CMR ENGINEERING COLLEGE

(Approved by AICTE- New Delhi, Affiliated to JNTUH)
Kandlakoya(V), Medchal Road, Hyderabad

Department of CSE-Data Science

LIFT Planner (Lab Improvement for Future Trends)

On

AI USING PYTHON LAB

2025-2026

BY

N. SUMANJALI

Assistant Professor

AI USING PYTHON LAB

Sl. No	CONTENT
1	OBJECTIVES AND RELEVANCE
2	SCOPE
3	PREREQUISITES
4	SYLLABUS AUTONOMOUS
5	LEAD EXPERIMENTS
6	LAB SCHEDULE
7	SUGGESTED BOOKS
8	WEBSITES (USEFUL LINKS)
9	EXPERT DETAILS
10	MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS
11	INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)
12	SHADOW ENGINEERING
13	CALIBIRATION , TESTING AND INSPECTION
14	MAINTAINANCE AND TROUBLESHOOTING
15	ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL
16	VIRTUAL LABS REFERENCE LINKS

1. OBJECTIVES AND RELEVANCE

The objective of an AI using Python lab typically revolve around giving students or practitioners hands-on experience with the practical applications of Artificial Intelligence (AI) techniques using Python.

2. SCOPE

The scope of AI using Python labs is vast, covering a wide range of topics and practical applications in the field of Artificial Intelligence. Python, being one of the most popular languages for AI development, provides an accessible yet powerful platform for learning, experimentation, and real-world implementation of AI models

3. PREREQUISITES

Course "PYTHON PROGRAMMING"

4. SYLLABUS (AUTONOMUS)

WEEK 1: Write a Program to Implement Breadth First Search using Python.

WEEK 2: Write a Program to Implement Depth First Search using Python

WEEK 3: Write a Program to Implement Tic-Tac-Toe game using Python

WEEK 4: Write a program in python to solve 8 Puzzle problems

WEEK 5: Write a program in python for Water jug problem

WEEK 6 Write a program in python to solve Traveling salesman problem

WEEK 7: Write a program in python to solve Tower of Hanoi

WEEK 8: Write a Program to Implement Monkey Banana Problem using Python.

WEEK 9: Write a program in python to solve Alpha-Beta Pruning problem

WEEK 10: Write a Program to Implement 8-Queens Problem using Python.

5. LEAD PROGRAMS

1. Write a Program to Implement Missionaries-Cannibals Problems using Python.
2. Write a python program to implement simple Chatbot?.

6. (A).LAB SCHEDULE:

CYCLE 1:

Batches	week-1	week-2	week-3	week-4	week-5	week-6
B1(6701 to 6736) &22-42	Demo & Introduction	Exp.1	Exp 2	Exp 3	Exp4	Exp 5
B2(6737 to LE05)	Demo & Introduction	Exp.1	Exp 2	Exp 3	ExP4	Exp 5

Batches	week-7	week-8	week-9	week-10	week-11	week-12	week-13
B1(6701 to 6736)&22-42	Exp.6	Exp.7	Exp.8	Exp.9	Exp 10	Lead 1	Lead 2
B2(6737 to LE04)	Exp.6	Exp.7	Exp.8	Exp.9	Exp 10	Lead 1	Lead 2

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

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	viva	viva	viva	viva	viva	viva	viva	viva	viva	viva	viva	viva
B2	viva	viva	viva	viva	viva	viva	viva	viva	viva	viva	viva	viva

C. Scheme of Evaluation:

LAB INTERNAL:								
Day to Day Evalution-10					**Internal Exam-30**			
Uniform	**Observation & Record**	**Performance Of the Experiment**	**Result**	**Viva**	**Write up**	**Execution & Results**	**Viva**	**Project**
Marks-2	**Marks-2**	**Marks-2**	**Marks-2**	**Marks-2**	**Marks-5**	**Marks-5**	**Marks-10**	**Marks - 10**

LAB EXTERNAL: MARKS : 60				
Write Up Of Program	**Experiment / Program**	**Evaluation Of Results**	**Another Program Presentation**	**Viva - Voce**
Marks-10	**Marks-15**	**Marks-15**	**Marks-10**	**Marks-10**

7. SUGGESTED BOOKS:

TEXT BOOKS:

1. Artificial Intelligence a Modern Approach, Third Edition, Stuart Russell and Peter Norvig , PearsonEducation.
2. Learn Python the Hard Way: 3rd Edition. ...

REFERENCE BOOKS:

1. Artificial Intelligence,3rdEdn,E.RichandK.Knight(TMH)
2. Artificial Intelligence,3rdEdn.,PatrickHennyWinston,PearsonEducation.
3. Artificial Intelligence, Shivani Goel, Pearson Education.
4. Python Programming: An Introduction to Computer Science (3rd Edition)
5. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.
6. Automate The Boring Stuff with Python, 2nd Edition: Practical Programming For Total Beginners.
7. Python: The Complete Reference, Martin C. Brown (Author)

8. WEBSITES

https://www.w3schools.com/python/python_ml_getting_started.asp
https://github.com/
https://jupyter.org/
https://github.com/
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 -

9. 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

1.Ramez Elmasri,
Department of Computer Science and Engineering,
University of Texas, Arligton

2.Temer ozsu,
Department of Computer Science and Engineering,
University Of Alberte, Edmonton, Canada.

NATIONAL

1.Prof. Partha Pratim Das
Department of Computer Science and Engineering, IIT Karagpur

2. Prof. D. Janakiram
Department of Computer Science and Engineering, IIT Madras
3. Dr. S. Srinath
Department of Computer Science and Engineering, IIIT Bangalore

REGIONAL

1.Dr. V. Sidda Reddy,
Prof, Dept. of CSE, CVR College of Engineering, Hyderabad,
Mobile. No. 9849538425

2.Dr. K Kamakshaiah
Associate Professor, Department of CSE, GCET, Hyderabad,
Mobile. No. 8099903505

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

AI-Based Travelling Salesman Problem (TSP)

Single-Page Project Description

Abstract:
The AI-Based TSP System is designed to compute the shortest possible route that visits each city exactly once and returns to the starting point. Since TSP is NP-Hard, exact algorithms become infeasible for large datasets. This system integrates a centralized route database with AI algorithms to provide fast, near-optimal solutions and an interactive virtual workspace for transportation planners and logistics teams.

System Overview:
The system uses a **Database Server Module** for storing city coordinates, distance matrices, and historical route data. The **Client Application Module** includes tools for route navigation, visualization, algorithm execution, and collaborative route planning.

Modules and Functions

1. Route Navigator

- Displays city nodes, distances, and paths in a graphical interface.
- Allows dynamic navigation through SQL-based retrieval of route segments.
- Supports random traversal and AI-guided exploration of the TSP graph.

2. AI Algorithm Engine

Implements several AI/meta-heuristic algorithms:

- **Genetic Algorithm (GA)**: Uses selection, crossover, mutation to evolve routes.
- **Ant Colony Optimization (ACO)**: Uses pheromone updates to converge on shortest paths.
- **Simulated Annealing (SA)**: Gradually optimizes by accepting probabilistic improvements.
- (Optional) **Neural Network / Reinforcement Learning** models.
  The engine compares all results and selects the best route.

3. Route Viewer

- Visualizes the distance matrix, city graph, and optimized path.
- Provides zoom, pan, and step-by-step playback of algorithm progression.
- Allows comparison of multiple algorithm outputs.

4. Sequence Manager

- Manages the step-wise execution of algorithms.
- Supports prefetching of required route data for faster computation.
- Allows custom algorithm sequences for experimentation.

5. Session Manager

- Enables collaborative route-planning sessions.
- Allows users to start, join, or leave sessions.
- Saves best routes, algorithm logs, and recommendations.

6. Synchronization Manager

- Ensures all users see identical route information in real time.
- Handles synchronized fetching of data from the database and aligns algorithm states across clients.

Outputs

- Optimal/near-optimal route
- Total distance/cost
- Visualized path and algorithm performance chart
- Exportable session logs and route reports

Applications

- Logistics and delivery route planning
- Ride-sharing and taxi dispatch optimization
- Drone path planning
- Robotics and warehouse automation
- Smart city transportation systems

11. PROPOSALS INDUSTRIAL VISIT

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

If students are 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		66	CSE-A	
2	MNC	software		66	CSE-B	
3	Level-3	software		66	CSE-C	

12. SHADOW ENGINEERING GUIDELINES FOR SHADOW ENGINEERING (VIP)

OBJECTIVES OF SHADOW ENGINEERING

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

13. CALIBRATION, TESTING & INSPECTION

1. Calibration Procedures

1.1 Environment Calibration

- Ensure the correct **Python version** is installed (e.g., Python 3.10/3.11).
- Validate the default **conda environment** (base) and create isolated environments for each lab/project.
- Verify all core AI/ML libraries:
  - NumPy, Pandas
  - Scikit-learn
  - TensorFlow / PyTorch
  - Matplotlib / Seaborn
  - Jupyter Notebook
- Use conda list --explicit to generate reproducible environment specs.
- Confirm environment paths and interpreter selection inside IDEs (Jupyter, VS Code, PyCharm).

1.2 Package Calibration

- Check that all installed packages are compatible:
  conda update --all
- Validate CUDA & cuDNN compatibility for GPU-related computations.
- Ensure no conflicting versions of libraries are installed.

2. Testing Procedures

2.1 Functional Testing

1. Launch Jupyter Notebook, Spyder, and Python interpreter.
2. Run sample scripts to test:
   - Array operations (NumPy)
   - Data loading (Pandas)
   - Model training (Scikit-learn, TensorFlow, PyTorch)
3. Test GPU support:
4. import torch; print(torch.cuda.is_available())

   or

   import tensorflow as tf; tf.test.is_gpu_available()

2.2 Performance Testing

- Run benchmark programs to measure:
  - Execution speed
  - GPU utilization
  - Training time for small ML models
- Compare performance with baseline logs.

2.3 Dependency Testing

- Verify that environment.yml or requirements.txt produces an identical environment.
- Test creation of new environments:
  conda create -n test_env python=3.10

3. Inspection Procedures

3.1 Software Inspection

- Verify Anaconda Navigator opens without errors.
- Inspect the integrity of environments:
  - No corrupted or partially installed packages
  - No duplicated environments
- Check available disk space (Anaconda environments consume large space).

3.2 Security Inspection

- Scan for untrusted or unwanted pip installations.
- Verify that package downloads use official Anaconda channels (conda-forge, defaults).
- Confirm environment-level permissions in lab computers.

3.3 Update Inspection

- Inspect recent updates using:
  conda info and conda list --revisions
- Ensure all security patches are applied.
- Remove outdated and unused environments periodically.

4. Documentation Requirements

Maintain the following records:

- Installed Anaconda version report
- Environment specification file (environment.yml)
- GPU compatibility logs
- Update history log
- Testing and performance benchmark results
- Monthly inspection checklist

12. MAINTAINANCE

TROUBLESHOOTING:

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	**ANACONDA**	**01/07/2025**	**Soft wares installed**	

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

13. ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL

Accreditation is the formal recognition, authorization and registration of a laboratory that has demonstrated its capability, competence and credibility to carry out the tasks. It provides the feedback to laboratories as to whether they are performing according to the technical competence as per guidelines of NABL (National Accreditation Board for Testing and Calibration Laboratories).

The laboratory should carry out the following important tasks towards getting ready for accreditation from NABL.

1. Preparation of methodology for each experiment
2. Preparation of Standard Operating procedure for each equipment
3. Preparation of Laboratory Manual as per the guidelines specified by Combined Lab Team(CLT) headed by Principal/HOD/Dean/in charge
4. Ensure Effective environmental conditions (temperature, humidity, storage and placement) in the laboratories by implementing proper housekeeping and cleaning of the equipments from dust, dirt etc.
5. Ensure Calibration of instruments/equipment (Only NABL accredited authorized laboratories provide calibration).
6. All the details of Calibration should be included in the format specified exclusively for calibration procedure.
7. Ensure proper implementation of all the documents, formats to be included in the lab manual.
8. Impart training for all the technicians working in labs about the importance of documentation, log sheets, operating procedure of the lab.
9. Incorporate Internal Lab audits for effective functioning of the laboratories. Audits may be once in a month or 3 months or at the end of the semester. The audit schedule will be decided by the Chairman and Principal of the CLT team.
10. Auditors should submit the detailed report of each lab duly signed by them to the Principal.
11. Each lab should maintain all the bills/invoices of each instrument or equipment in a separate file.
12. All the stock registers either consumable or non consumable should be updated whenever any purchases of consumables or equipment takes place.
13. All the safety precautions should be properly displayed in front of each lab.
14. All the LEAD experiments should be maintained separately in a record / recorded in a separate folder.

15. Based on the Pre Assessment report submitted by the auditor, corrective actions should be carried out by each lab in charge and that must be forwarded to concerned HOD and Principal.

16. VIRTUAL LABS REFERENCE LINKS

1. **https://python-iitk.vlabs.ac.in/?utm_source=chatgpt.com**
2. **https://python-iiith.vlabs.ac.in/?utm_source=chatgpt.com**
3. **https://www.vlab.co.in/broad-area-computer-science-and-engineering?utm_source=chatgpt.com**