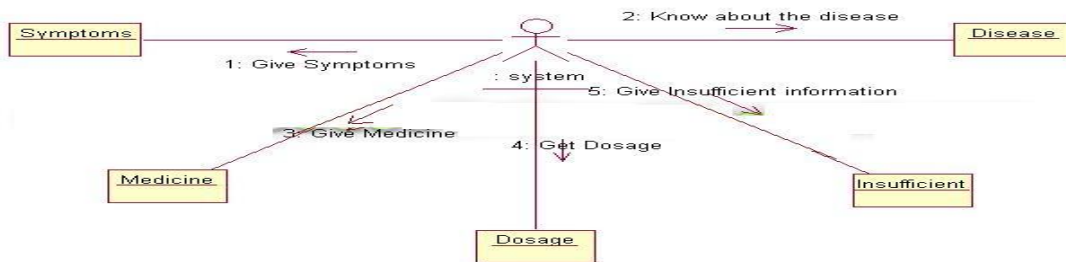
Use Case Diagram For Medical Expert



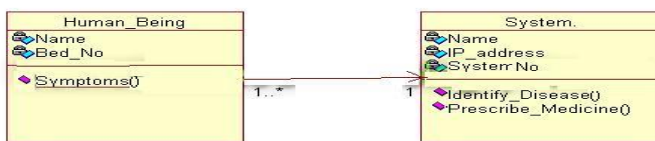
Sequence Diagram For Medical Expert System



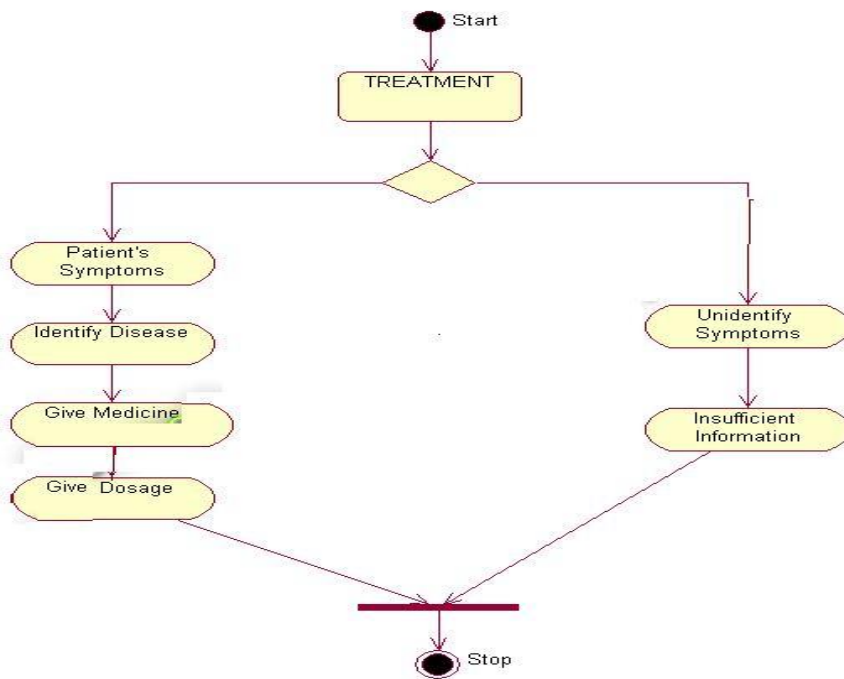
Collaboration Diagram For Medical Expert System



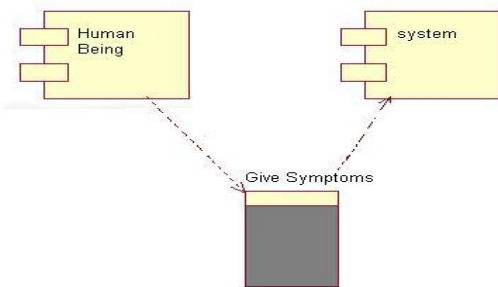
Class Diagram For Medical Expert System:



Activity Diagram For Medical Expert System:



Component Diagram For Medical Expert System



FUNDED/UNFUNDED PROPOSALS (if any):

1. There are no any funded programs or unfunded programs concerned for this lab

10. PROPOSALS

INDUSTRIAL VISITS

Industry proposals (week wise industrial visits)(in house or outside visit)/training programmes:

TABLE 1: INDUSTRIAL VISITS

| S.No | Type of industry | Nature of industry | Date of visit | No. of students participated | Year/branch | Remarks |
|------|------------------|--------------------|---------------|------------------------------|-------------|---------|
| | | | | | | |
| | | | | | | |

11. GUIDELINES FOR SHADOW ENGINEERING (VIP)

INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM):

SHADOW ENGINEERING:

Objectives of shadow engineering:

1. The program which uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

**TABLE 2: INDUSTRIAL TRAINING (Shadow Engg)
(Career Visit Approval)**

| S.No | Name of the Course | Nature of industry | Duration of Training | Authority | Date of Training/Certificate No. | remarks |
|------|--------------------|--------------------|----------------------|-----------|----------------------------------|---------|
| | | | | | | |
| | | | | | | |

12. CALIBRATION/INSTALLATION AND TESTING:

Calibration: There is no any equipment to be calibrated for concerning to this lab.

Installation: Aim of this concept is to make and maintain installation procedure for a Rational Rose software in existing equipment

1. Enter the following URL into your web browser:

<http://www.ibm.com/developerWorks/toolbox/download.html>

You should now be at the page below. Click on the “Enterprise Level” hyperlink

2. Click on “Rational” then “IBM Rational Rose Enterprise V2003.06.12 for Windows Evaluation”.
3. The page that you are directed to, contains further instructions for installing Rational Rose. You should print this page. There are two download methods discussed at the bottom of the page. The installation file is approximately 360 MB in size. These instructions will demonstrate using the Download Director link, which offers additional features, including the capability of continuing after a broken connection.

After selecting a download method, you will be directed to the software license agreement page below (background). After reading the agreement, select “I agree” and a username/password prompt will appear (foreground). Enter “csfullerton” in the username field and “titan_fullerton” in the password field then click on “ok”.

4. If you have either a Java Runtime or Java SDK installed, the security warning message shown below will appear. If the security warning message did not appear, proceed to instruction 4a. Click on yes if you would to continue.

Then select a download directory and click on OK to continue

After which, the Download Director progress applet shown below will appear

You can click on launch to begin installing Rational Rose Enterprise edition now, or you can install it later using the “C57CJML.exe” file that is located in the download directory you specified previously (click on details if you have forgotten the location).

5. . This screen explains how to use the Download Director software. If you do not have a Java runtime environment (a Java virtual machine), we encourage you to install the Java SDK available from <http://java.sun.com> (directions for installing the SDK are available there). You will need to install the Java SDK at some point since many MSE projects will be implemented in Java. If you’re not sure whether you have either a Java SDK or JRE installed on your machine, you can test this at <http://www.java.com> by clicking on the “Get It Now” link. If either the SDK or JRE are installed, a screen will notify you that you have “Java Software Installed”. Next Click on continue downloading

Select an install directory

Click next.

Select Rational Rose Enterprise Edition from the Rational Products folder below (see the blue highlighted file below) and click NEXT.

Click on UPDATE to continue.

- As the installation instruction says, you also need to download and save the license key file - "C53A1NA.upd" to run the Rational Software

For source mse.ecs.fullerton.edu/docs/how2install-Eval-Rational%20Rose.doc

http://share.auditory.ru/2010/Timofey.Kovalev/laba/rational_rose/install.pdf

Testing : Aim of this concept is to test the equipment after installation whether it meets the existing standards.

13. PREVENTIVE MAINTENANCE SCHEDULE

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

| S.No. | No. of. Systems installed with the software | Date of Maintenance | Type of Activity | Remarks |
|-------|---|---------------------|-------------------|---------|
| 1 | 60 | 30/06/2014 | For casetools lab | |

(2) Maintenance Reports duly signed by in charges as well as HODs and duly approved by the Principal periodically.

TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S.No., Equipment Name , Type of Problem (Too much Noise, Abnormal Sound, Corrupt Software, Anti Virus Problem, Missing of Display, CRT not working, Motor is not giving signal, Digital display is not working, Break of tools, Misalignment of machine elements, PLC is not properly working), Expected Reasons (Bearing failure, Improper alignment of machine centres, Missing of vibration pads etc)

Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Equipment Name | Type of Trouble | Remedial Activity | Remarks |
|-------|----------------------------|----------------|-----------------|-------------------|---------|
| | | | | | |

SOFTWARE TESTING LAB

CONTENTS:

1. OBJECTIVES AND RELEVANCE
2. SCOPE
3. PREREQUISITES
4. SYLLABUS AS PER JNTUH
5. LAB SCHEDULE
6. SUGGESTED BOOKS
7. WEBSITES (USEFUL LINKS)
8. EXPERT DETAILS
9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D
10. PROPOSALS
11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)
12. CALIBRATION, TESTING AND INSPECTION
13. PREVENTIVE MAINTENANCE SCHEDULES
14. TROUBLESHOOTING

1. OBJECTIVES AND RELEVANCE :

The main objective of the lab course is to gain practical knowledge by learning various testing tools such as win runner, selenium, Bugzilla, Test Director and Test link to perform automate testing.

2. SCOPE:

The scope of this lab is to make students to implement appropriate testing tools and perform testing on applications.

3. PREREQUISITES:

Students should have foundation in fundamentals programming .The student should have programming experience (in C/ C++/ JAVA). The student should have the knowledge on the scripting languages such as HTML.

4. JNTUH SYLLABUS:

PREAMBLE: This lab covers the experiments in software testing lab. The JNTU has given 11 experiments in the syllabus out of which first two experiments are based on the ‘C’ language, following three experiments deal with generation of reports on various bugs, test cases and test plan. The next five experiments deal with testing tools and the final experiment is to do mini project and create various testing documents and the final test report document.

EXPERIMENT NO: 1

AIM: Write program in ‘C’ language to demonstrate the working of the following construct i) do..while ii) while..do iii) if...else iv) switch v) for

OBJECTIVE:

The main objective of this experiment is to understand clearly the working procedure of various conditional and iterative statements.

PRE REQUISITES

The basic syntax and flow chart of this conditional and iterative statement is needed.

DESCRIPTION:

1. Introduction to Experiment-20 min
2. Selecting the suitable example for conditional and iterative statement.
3. Writing the code in ‘C’ language and execute.
4. Test the application for various inputs.

APPLICATIONS:

1. Applicable to write a code that makes decisions and repeats actions.
2. Applicable to execute a set of statements only if some condition is met.
3. Applicable to execute a set of statements zero or more times, until some condition is met.

EXPERIMENT 2:

AIM: A program written in ‘C’ language for matrix multiplication fails.

OBJECTIVE:The main objective of this experiment is to know the basic condition to get two matrix get multiplied, introspect the causes for its failure and know the possible reasons for its failure

PRE REQUISITES:

The student should have the knowledge on multi dimensional arrays concepts.

The student should also know the initialization and accessing of elements from the multi-dimensional array.

DESCRIPTION:

1. Introduction to Experiment-20 min
2. Accepting the numbers rows and columns for each matrix.
3. Test the condition for matrix multiplication.
4. Initialize the values for each matrix.
5. Execute the code for matrix multiplication.
- 6 Test the application for various inputs.

APPLICATION:

Introspects the causes for the failure matrix multiplication and list the reasons for the failure.

EXPERIMENT 3:

AIM:

Take any system (e.g. ATM system) and study its system specifications and report the various bugs.

OBJECTIVE:

The main objective is to know how to deal with the system specifications and how to generate a report on various bugs

PRE REQUISITES:

1. Need the system requirement and specification of ATM System.
2. Need to know how to generate report.

DESCRIPTION:

1. Describing the working nature of automated teller machine
2. Describing the various services provided by ATM to customer.
3. Understand the Specifications of the system.
4. Identifying the various scenarios causing bugs during transactions.
5. Finally generate the report by listing the various bugs.

APPLICATION:

This knowledge will help the student to generate the report on various bugs in any system by studying its specification.

EXPERIMENT 4:

AIM:

Write the test cases for any known application (E.g.: Banking application)

OBJECTIVE:

The major objective is to make the student understand how to write the test case for any System.

PRE REQUISITES:

1. Should know the characteristics of a Banking application.
2. Have the knowledge how the system is going to serve the customer.

DESCRIPTION:

1. In-depth study of the Banking application on which test cases are to be written.
2. Identify the **Fields in test cases such as**

Test case id:

Unit to test: What to be verified?

Assumptions:

Test data: Variables and their values

Steps to be executed: Expected result:

Actual result: Pass/Fail: Comments

3. Write the test cases on each operation of the Banking system such as opening bank account, account transaction through net etc.

APPLICATION:

A test case is used by quality engineer to determine whether an application is working correctly or not.

EXPERIMENT 5:

AIM:

Create a test plan document for any application (e.g. library management system)

OBJECTIVE:

The main objective is to understand how to write the test plan for any given system.

PRE REQUISITES:

1. System architecture outlining the whole system from user interface, logic or business layer and database.
2. Project schedule i.e. what will be developed per iteration.
3. High level of requirements, use cases and features to be implemented etc.

DESCRIPTION:

1. In-depth study of the Banking application on which test plan are to be written
2. Test plan should include the following:
 - i) Writing an introduction. An introduction includes a general description and schedule of a test, as well as any related documents.
 - ii) Writing a section on required resources. This section describes all of the resources needed to complete the testing, including hardware, software, testing tools, and staff.
 - iii) Writing a section on what you are going to test.
 - iv) Writing a section on what you will not be testing.
 - v) Writing a section on risks and dependencies.
 - vi) Writing a section on the results of your project.

APPLICATION:

A test plan is used to verify that a product is tested in accordance with its design specifications and that the product meets all requirements.

EXPERIMENT 6:

AIM:

Study of any testing tool (e.g. win runner)

OBJECTIVE:

The main objective is to make the student know about the testing tool.

PRE REQUISITES:

1. Should have a fair idea about the requirement specifications before you begin with testing.

2. You need to be aware of the software and hardware requirements of the application you are working with.

DESCRIPTION:

Testing with win runner includes 4 main stages:

1. Selecting manual functional test cases to be automated
2. Creating automated test scripts in TSL (test script language)
3. Running automated test scripts on builds
4. Analyzing automated test script result.

APPLICATION:

1. Win runner is used as functional testing tool
2. It supports various technologies like JAVA, .NET for automated functional testing
3. Used to convert manual functional tests into Test Script Language

EXPERIMENT 7:

AIM:

Study on any web testing tool (e.g.: selenium)

OBJECTIVE:

To enable the student to test any web based application.

PRE REQUISITES:

1. Any browser such as Firefox
2. Java development kit.
3. J unit for java remote client.(ONLY if you are using java client, you also need to add selenium java client driver in class path)
4. Student should have knowledge on Script language such as HTML

DESCRIPTION:

Describe about the selenium tool and its various versions.

Description on installation of selenium tool kit

Explain about the procedure for testing any web application which includes the following steps:

1. Record your test case using selenium IDE.
2. Play your test case using selenium IDE itself to make sure that your test case works.
3. Export your recording test case as html file.
4. Create a test suite.
5. Run your exported html case in Firefox using selenium.
6. (Option) Run your exported test case in programming like java.

APPLICATION:

1. Selenium testing tool is used for testing any web based application.
2. Developers use it for “browser” regression testing. (**Regression testing** is a type of software that seeks to uncover new software bugs in the system after enhancements.)

EXPERIMENT 8:

AIM:

Study any bug tracking tool (e. g : Bugzilla, bugbit)

OBJECTIVE:

To enable the student to use "Defect Tracking System" or "Bug- Tracking System"

PRE REQUISITES:

A compatible database management system.

A compatible web server

DESCRIPTION:

Describe about the Bugzilla tool and its various versions.
Description of step by step installation of Bugzilla
Explain about various components of Bugzilla such as database.
Describing various operations to use bugzilla includes

- i) Create a bugzilla account
- ii) Bugzilla query page
- iii) Creating and managing bug reports

APPLICATIONS:

- i. It used for bug tracking and code changes
- ii. It is also used to communicate with teammates
- iii. It is used for managing quality assurance (QA)

EXPERIMENT 9:

AIM:

Study of any test management tool (e.g.: Test Director)

OBJECTIVE:

To make student to use Test Director for creating database of manual and automated tests, build test cycle, run test and report and track defects.

PRE REQUISITES:

Student should have knowledge on test plans, test scripts and test cases

DESCRIPTION:

Describe about the Test Director and its various versions.
Description of step by step installation of Test Director
Explaining the workflow of Test Director that consists of the following 3 main phases:

- 1. Planning tests
- 2. Running tests
- 3. Tracking defects

Planning Tests phase includes

- 1. Defining your testing goals.
- 2. Defining test subjects.
- 3. Defining tests.
- 4. Designing test steps.
- 5. Automating tests.
- 6. Analyzing the test plan.

Running test phase includes

- 1. Creating test sets
- 2. Running test sets
- 3. Analyzing the testing progress

Tracking Defects phase includes

- 1. Reporting the defects detected in your application
- 2. Tracking defects
- 3. Analyzing defect tracking

APPLICATION:

- 1. Test Director is used to simplify and organizes test management by giving systematic control over the testing process.
- 2. Test Director provides anytime, anywhere access to testing assets.

EXPERIMENT 10.

AIM: Study of any open source –testing tool (e.g.: Test Link)

OBJECTIVE:

To make the student use any open source testing tool for storing the information of how testing is to be done, plan testing activities and report the status of quality assurance activities.

PRE REQUISITES:

Student should have knowledge of access to a web server and a database in order to install and run it.

DESCRIPTION:

Explanation of Test Link tool and its versions.

Explaining steps in installation of Test Link tool

Explaining the work flow of test link for testing the application.

APPLICATION:

1. Used for collecting and organizing your cases dynamically.
2. Used for managing the testing process.
3. Used for tracking bug status, defects and projects tasks, and allows collaboration across the team.

Lead: A study on QTP

5(A) LAB SCHEDULE: The lab schedule should be planned once in a week. The week wise scheduled experiment should be completed

CYCLE 1:

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------------|--------|----------|----------|----------|----------|----------|--------|--------|
| B1(501 to 512) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |
| B2(513 to 524) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |
| B3(525 to 536) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |
| B4(537 to 548) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |
| B5(549 to 560) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week-16 |
|----------------|----------|----------|----------|----------|-----------|-----------|----------|---------|
| B1(501 to 512) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | Test |
| B2(513 to 524) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | Test |
| B3(525 to 536) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | Test |
| B4(537 to 548) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | Test |
| B5(549 to 560) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | Test |

CYCLE 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------------|--------|----------|----------|----------|----------|----------|--------|--------|
| B6(561 to 572) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |
| B7(573 to 584) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |
| B8(585 to 596) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |
| B9(597 to 5A8) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |
| B10(5A9 to 5C0) | Demo | Exp no:1 | Exp no:2 | Exp no:3 | Exp no:3 | Exp no:4 | test | Exp.5 |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week-16 |
|-----------------|----------|----------|----------|----------|-----------|-----------|----------|---------|
| B6(561 to 572) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | test |
| B7(573 to 584) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | test |
| B8(585 to 596) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | test |
| B9(597 to 5A8) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | test |
| B10(5A9 to 5C0) | Exp no:6 | Exp no:7 | Exp no:8 | Exp no:9 | Exp no:10 | Exp no:11 | LEAD1 &2 | test |

(B). VIVA SCHEDULE:

The viva schedule should be planned prior to the lab experiment.

CYCLE 1:**ROUND - 1**

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|
| B1,B2,B3 | viva | | | | | viva | | |
| B4,B5,B1 | | Viva | | | | | viva | |
| B2,B3,B4 | | | viva | | | | | viva |
| B5,B1,B2 | | | | viva | | | | |
| B2,B3,B4 | | | | | viva | | | |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|----------|--------|---------|---------|---------|---------|---------|
| B1,B2,B3 | | | viva | | | |
| B4,B5,B1 | | | | viva | | |
| B2,B3,B4 | | | | | viva | |
| B5,B1,B2 | viva | | | | | viva |
| B2,B3,B4 | | viva | | | | |

CYCLE 2:**ROUND - 1**

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------|--------|--------|--------|--------|--------|--------|--------|--------|
| B6,B7,B8 | viva | | | | | viva | | |
| B9,B10,B6 | | viva | | | | | viva | |
| B7,B8,B9 | | | viva | | | | | viva |
| B10,B6,B7 | | | | viva | | | | |
| B8,B9,B10 | | | | | viva | | | |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|-----------|--------|---------|---------|---------|---------|---------|
| B6,B7,B8 | | | viva | | | |
| B9,B10,B6 | | | | viva | | |
| B7,B8,B9 | | | | | viva | |
| B10,B6,B7 | viva | | | | | viva |
| B8,B9,B10 | | viva | | | | |

ROUND - 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|
| SG1 | viva | | | | | | | | | |
| SG2 | | | viva | | | | | | | |
| SG3 | | | | viva | | | | | | |
| SG4 | | | | | | | viva | | | |
| SG5 | | | | | | | | | viva | |

*SG: Selected Group with a maximum of 6 or 12 students

(C) SCHEME OF EVALUATION OF LABS:

The scheme of evaluation for the lab internal and lab external as follows

| <u>LAB INTERNAL:</u> | | | | | | | |
|---------------------------------|---------------------------------|--------------------------------------|----------------|----------------|-------------------------|--------------------------------|----------------|
| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-4 | Marks-3 | Marks-3 |
| Total Marks-25 | | | | | | | |

| <u>LAB EXTERNAL:</u> | | | |
|-----------------------------|--|--|---|
| S.NO | Write up | Final Evaluation | Viva |
| 1 | 1. Aim 2. Procedure 3. Program 4. Expected output | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

6. SUGGESTED BOOKS:

1. Software Testing Concepts and Tools, P.Nageswara Rao, dreamtech press.
2. Effective Software Testing Methodologies by William E.Perry
3. Software Testing Tools, Dr.K.V.K.K.Prasad, dreamtech Press.
4. Effective methods of software testing ,perry,John Wiley 2nd Edition
5. Foundations of Software Testing ,A.P. Mathur,pearson

7. WEBSITES (USEFUL LINKS):

- <http://www.slideshare.net/himanshuhora/load-runner-win-runner>.
- <http://www.etestinghub.com/winrunner.php>
- <http://www.pushtotest.com/selenium-tutorial-for-beginners-blog.html>.

- http://erik.doernenburg.com/wp-content/archive/Automated_Web_Testing.pdf
- <http://www.bugzilla.org/docs/4.4/en/pdf/Bugzilla-Guide.pdf>.
- <http://www.ebookzdownload.com/uploads/TestDirector-Tutorial.pdf>.
- <https://wiki.jenkins-i.org/download/attachments/753702/jenkins.pdf>

8. EXPERT DETAILS:

The expert details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally.

INTERNATIONAL:

1.Dr.Basili, Victor, ph.D

professor at the University of Maryland.
Phone: 301.405.2668
Email: basili@cs.umd.edu

2.Dr.Phyllis Frank ,Ph.D

Professor in the Department of Computer and
Information Science at Polytechnic University,
Phone: (718) 260-3870
Email: pfrankl@poly.edu

NATIONAL

- 1 .Y Raghu Babu Reddy, Ph.D, Associate Professor,IIIT, Hyderabad,
email-id: raghu.reddy@iiit.ac.in
2. *Dr. Neelima Satyam, ph.D*, Associate Professor with IIIT-HT:04066531000
Email-id: jayadev@iiit.ac.in

REGIONAL

- 1.Mr.P.Prakash,Asst.Prof,Dept of CSE,JBIT,Hyd.
- . 2.Mr.kishore Babu,Asst prof,Malla Reddy Engineering college,Hyd

9. Project/Consultancy/R&D Proposals :

PROPOSAL FOR R & D ACTIVITY:

Software Testing Strategies and Techniques

Sheetal Thakare¹, Savita Chavan², Prof. P. M. Chawan³

Abstract: This paper discusses different software strategies such as unit testing, integration testing, validation testing and techniques such as white box and black box testing for conventional and object oriented software development. Strategy provides a guideline to conduct testing. The testing carried out at different stages in software development life cycle is described by the various testing methods.

❖ **An article/white paper from a magazine /journal/weekly/any periodical in the entitled Subject**

Test Result Reporting

Abstract:

This paper is intended to highlight the importance of result reporting in the context of software testing. Result reporting can be at various stages of testing like system, integration etc... This

document addresses some of the areas of result reporting at a high level involving independent

system testing (Black box testing) keeping in mind, the customer as one of the audience.

PROPOSAL FOR R&D

Title: Bug Tracking for Improving Software Reliability (BTS)

Abstract: Bug Tracking for Improving Software Reliability (BTS) is an automated system that can be useful to employees and the managers in any functional organization. Bug Tracking System gives the facility to define the tasks in the organization and also allows the managers to track the bugs spent by the employee for that particular task. A report generation facility is supported in BTS that allows the managers to analyze the skills utilized by the employee and those which are not utilized. This tool can help managers for Bug estimation per project or application. This tool helps employees to document their Bugs and analyze

This project aims at the creation of a Bug Tracking System. This project will be accessible to all developers and its facility allows developers to focus on creating the database schema while letting the application server define table based on the fields in JSP and relationships between them. This system provides the following facilities.

The objectives of this system are:

To keep track of employee skills and based on the skills assigning of the task is done to an employee. Employee does bugs capturing. It can be done on daily basis. Various Reports are generated by this System for an employee and as well as to a manager

PROPOSAL FOR PROJECT ACTIVITY

Title: University Admission System

Abstract

The University Admission System is to automate the complete admission system starting from the notification to admission process. It also includes the Statistical reports on daily activities in the admission process.

The system enables online admissions saving the time of the geographically scattered students. It enables reducing time in activities, centralizes data handling and allows paperless admission with reduced manpower. It improves the operational efficiency and reduces the cost. It also provides consistent view of data and helps to integrate with other institutions for verification of marks and details.

Scope

- Create different system users and assign different roles with related permissions.
- The Notification will be released both online and manually.
- Students can apply online or offline.
- Verification of marks/details is done by interacting with different agencies like school boards or college management systems.
- Hall tickets will be issued for eligible candidates.
- Ranks will be generated depending upon the marks obtained for qualified candidates and call letters will be sent to short listed candidates.
- Admissions will be done online, saving the time of students.
- Centralized data is maintained.
- The management of seats will be done by the Convener online and availability of seats is announced.
- Statistical reports on daily activities in the admission process will be generated.
- The system provides security through password authentication. The Administrative staff and system administrators have their own login ids and these ids help in maintaining the integrity of the system.

Proposal for Consultancy Task / Project Design & Development

I. Project Title: e-Cops

Description:

To deliver next generation police and law enforcement reporting tools, and setting up intelligence platforms that agencies use to take incoming incident reports, lessen live employee resources and allow these enforcement agencies to reallocate resources to much needed community areas.

PURPOSE OF THE SYSTEM

This system provides crime details which occurred all over India. This information is useful for government as well as for public. Anyone can see what type of crime rate is more in a particular state and using that information action can be taken to avoid it.

Functional Requirements/ Scope of the project

- i.** Administrator should be able to create/edit a virtual police station (PS) which represents a real police station as a first time setup.
- ii.** Appointing of police to a particular police station which is present in a specific zone or to a specific district as a first time setup; he should be transferable at later time.
- iii.** PS should have areas of control which can be modified at a later time.
- iv.** Police station has several departments like Law and Order, Women Protection, Cybercrime, Traffic and control, CBI, etc. Separate module for each dept would be needed.
- v.** When a complaint is made it undergoes various processes like FIR, Charge Sheet, Property Seizure, Court Disposal etc all these activities are performed by a PS.
- vi.** Maintaining the criminal information state wise/area wise/age wise is mandatory.
- vii.** Sharing of case details with PS in other states is needed.
- viii.** Communication between officers is mandatory through Mails, chat.
- ix.** Secured registration of citizens is needed where they need to provide proof of citizenship, which will be cross checked by the police officer of that area.

No. of Modules

1. User
2. Police
3. Administrator
4. Authentication
5. Registration
6. Search

Module Description:

Administrator: Administrator is the owner of the site. He can add police station and as well as adds a police for that police station. He will accept/reject the user request. After observing the user details he will accept user request then only a user can login.

Police: Police is the person who add the crimes and criminal details to this site. He can maintain the present running details and he can add investigation report for any new complaint.

User: User is the person who is interested to watch criminal details as it may be required for his/her studies. He can search criminal details through various ways like by date, name, crime type etc.

Searching: Different searching techniques are provided here for searching the criminal details.

Authentication: Authentication is the process of granting or denying access to a Web based Application. It is the process of determining whether someone or something is, in fact, who or what it is declared to be. Authentication is commonly done through the use of credentials i.e. user name and password.

Registration: The system has a process of registration. Every user needs to submit his complete details including user name and password in the form of registration. Only after the user registration is completed, a user can log in to the system by using his user id and password.

II. Project Title: TRAINING AND PLACEMENT CELL PORTAL.

Abstract:

Training and Placement Cell is a total management and informative system, which provides the up-to-date information of all the students in a particular college. **TPC** helps the colleges to overcome the difficulty in keeping records of hundreds and thousands of students and searching for a student eligible for recruitment criteria from the whole thing. It helps in effective and timely utilization of the hardware and the software resources.

The home page contain various links such as links to login, various services like Events happened, achievements and recruiter details etc.,. The administrator will create the users and the users will use the accounts created by administrator. When the user enters into his respective page, he can update his details, and the details are to be approved by the administrator. All the users have some common services like changing password, updating details, searching for details, checking the details, mailing to administrator, and reading the material uploaded by admin if the user is a student. Administrator has the services to add events and achievements and he can reply to the mails sent by users. He can upload materials, search for student details, and he has the right to approve the students.

This package is developed in windows platform. The programming language used is JSP with three tier architecture. Oracle 8i is used as backend database for the details to be stored.

10. PROPOSALS

Relevant for Industrial Visit Proposals

(VIP= Visit of Industrial Proposals)

If students visited different software development companies like Data analytixds, Redhat Inc, fedora inc.

| S.No | Type of industry | Nature of industry | Industry name | Date of visit | No. of students participated | Year/Branch | Remarks |
|------|------------------|--------------------|---------------|---------------|------------------------------|--------------------------|---------|
| 1 | MNC | Software industry | CIGNITI | 28/09/2014 | 120 | B.Tech IV year/I sem CSE | |
| 2 | MNC | Software industry | MIND Q | 10/10/2014 | 120 | B.Tech IV year/I sem CSE | |
| 3 | | | | | | | |

11. GUIDELINES FOR SHADOW ENGINEERING (VIP)

INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM):

OBJECTIVES OF SHADOW ENGINEERING:

1. The program which uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

**TABLE 2: INDUSTRIAL TRAINING (Shadow Engg)
(Career Visit Approval)**

| S.No | Name of the Course | Nature of industry | Duration of Training | Authority | Date of Training/Certificate No. | Remarks |
|------|--------------------------------|--------------------|----------------------|-----------|----------------------------------|---------|
| 1 | Software testing methodologies | MNC | 1 WEEK | Director | | |
| | | | | | | |

12. CALIBRATION/INSTALLATION AND TESTING:

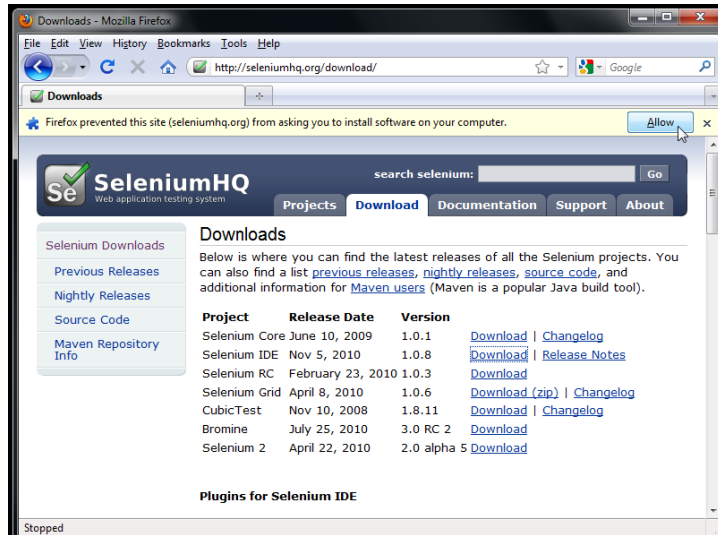
Calibration: There is no need to calibrate any equipment for this concerned lab.

Installation: Steps for downloading and installation of Selenium IDE

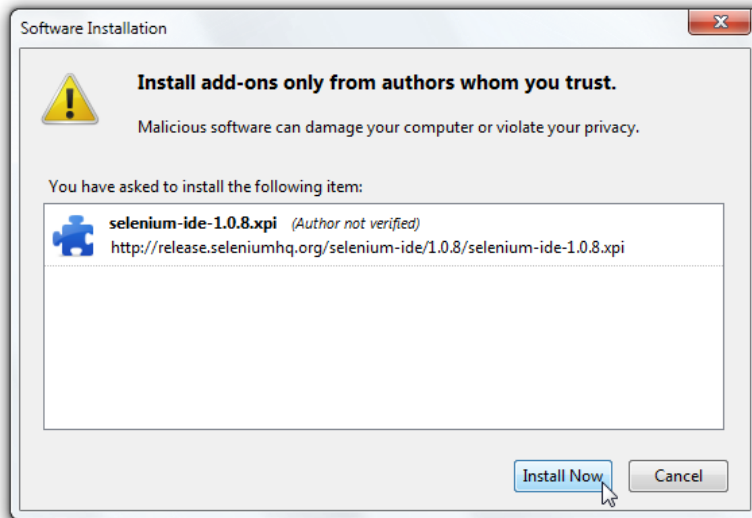
Installing the IDE

Using Firefox, first, download the IDE from the Selenium HQ [downloads page](http://seleniumhq.org/download/)

Firefox will protect you from installing add-ons from unfamiliar locations, so you will need to click ‘Allow’ to proceed with the installation, as shown in the following screenshot.

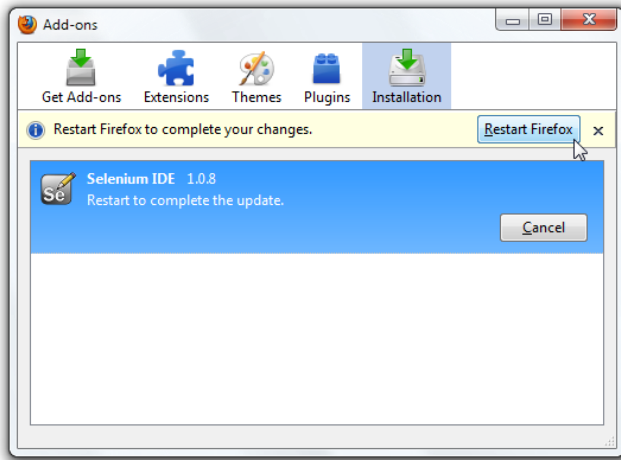


When downloading from Firefox, you'll be presented with the following window.

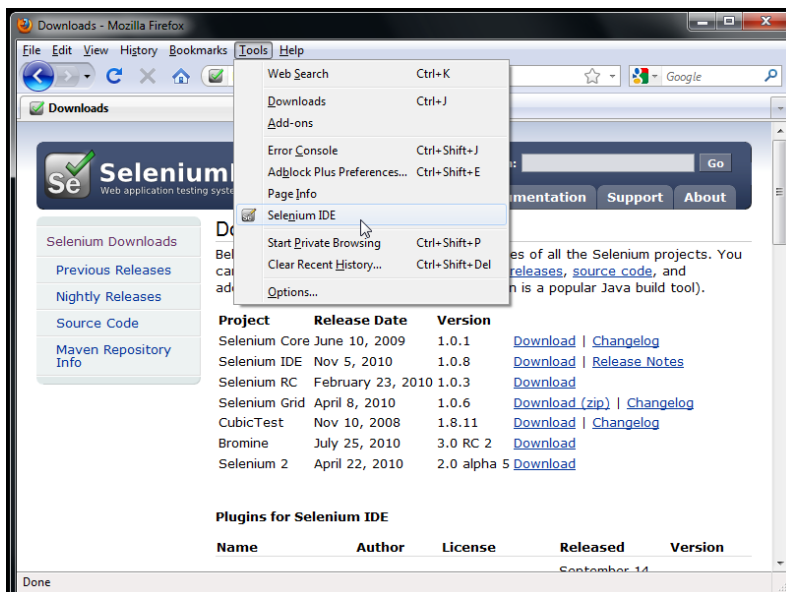


Select Install Now. The Firefox Add-ons window pops up, first showing a progress bar, and when the download is complete, displays the following.

LIFT MANUAL



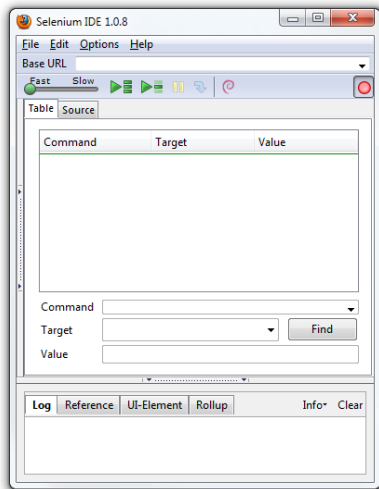
Restart Firefox. After Firefox reboots you will find the Selenium-IDE listed under the Firefox Tools menu.



Opening the IDE

To run the Selenium-IDE, simply select it from the Firefox Tools menu. It opens as follows with an empty script-editing window and a menu for loading, or creating new test cases.

LIFT MANUAL



Testing : Aim of this concept is to test the software after installation whether it meets the existing standards.

How to Use Selenium IDE?

The Selenium IDE is currently available only on Firefox. Chrome users might want to look at alternatives like iMacros, Sahi and Scirocco.

First of all, install Selenium IDE in Firefox by grabbing a copy of the software from the Selenium HQ. It'll install as a set of 5 extensions with separate formatters for Python, Ruby, etc. (language formatters are optional; install formatters only for languages you want to use). Once installed, restart Firefox. If you now go to the main Firefox options menu -> Add Ons -> Extensions, you should see Selenium IDE along with language formatters for Ruby, Python, C# and Java.

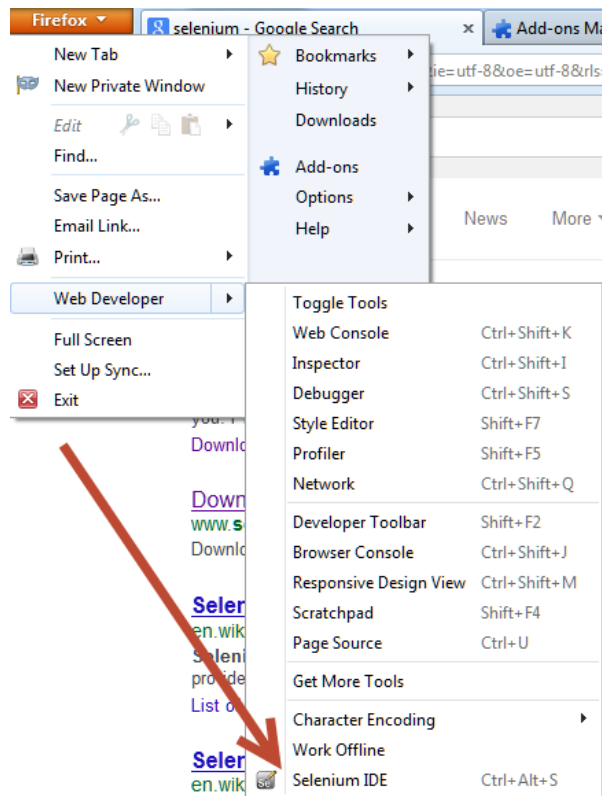


You are now ready to use Selenium. Below, we'll learn how to record and play back actions using Selenium.

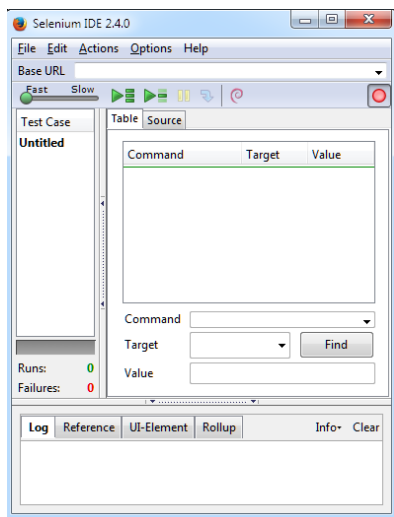
Using Selenium

To use the Selenium IDE, go to Firefox -> Web Developer -> Selenium IDE.

LIFT MANUAL



Alternatively, press CTRL + ALT + S to open the Selenium IDE



window.

Let's take a closer look at the Selenium IDE window:

- The left pane lists all the open test cases.
- The 'Base URL' shows the URL the test is being performed on.
- The 'Table' pane will show a list of actions recorded as part of the test. You can see how this works in the example below.
- The 'Fast-Slow' slider dictates how fast/slow playback for the recorded test should be.

- The 'Play' button plays back the test.
- The big red button at the rightmost edge is the 'Record' toggle switch. Switch it on to record interactions on any website.
- The bottom pane shows a log of actions and a reference list.

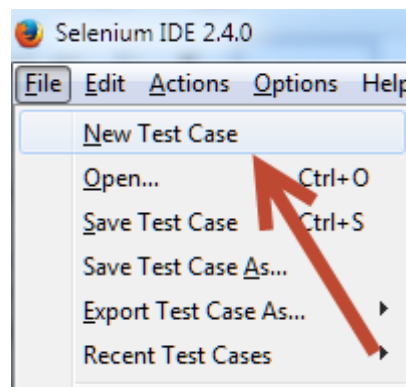
There are a number of other options – Selenium IDE is a fairly powerful tool, after all – but for now, we only need to be concerned with the options listed above.

New to software testing? [Learn the basics of technical web testing](#) in this course!

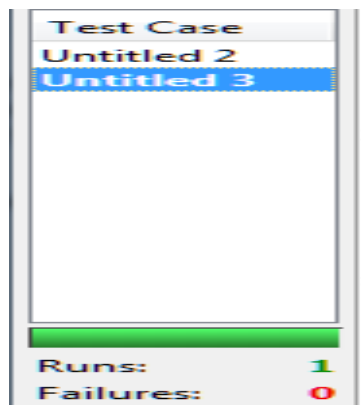
Creating a Simple Test in Selenium IDE

For the purpose of this tutorial, we will create a simple test that will enter a particular query into Google. In essence, this will be more like a macro than an actual 'test', but it will demonstrate the power of automated testing using Selenium IDE.

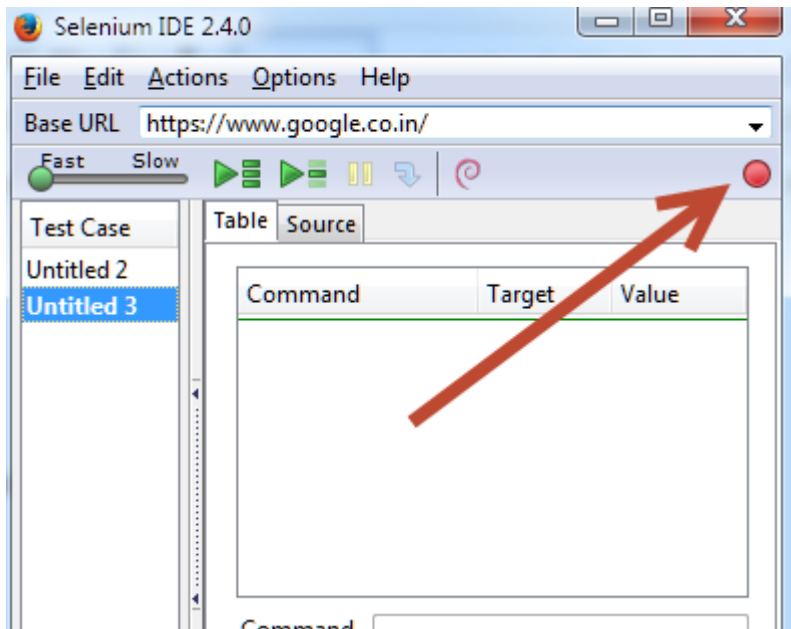
Step 1: Create a new test case by going to File -> New Test Case.



A new untitled test should show up in the test case pane to the left.

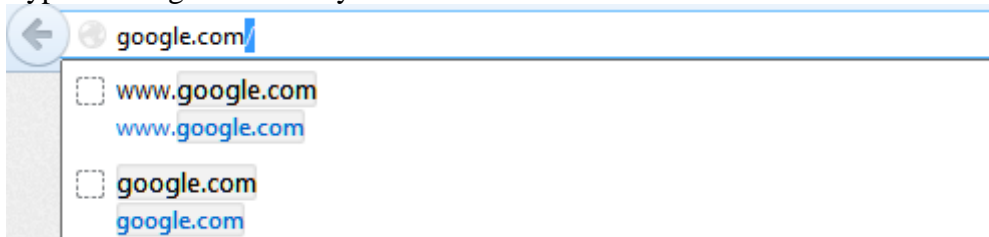


Step 2: We are now going to record our test. To do this, hit the red record switch to the right.



Now, perform the following actions in sequence:

1. Type in Google.com into your Firefox address bar:



- 2.

2. On the Google.com homepage, search for any query. For this example, we used the query 'automated testing'.

3. Once the results page opens, stop the test by clicking the red toggle switch.

Step 3: Now that we've recorded the test, let's play it back.

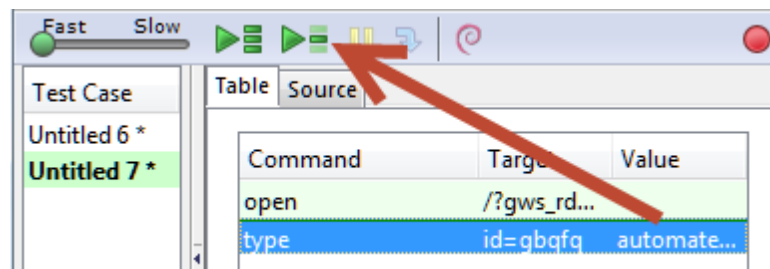
This is what your 'Table' pane should look like now:

LIFT MANUAL

| Command | Target | Value |
|---------|-------------|-------------|
| open | /?gws_rd... | |
| type | id=gbqfq | automate... |

| | | |
|---------|-------------------|------|
| Command | type | |
| Target | id=gbqfq | Find |
| Value | automated testing | |

Click on the 'Play current test case' button.



This will automatically open Google.com, type in 'automated testing' into the query field, and display the search results.

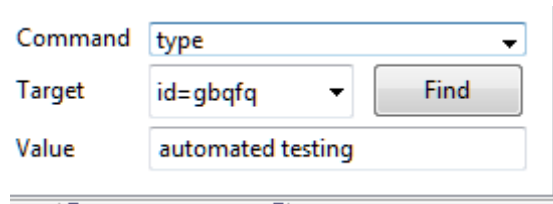
If you take a closer look at the two commands in the Table pane, you'll see that you can not only change the command, but also alter the values associated with it.

For example, the first command is 'open'. You can change this to anything else – Selenium IDE offers a huge list of built-in commands. You can also change the 'Target' url to anything that you like.

| | | |
|---------|-------------------|------|
| Command | open | |
| Target | /?gws_rd=cr&ei=() | Find |
| Value | | |

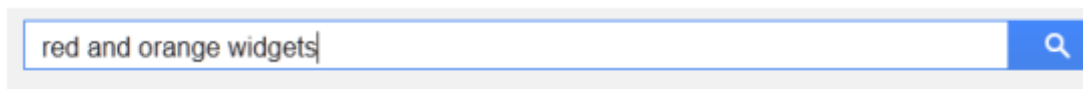
The next command is 'type'. This basically instructs Selenium to type-in the desired query into the search bar. You can change the query to anything you want by changing the 'value'.

LIFT MANUAL



Command type
Target id=gbqfq Find
Value automated testing

For example, if you change it to ‘red and orange widgets’, Selenium IDE will automatically query ‘red and orange widgets’ into Google.



red and orange widgets

This is one of the most basic examples of what Selenium IDE can do. By combining different commands, you can make the IDE run complex tasks and test for bugs automatically.

The list of hardware/software for testing:

Necessity of tools for development and testing such as .net, Java and Testing Tool (e.g. Win runner), Test Management Tool (e.g. Test Director) Bug Tracking Tool (e.g. Bugzilla, bugbit), Web Testing tool (e.g. Selenium)

13. PREVENTIVE MAINTAINANCE SCHEDULES

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

| S.No. | Name of the Equipment | Date of Maintenance | Type of Activity | Remarks |
|-------|-----------------------|---------------------|------------------------------|---------|
| 1 | System | | Testing software's installed | |
| | | | | |
| | | | | |
| | | | | |

(2) Maintenance Reports duly signed by in charges as well as HODs and duly approved by Principal periodically.

14.TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S.No., Equipment Name , Type of Problem (Abnormal Sound, Corrupt Software, Anti Virus Problem, Missing of Display, CRT not working,), Expected Reasons .Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Equipment Name | Type of Trouble | Remedial Activity | Remarks |
|--------------|-----------------------------------|-----------------------|------------------------|--------------------------|----------------|
| | | | | | |

LINUX PROGRAMMING LAB

CONTENTS:

- 1. OBJECTIVES AND RELEVANCE**
- 2. SCOPE**
- 3. PREREQUISITES**
- 4. SYLLABUS AS PER JNTUH**
- 5. LAB SCHEDULE**
- 6. SUGGESTED BOOKS**
- 7. WEBSITES (USEFUL LINKS)**
- 8. EXPERT DETAILS**
- 9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D**
- 10. PROPOSALS**
- 11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
- 12. CALIBRATION, TESTING AND INSPECTION**
- 13. PREVENTIVE MAINTENANCE SCHEDULES**
- 14. TROUBLESHOOTING**

1. OBJECTIVES AND RELEVANCE

The main objective of the lab course is to gain practical knowledge of Linux utilities (Commands), shell script and client server communication using sockets and multithread programming.

2. SCOPE

The main scope of Linux Programming is development of any application with Linux platform in a highly secured manner by using Linux commands and shell script. In addition with provides inter process communication using system v ipc

3. PREREQUISITES

This lab recommends continuous practice of various console/window based operating systems. It needs requisite knowledge about basics of Disk Operating System commands and GUI of window based operating system.

4. LINKING OF LINUX PROGRAMMING SUBJECT WITH LAB EXPERIMENTS

Preamble: This lab covers the Linux shell script and C-programs in Linux environment. The JNTUH has given 28 lab programs in the syllabus out of which first nine programs are based on the shell script and rest of all based on C programs.

1. Write a Shell Script that accepts a file name, starting and ending line numbers as arguments and displays all lines between the given line numbers.(Unit –1)

2. Write a shell script that deletes all lines containing a specific word in one or more files supplied as arguments to it. (Unit-1)

3. Write a shell script that displays a list of files in current directory to which the user has read, write and execute permissions. (Unit -2)

4. Write a shell script that receives any number of file names as arguments, checks if every argument is a file or directory, and when it is a file, reports the no. of lines in it.(Unit -2)

5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.(Unit-2)

6. Write a shell script to list all of the directory files in a directory. (Unit -2)

7. Write a shell script to find factorial of a given number. (Unit-2)

Objectives: The main objective of the above 7 programs is to enable the student to learn the various shell Meta characters, programming concepts like control structures and loops.

Prerequisites: Requires knowledge of basic syntax of control structures and loops in shell programming.

Application: Students can develop various applications file management system.

8. Write awk script to count no. of lines in a file that do not contain vowels. (Unit -1)

9. Write awk script to count no. of characters, words, lines in a given file. (Unit -1)

Objectives: The main objective of the above 2 programs is to enable the student to learn the various awk concepts like table creation and insert/delete/update/retrieve the records from it.

Prerequisites: Requires knowledge of database commands like create, update, insert and delete.

Application: The student can develop various applications like database creation and perform various add/remove operations.

10. Write a C program that makes a copy of a file using standard I/O and System calls.(Unit -3)

11. Implement in C the following UNIX commands using system calls:

A.cat B.ls C.mv (Unit 3)

12. Write a program that takes one or more file/directory names as command line input and reports the following information on the file:

A. file type B. Number of links C. Time of last access D. read, write and execute permissions. (Unit 2)

13. Write a C program to implement ls -l command. (Unit 3)

14. Write a C program to list every file in a directory, node number and file name. (Unit 3)

15. Write a C program to demonstrates redirection of standard output to a file ls>f1 (Unit 3)

16. Write a C program to create a child process and allow the parent to display parent and child to display child on screen. (Unit 3)

Objectives: The main objective of the above 7 programs is to enable the student to execute the various command by using C programs.

Prerequisites: Requires knowledge of C language.

Application: To learn basic file manipulation operations by using programming instead of commands like create, copy and remove.

17. Write a C program to create Zombie process (Unit 5)

18. Write a C program to create Orphan process. (Unit 5)

Objectives: The main objective of the above 2 programs is to enable the student to learn the various process formats.

Prerequisites: Requires knowledge of process state diagram and functions like fork() and vfork().

Application: To apply the concepts in process scheduling algorithms.

19. Write a C program that illustrates how to execute two commands concurrently with a command pipe (Unit 5)

20. Write a C program that illustrates communication between 2 unrelated processes using named pipe. (Unit 5)

21. Write a C program to create a message queue with read and write permissions to write 3 messages to it with different priority numbers. (Unit -5)

22. Write a C program that receives the messages from message queue and displays them. (Unit 6)

23. Write a C program to allow cooperation processes to lock a resource for exclusive use using a) semaphores b) flock or lockf system calls (Unit 6)

24. Write a C program that illustrates suspending and resuming processes using signals. (Unit 4)

25. Write a C program to implement producer-consumer Problem using two processes. (Unit 5)

Objectives: The main objective of the above 7 programs is to enable the student to learn the various inter process communication.

Prerequisites: Require knowledge about basics of IPC.

Application: To apply this concept in development of applications like reservation systems and retail marketing software.

26. Write a Client-Server program using UNIX domain Sockets. (Unit 8)

27. Write a Client-Server program using internet domain sockets. (Unit 8)

28. Write a C program that illustrates 2 process communicating using shared memory.(Unit 6)

Objectives: The main objective of the above 3 programs is to enable the student to learn the various remote inter process communication.

Prerequisites: Requires knowledge about basics of IPC.

Application: Applicable in development of ARP and DHCP protocols

LEAD PROGRAM

29. Write a shell Script to create 100 users as named from user1 to user100.

30. Secure copy between processes

Objectives: The main objective of the above 2 programs is to enable the student to learn the instant creating of users and to securely copy the contents from one user to other.

Prerequisites: Requires knowledge about basics of Unix Commands like adduser and password.

Application: Apply in Linux operating system application like human resource management application with high security.

5 (A) LAB SCHEDULE:

CYCLE 1:

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------------|-----------------------|-----------|------------|-----------|----------|-------------|--------|-----------|
| B1(501 to 512) | Demo & Basic commands | Exp.1,2&3 | Exp.4 ,5&6 | Exp.6,7&8 | Exp.9&10 | Exp.11 & 12 | test | Exp.13&14 |
| B2(513 to 524) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B3(525 to 536) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B4(537 to 548) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B5(549 to 560) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |

LIFT MANUAL

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week-16 |
|----------------|------------|---------------|-----------|----------------|------------|-------------|----------|---------|
| B1(501 to 512) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |
| B2(513 to 524) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |
| B3(525 to 536) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |
| B4(537 to 548) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |
| B5(549 to 560) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |

CYCLE 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------------|-----------------------|-----------|------------|-----------|----------|-------------|--------|-----------|
| B6(561 to 572) | Demo & Basic commands | Exp.1,2&3 | Exp.4 ,5&6 | Exp.6,7&8 | Exp.9&10 | Exp.11 & 12 | test | Exp.13&14 |
| B7(573 to 584) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B8(585 to 596) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B9(597 to 5A8) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B10(5A9 to 5C0) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week 16 |
|-----------------|------------|---------------|-----------|----------------|------------|-------------|----------|---------|
| B6(561 to 572) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |
| B7(573 to 584) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |
| B8(585 to 596) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |
| B9(597 to 5A8) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |
| B10(5A9 to 5C0) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |

4.(B) VIVA SCHEDULE: The viva schedule should be planned prior starting to the lab experiment.

CYCLE 1:

ROUND - 1

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|
| B1,B2,B3 | viva | | | | | viva | | | | | viva | | | |
| B4,B5,B1 | | viva | | | | | viva | | | | | viva | | |
| B2,B3,B4 | | | viva | | | | | viva | | | | | viva | |
| B5,B1,B2 | | | | viva | | | | | viva | | | | | viva |
| B2,B3,B4 | | | | | viva | | | | | viva | | | | |

LIFT MANUAL

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 |
|-----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|
| B6,B7,B8 | viva | | | | | viva | | | | | viva | | | |
| B9,B10,B6 | | viva | | | | | viva | | | | | viva | | |
| B7,B8,B9 | | | viva | | | | | viva | | | | | viva | |
| B10,B6,B7 | | | | viva | | | | | viva | | | | | viva |
| B8,B9,B10 | | | | | viva | | | | | viva | | | | |

CYCLE 2:

ROUND - 1

ROUND - 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|
| SG1 | viva | | | | | | | | | |
| SG2 | | | viva | | | | | | | |
| SG3 | | | | | viva | | | | | |
| SG4 | | | | | | | viva | | | |
| SG5 | | | | | | | | | viva | |

*SG: Selected Group with a maximum of 6 or 12 students

(C) Scheme of Evaluation:

The scheme of evaluation for internal and external exams as follows:

| LAB INTERNAL: | | | | | | | |
|---------------------------------|---------------------------------|--------------------------------------|----------------|----------------|-------------------------|--------------------------------|----------------|
| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-4 | Marks-3 | Marks-3 |
| Total Marks-25 | | | | | | | |

| LAB EXTERNAL: | | | |
|-----------------------|---|--|---|
| S.NO | Write up | Final Evaluation | Viva |
| 1 | 1. Aim 2. Procedure 3. Program 4. Expected output. | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

6. SUGGESTED BOOKS:

- I) Advanced UNIX programming by B.Venkateswarlu PHI publications 3rd edition.
- ii) Unix System Programming using C++, T.Chan, PHI
- iii) UNIX Concepts and Applications, 4th Edition, Sumitabha Das, TMH.

7. WEBSITES

6. www.advancedlinuxprogramming.com/.../advanced-linux-programming.
7. fita.hua.edu.vn/pttien/Setups/.../beginning-linux-programming.pdf
8. www.cse.hcmut.edu.vn/~hungnq/courses/nap/alp.pdf
9. <https://sites.google.com/site/linuxlabjntu09/home>
5. <http://youtube.com/linux>
6. <http://www.nptel.com/computerscience/Linuxprogramming>
7. <http://nptel.iitm.ac.in/courses.php?disciplineId=106>
8. <http://manuals.bioinformatics.ucr.edu/home/linux-basics>

8. EXPERTS DETAILS

The expert details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally.

INTERNATIONAL

1. Dr. Ken Thompson M.S.Ph.D,
2. Dr. Dennis Ritchie Ph.D Bell Labs, USA
3. Dr.W.R.Stevens , University of Machigan, USA.

NATIONAL

1. Dr. Anuksha Gupta LabVIEW Professional Development System for Linux
2. Dr. krishnan Venkateswaran: General Manager - 4G Telecom Protocol Development at IBM India.

REGIONAL

4. Mohammed Ahsan Siddiqui, Assoc. Prof, IT Dept. BIET.Hyderabad.
5. Mr.S.Madhu Assoc.Prof , CSE Dept.GNEC, Hydearabad
6. Prof.P. Viswapathi, Pro& HOD, Dept. of CSE, CMREC,Hyderabad.

9. MAPPING LAB WITH PROJECT/CONSULTANCY/R&D PROPOSALS

Proposal 1: Project Design & Execution

Proposal 2: R& D Level Project Design & Execution

Proposal 3: Consultancy Task / Project Design & Development

Proposal 1: Project Design & Execution

Title: Data communication over LAN with platform independent through Remote Procedure Calls (RPC)

ABSTRACT:

Remote Procedure Call (RPC) is a powerful technique for constructing distributed, client/server based applications. This project is to explain RPC mechanism, how to use the Sun RPC convert local procedures to remote procedures

Proposal 2: R& D Level Project Design & Execution

Title: Process to Process Load Balancing for Linux using Kappa system

Abstract: Process migration is the act of transferring a process between two computers. If used effectively, process migration could be used to improve the throughput of a cluster of computers. To perform this, a process must be migrated from a "slow" system to a "fast" system. To measure the "fastness" of a system, various parameters must be considered, such as the current load of the system, the time slices expected to be given to a new process, and the speed of the hardware.

In our system, named Kappa, the various parameters are aggregated together to form the system metric of a system. This metric is then used to determine which computer the process should be migrated to. In order to perform this, load balancing needs to be performed. Load Balancing in Kappa is performed in a peer-2-peer network in two steps: An initial approximation stage, and a probing stage. In the approximation stage, a rough measure of the distribution of the system metric is calculated within the cluster. In the probing stage, the initial information collected is used to direct a probe across the network to find the optimal target for load balancing. The actual process migration takes places transparently once the target machine is found.

Kappa was implemented in Linux, and allows for transparent process migration between paired Linux computers. It is implemented as a user-space program, with minimal hooks in the system calls. Kappa was tested for a cluster of Linux computers with varying system metric, and an improvement of 50% to 200% in execution time was recorded for CPU intensive processes.

Proposal 3: Consultancy Task / Project Design & Development

Linux Containers and the Future Cloud

Abstract: Linux-based container infrastructure is an emerging cloud technology based on fast and lightweight process virtualization. It provides its users an environment as close as possible to a standard Linux distribution. As opposed to para-virtualization solutions (Xen) and hardware virtualization solutions (KVM), which provide virtual machines (VMs), containers do not create other instances of the operating system kernel. Due to the fact that containers are more lightweight than VMs, you can achieve higher densities with containers than with VMs on the same host (practically speaking, you can deploy more instances of containers than of VMs on the same host). Another advantage of containers over VMs is that starting and shutting down a container is much faster than starting and shutting down a VM. All containers under a host are running under the same kernel, as opposed to virtualization solutions like Xen or KVM where each VM runs its own kernel. Sometimes the constraint of running under the same kernel in all containers under a given

host can be considered a drawback. Moreover, you cannot run BSD, Solaris, OS/x or Windows in a Linux-based container, and sometimes this fact also can be considered a drawback.

The idea of process-level virtualization in itself is not new, and it already was implemented by Solaris Zones as well as BSD jails quite a few years ago. Other open-source projects implementing process-level virtualization have existed for several years. However, they required custom kernels, which was often a major setback. Full and stable support for Linux-based containers on mainstream kernels by the LXC project is relatively recent, as you will see in this article. This makes containers more attractive for the cloud infrastructure. More and more hosting and cloud services companies are adopting Linux-based container solutions. In this article, I describe some open-source Linux-based container projects and the kernel features they use, and show some usage examples. I also describe the Docker tool for creating LXC containers

10. PROPOSALS

Relevant IV(Industrial Visit) (VIP= Visit of Industrial Proposals)

If students are visited different software development companies like Data analytixds, Redhat Inc, fedora inc.

| S.No | Type of industry | Nature of industry | Name of the company | Date of visit | No. of students participated | Year/branch | Remarks |
|------|------------------|----------------------|-----------------------------|---------------|------------------------------|-------------|---------|
| 1 | MNC | Software Development | Data analytixds | | | | |
| 2 | MNC | Net working | Cisco Systems India pvt Ltd | | | | |
| 3 | MNC | Network Security | Redhat Inc | | | | |
| 4 | MNC | Network Security | Fedora Inc | | | | |

11. GUIDELINES FOR SHADOW ENGINEERING (VIP)

INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM):

OBJECTIVES OF SHADOW ENGINEERING:

1. The program uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

**TABLE 2: INDUSTRIAL TRAINING (Shadow Engg)
(Career Visit Approval)**

| S.No | Name of the Course | Nature of industry | Duration of Training | Authority | Date of Training/Certificate No. | Remarks |
|------|--------------------------|--------------------|----------------------|------------|----------------------------------|---------|
| 1 | Linux socket programming | Linux Admin | 2days | Fedora Inc | 14-08-2014 | |
| 2 | Multithread engineering | Shell script | 2days | Data Quest | 28-08-2014 | |
| 3 | | | | | | |

12. Calibration & Testing

Calibration: There is no need to calibrate any equipment for this laboratory.

Installation: Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment

Linux OS Installation steps :(Red hat Linux GUI)

Step 1 – Insert the Red Hat Linux DVD into the DVD-drive of your computer. As soon as the following screen pops up, press ‘Enter’ to install Red Hat Enterprise Linux (RHEL) through GUI mode.

Step 2- RHEL installer would then prompt you conduct a check as to whether the CD media from which you’re installing is functioning correctly or not. Choose ‘Skip’, press enter and the installation would begin.

Step 3- Next, we need to select the language- English or any other language as per your preference, and then press ‘Next’.

Step 4- In this step, the RHEL installer would ask you about the appropriate type of keyboard for the system. We take the ‘US English’ keyboard; you can pick any other option depending on the type of your keyboard. Then press ‘Next’ to move to the next step.

Step 5- Next, the installer would ask for an ‘installation number’ if you wish to install full set of Red Hat functionalities. Enter the installation number and press ‘OK’ if you have an officially licensed installation number (for corporate clients that buy Red Hat’s backup support and full features).

Step 6- The Red Hat installer would then require you to create partitions in your computer’s hard disk for the installation. You can do it in four ways but the simplest way is to select ‘Use free space on selected drives and create default layout’ as this option will not affect any other OS residing in your system.

Step 7- In this step you must create the required system partitions and mount points such as ‘/boot’, ‘/home’, ‘swap’ etc which are required for the Linux’s proper functioning.

Step 8- This step pertains to the default OS that will be loaded by the GRUB loader

Step 9- This step pertains to the network settings of the Linux system that you are going to install. You can select the Ethernet devices through which the system would communicate with other devices in the network.

Step 10- The next step is to adjust the system clock to your particular time zone. Select your time zone and then click ‘Next’.

Step 11 – This is a very important step that deals with the root (super-user) password for the system. Type the password and confirm it and then click next.

Step 12 – The RHEL installer would then prompt you about if you wish to install some extra ‘Software Development’ or ‘Web Server’ features. By default, keep it at ‘Customize later’ and press ‘Next’.

LIFT MANUAL

Step 13- This next step will initiate the installation of Red Hat Linux, press 'Next' to begin the process.

Step 14- Upon the completion of installation you should the following screen. Press Reboot and you'd be ready to use your newly installed Red Hat Linux OS.

Testing: able to test the Linux utilities by using advanced testing tools like Quick Test Professional (QTP).

13. .PREVENTIVE MAINTENANCE SCHEDULES:

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

| S.No. | Name of the Equipment | Date of Maintenance | Type of Activity | Remarks |
|-------|-----------------------|---------------------|------------------|---------|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

(2) Maintenance Reports duly signed by in charges as well as HODs and duly approved by Principal periodically.

14. TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S.No., Equipment Name , Type of Problem (Abnormal Sound, Corrupt Software, Anti Virus Problem, Missing of Display, CRT not working,), Expected Reasons .Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Equipment Name | Type of Trouble | Remedial Activity | Remarks |
|-------|----------------------------|----------------|-----------------|-------------------|---------|
| | | | | | |
| | | | | | |
| | | | | | |

DATA MINING AND WARE HOUSING LAB

CONTENTS:

- 1. OBJECTIVES AND RELEVANCE**
- 2. SCOPE**
- 3. PREREQUISITES**
- 4. SYLLABUS AS PER JNTUH**
- 5. LAB SCHEDULE**
- 6. SUGGESTED BOOKS**
- 7. WEBSITES (USEFUL LINKS)**
- 8. EXPERT DETAILS**
- 9. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D**
- 10. PROPOSALS**
- 11. GUIDELINES FOR SHADOW ENGINEERING (VIP) AND INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
- 12. CALIBRATION, TESTING AND INSPECTION**
- 13. PREVENTIVE MAINTENANCE SCHEDULES**
- 14. TROUBLESHOOTING**

1. OBJECTIVES AND RELEVANCE

The main objective of the lab course is to gain practical hands on experience of data mining techniques using weka tool.

2. SCOPE

To make use of the weka tool for mining Credit Risk Assessment of an applicant in the German Credit Data set. Assessing the credit worthiness of an applicant is of crucial importance for banks. Using Weka data mining tool we have to develop a system to help a loan officer decide whether the credit of a customer is good or bad.

3. PREREQUISITES

Basic RDMS knowledge is required

PREAMBLE

This lab covers the experiments in DWDM subject. The JNTU has given 8 experiments in the syllabus.

4. SYLLABUS-JNTU

Credit Risk Assessment

Description: The business of banks is making loans. Assessing the credit worthiness of an applicant is of crucial importance. You have to develop a system to help a loan officer decide whether the credit of a customer is good or bad. A bank's business rules regarding loans must consider two opposing factors. On the one hand, a bank wants to make as many loans as possible. Interest on these loans is the banks profit source. On the other hand, a bank cannot afford to make too many bad loans. Too many bad loans could lead to the collapse of the bank. The bank's loan policy must involve a compromise. Not too strict and not too lenient. To do the assignment, you first and foremost need some knowledge about the world of credit. You can acquire such knowledge in a number of ways.

1. Knowledge engineering: Find a loan officer who is willing to talk. Interview her and try to represent her knowledge in a number of ways.
2. Books: Find some training manuals for loan officers or perhaps a suitable textbook on finance. Translate this knowledge from text to production rule form.
3. Common sense: Imagine yourself as a loan officer and make up reasonable rules which can be used to judge the credit worthiness of a loan applicant.
4. Case histories: Find records of actual cases where competent loan officers correctly judged when to and when not to approve a loan application.

The German Credit Data

Actual historical credit data is not always easy to come by because of confidentiality rules. Here is one such data set consisting of 1000 actual cases collected in Germany. In spite of the fact that the

LIFT MANUAL

data is German, you should probably make use of it for this assignment (Unless you really can consult a real loan officer!) There are 20 attributes used in judging a loan applicant (ie, 7 Numerical attributes and 13 Categorical or Nominal attributes). The goal is to classify the applicant into one of two categories, Good or Bad.

Subtasks:

1. List all the categorical (or nominal) attributes and the real valued attributes separately.
2. What attributes do you think might be crucial in making the credit assessment? Come up with some simple rules in plain English using your selected attributes.
3. One type of model that you can create is a Decision tree. Train a Decision tree using the complete data set as the training data. Report the model obtained after training.
4. Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly? (This is also called testing on the training set) why do you think we cannot get 100% training accuracy?
5. Is testing on the training set as you did above a good idea? Why or why not?
6. One approach for solving the problem encountered in the previous question is using cross-validation. Describe what is cross validation briefly. Train a decision tree again using cross validation and report your results. Does accuracy increase/decrease? Why?
7. Check to see if the data shows a bias against “foreign workers” or “personal-status”. One way to do this is to remove these attributes from the data set and see if the decision tree created in those cases is significantly different from the full data set case which you have already done. Did removing these attributes have any significant effect? Discuss.
8. Another question might be, do you really need to input so many attributes to get good results? May be only a few would do. For example, you could try just having attributes 2,3,5,7,10,17 and 21. Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones you want.)
9. Sometimes, The cost of rejecting an applicant who actually has good credit might be higher than accepting an applicant who has bad credit. Instead of counting the misclassification equally in both cases, give a higher cost to the first case (say cost 5) and lower cost to the second case. By using a cost matrix in weak. Train your decision tree and report the Decision Tree and cross validation results. Are they significantly different from results obtained in problem 6.
10. Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees? How does the complexity of a Decision Tree relate to the bias of the model?
11. You can make your Decision Trees simpler by pruning the nodes. One approach is to use Reduced Error Pruning. Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross validation and report the Decision Trees you obtain. Also Report your accuracy using the pruned model. Does your Accuracy increase?
12. How can you convert a Decision Tree into “if-then-else rules”? Make up your own small Decision Tree consisting 2-3 levels and convert into a set of rules. There also exist different

LIFT MANUAL

classifiers that output the model in the form of rules. One such classifier in weka is rules. PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one! Can you predict what attribute might be in this data set? OneR classifier uses a single attribute to make decisions(it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART, oneR.

The students will be given a demo on how to use the weka tool for solving the above tasks. The necessary manual and tutorial will be given.

5(A) LAB SCHEDULE:

CYCLE 1:

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|----------------|-----------------------|-----------|------------|-----------|----------|-------------|--------|-----------|
| B1(501 to 512) | Demo & Basic commands | Exp.1,2&3 | Exp.4 ,5&6 | Exp.6,7&8 | Exp.9&10 | Exp.11 & 12 | test | Exp.13&14 |
| B2(513 to 524) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B3(525 to 536) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B4(537 to 548) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B5(549 to 560) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week-16 |
|----------------|------------|---------------|-----------|----------------|------------|-------------|----------|---------|
| B1(501 to 512) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |
| B2(513 to 524) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |
| B3(525 to 536) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |
| B4(537 to 548) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |
| B5(549 to 560) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | Test |

LIFT MANUAL

CYCLE 2

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 |
|-----------------|-----------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| B6(561 to 572) | Demo & Basic commands | Exp.1,2&3 | Exp.4 ,5&6 | Exp.6,7&8 | Exp.9&10 | Exp.11 & 12 | test | Exp.13&14 |
| B7(573 to 584) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B8(585 to 596) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B9(597 to 5A8) | Demo & Basic commands | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |
| B10(5A9 to 5C0) | Demo & Basic | Exp.1&2 | Exp.3 &4 | Exp.5 &6 | Exp.7 &8 | Exp.9 & 10 | test | Exp.13&14 |

| Batches | week-9 | week-10 | week-11 | week-12 | week-13 | week-14 | week-15 | week-16 |
|-----------------|---------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| B6(561 to 572) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |
| B7(573 to 584) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |
| B8(585 to 596) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |
| B9(597 to 5A8) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |
| B10(5A9 to 5C0) | Exp.15 &16 | Exp.17,18 &19 | Exp.20&21 | Exp.22 ,23 &24 | Exp 25& 26 | Exp 27 & 28 | LEAD1 &2 | test |

B) VIVA SCHEDULE: The viva schedule should be planned prior to the lab experiment.

CYCLE 1:

ROUND - 1

| Batches | we ek- 1 | wee k-2 | wee k-3 | wee k-4 | wee k-5 | wee k-6 | wee k-7 | wee k-8 | wee k-9 | wee k-10 | wee k-11 | wee k-12 | wee k-13 | wee k-14 |
|----------|----------------|------------|------------|------------|------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|-------------|
| B1,B2,B3 | viva | | | | | viva | | | | | viva | | | |
| B4,B5,B1 | | viva | | | | | viva | | | | | viva | | |
| B2,B3,B4 | | | viva | | | | | viva | | | | | viva | |
| B5,B1,B2 | | | | viva | | | | | viva | | | | | viva |
| B2,B3,B4 | | | | | viva | | | | | viva | | | | |

| Batches | week-1 | week-2 | wee k-3 | week-4 | week-5 | wee k-6 | we ek- 7 | we ek- 8 | we ek- 9 | we ek- 10 | we ek- 11 | we ek- 12 | we ek- 13 |
|---------------|--------|--------|------------|--------|--------|------------|----------------|----------------|----------------|-----------------|-----------------|-----------------|-----------------|
| B6,B7,B8 | viva | | | | | viva | | | | | viva | | |
| B9,B10, B6 | | viva | | | | | viva | | | | | viva | |
| B7,B8,B9 | | | viva | | | | | viva | | | | | viva |
| B10,B6, B7 | | | | viva | | | | | viva | | | | |
| B8,B9,B10 | | | | | viva | | | | | viva | | | |

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|
| SG1 | viva | | | | | | | | | |
| SG2 | | | viva | | | | | | | |
| SG3 | | | | | viva | | | | | |
| SG4 | | | | | | | viva | | | |
| SG5 | | | | | | | | | viva | |

*SG: Selected Group with a maximum of 6 or 12 students

(C) Scheme of Evaluation:

The scheme of evaluation for internal and external exams as follows:

| LAB INTERNAL: | | | | | | | |
|--------------------------|----------------------|-------------------------------|---------|---------|------------------|---------------------|---------|
| Day to Day Evaluation-15 | | | | | Internal Exam-10 | | |
| Uniform | Observation & Record | Performance Of the Experiment | Result | Viva | Write up | Execution & Results | Viva |
| Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-3 | Marks-4 | Marks-3 | Marks-3 |
| Total Marks-25 | | | | | | | |

| LAB EXTERNAL: | | | |
|-----------------------|---|---|--|
| S.NO | Write up | Final Evaluation | Viva |
| 1 | 1. Aim 2. Procedure 3. Program 4. Expected output. | Based on correctness of the program and Results | Based on understanding of Experiment and theoretical questions in the related subjects |
| Marks | 20 | 20 | 10 |
| Total Marks-50 | | | |

6. SUGGESTED BOOKS

Data mining : practical machine learning tools and techniques Ian H. Witten

7. WEB SITES

1. Weka Reference, <http://www.gnu.org/copyleft/gpl.html>
2. Prefuse Visualization Toolkit. See <http://prefuse.org/> for more information on the project
3. WekaWiki – <http://weka.wikispaces.com/>
4. Extensions for Weka's main GUI on WekaWiki –
5. <http://weka.wikispaces.com/Extensions+for+Weka%27s+main+GUI>
6. Adding tabs in the Explorer on WekaWiki –
<http://weka.wikispaces.com/Adding+tabs+in+the+Explorer>
7. Explorer visualization plugins on WekaWiki –
<http://weka.wikispaces.com/Explorer+visualization+plugins>
8. <http://home.etf.rs/~vm/os/dmsw/Morgan.Kaufman.Publishers.Weka.2nd.Edition.2005>. Elsevier.pdf

8. EXPERTS' DETAILS

The expert details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally.

INTERNATIONAL

1. Jiawei Han, Abel Bliss Professor, [Department of Computer Science Univ. of Illinois at Urbana-Champaign](#)
E-mail: [hanj\[at\]cs.uiuc.edu](mailto:hanj[at]cs.uiuc.edu)
Web: www.cs.uiuc.edu/homes/hanj
2. Dr. Rakesh Agrawal,
1065 La Avenida, Mountain View, CA 94043
[rakesh\[dot\]agrawal \[at\] microsoft\[dot\]com](mailto:rakesh[dot]agrawal[at]microsoft[dot]com)
3. Tom Mitchell (Carnegie Mellon University)
E-mail Tom.Mitchell@cmu.edu
Webpage <http://www.cs.cmu.edu/~tom/>

NATIONAL

1. Sunita Sarawagi (Indian Institute of Technology, Bombay)
E-mail sunita [at] iitb.ac.in
Webpage <http://www.it.iitb.ac.in/~sunita/>
2. Raghu Ramakrishnan
E-mail raghu@microsoft.com
Webpage <http://pages.cs.wisc.edu/~raghu/>

REGIONAL

1. Prof. Kamal karlapalem, International Institute of Information Technology ,Hyderabad, India .
e-mail kamal@gmail.com
webpage <http://www.iiit.ac.in/~kamal/>
2. Dr. Vikram Pudi, Associate Professor International Institute of Information Technology ,Hyderabad,
e-mail vikram@iiit.ac.in
webpage <http://faculty.iiit.ac.in/~vikram/>

9. MAPPING LAB WITH PROJECT/CONSULTANCY/R&D PROPOSALS

Proposal 1: Project Design & Execution

Title1: To implement Naive Bayesian spam filter. This experiment uses naïve Bayesian classifier to classify whether an email is a spam or not. It helps the student understand how a classifier algorithm is actually working. It can also be used for doing other text classification task

Title2: To implement nearest neighbor classifier for making movie recommendations. This experiment uses nearest neighbor classifier to make movie recommendation to the user. It helps the student to understand how nearest neighbor algorithm can be applied to a real data.

The datasets for the above experiments will be provided or can be downloaded from the link <https://gist.github.com/entaroadun/1653794>

Proposal 2: R& D Level Project Design & Execution

Title: Prediction of methane production in waste water treatment facility: a data-mining approach

Abstract

A prediction model for methane production in a wastewater processing facility is presented. The model is built by data-mining algorithms based on industrial data collected on a daily basis.

LIFT MANUAL

Because of many parameters available in this research, a subset of parameters is selected using importance analysis. Prediction results of methane production are presented in this paper. The model performance by different algorithms is measured with five metrics. Based on these metrics, a model built by the Adaptive Neuro-Fuzzy Inference System algorithm has provided most accurate predictions of methane production.

Keywords

Methane production prediction, Wastewater treatment facility Data-mining algorithms, Neural networks, Adaptive neuro-fuzzy mode.

10. PROPOSALS

If students are visited different software development companies like IBM, Redhat Inc, fedora inc.

| S.No | Type of industry | Nature of industry | Name of the company | Date of visit | No. of students participated | Year/branch | remarks |
|------|------------------|----------------------|---------------------|---------------|------------------------------|-------------|---------|
| 1 | MNC | Software Development | Amdocs Pvt Ltd | | | | |
| 2 | MNC | Data Mining | IBM India Pvt Ltd | | | | |

11. GUIDELINES FOR SHADOW ENGINEERING (VIP)

INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM):

OBJECTIVES OF SHADOW ENGINEERING:

1. The program uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

12. CALIBRATION, TESTING AND INSPECTION

Calibration: There is no need to calibrate any equipment for this laboratory.

Installation: Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment

Download WEKA Tool from <http://www.cs.waikato.ac.nz/~ml/weka/index.html> or internet and install then we are getting this below GUI.



According to user requirement select any label or button then proceed.

13. PREVENTIVE MAINTAINANCE SCHEDULES:

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

| S.No. | Name of the Equipment | Date of Maintenance | Type of Activity | Remarks |
|-------|-----------------------|---------------------|------------------|---------|
| | | | | |
| | | | | |
| | | | | |

(2) Maintenance Reports should be duly signed by in charges as well as HODs and duly approved by Principal periodically.

14. TROUBLE SHOOTING SCHEDULES:

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S.No., Equipment Name , Type of Problem (Abnormal Sound, Corrupt Software, Anti Virus Problem, Missing of Display, CRT not working,), Expected Reasons.Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Equipment Name | Type of Trouble | Remedial Activity | Remarks |
|-------|----------------------------|----------------|-----------------|-------------------|---------|
| | | | | | |
| | | | | | |