



Anekant Education Society's
Tuljaram Chaturchand College of Arts, Science & Commerce,
Baramati
(Empowered Autonomous)

Three/Four Year Honours/Honours with Research B.Sc. (Comp. Sci.)
Degree Program in Computer Science
(Faculty of Science)

CBCS Syllabus
T.Y.B.Sc. (Comp. Sci.) Mathematics
For Department of Mathematics

NEP-2.0
Choice Based Credit System Syllabus
(2023 Pattern)

(As Per NEP-2020)

To be implemented from Academic Year 2025-26

CBCS Syllabus as per NEP 2020 for T.Y.B.Sc. (Comp. Sci.) Mathematics (2023 Pattern)

Name of the Programme	: B.Sc. (Comp. Sci.)
Program Code	: USCS
Class	: T.Y.B.Sc. (Comp. Sci.)
Semester	: VI
Course Type	: Minor
Course Name	: Operations Research
Course Code	: COS-361-MN (B)
No. of Teaching Hours	: 30
No. of Credits	: 2

Course Objectives:

1. To introduce the fundamental concepts and formulation techniques of Linear Programming Problems (LPP).
2. To develop problem-solving skills using graphical and simplex methods for LPP.
3. To analyze special cases in LPP, including alternative, unbounded, and infeasible solutions.
4. To understand the concept of duality and learn conversion between primal and dual problems.
5. To study transportation models and apply methods for obtaining initial feasible and optimal solutions.
6. To apply Hungarian and other methods to solve assignment problems efficiently.
7. To enhance analytical and decision-making skills by applying optimization techniques to real-world problems.

Course Outcomes:

CO1: Students will be able to apply the basic concepts and formulations of Linear Programming Problems (LPP) to real-life optimization scenarios.

CO2: Students will solve LPP using graphical and simplex methods to determine optimal solutions.

CO3: Students will be able to analyze special cases in LPP, such as alternative, unbounded, and infeasible solutions, and understand duality concepts.

CO4: Students will be able to formulate the dual of a given primal problem and interpret the relationship between primal and dual solutions.

CO5: Students will develop initial feasible solutions for transportation problems using North-West Corner, Matrix-Minima, and Vogel's Approximation methods.

CO6: Students will be able to optimize transportation problems using the Modified Distribution (MODI) method and handle special cases.

CO7: Students will be able to solve assignment problems using the Hungarian method and address their special cases effectively.

Topics and Learning Points	
Unit 1: Linear Programming Problem	Teaching Hours 08
1.1 Introduction and basic definitions	
1.2 Formulation of LPP	
1.3 Graphical method	
1.4 Simplex method	
Unit 2: Special cases in LPP and concept of duality	08
2.1 Alternative solution	
2.2 Unbounded solution	
2.3 Infeasible solution	
2.4 Duality in Linear Programming	
2.5 Primal to dual conversion	
Unit 3: Transportation Models	08
3.1 Introduction and basic definitions	
3.2 Initial basic feasible solutions using North-West rule, Matrix-minima, Vogel's Approximation	
3.3 Modified distribution method for optimal solution	
3.4 Special cases in transportation problems	
Unit 4: Assignment Models	06
4.1 Introduction and basic definitions	
4.2 Hungarian method for assignment problem	
4.3 Special cases in assignment problems	

Text Book:

J. K. Sharma, *Operations Research: Theory and Applications*, Trinity Press, 6th Edition, 2016

Unit 1: Sections 2.1 to 2.8, 3.1 to 3.4 and 4.1 to 4.4

Unit 3: Sections 9.1 to 9.7

Unit 2: Sections 4.5, 4.6, 5.1 and 5.2

Unit 4: Sections 10.1 to 10.4

Reference Books:

1. H. A. Taha, *Operations Research: An Introduction*, Prentice Hall of India, 8th edition, 2006.
2. S. D. Sharma, *Operations Research*, Kedarnath Ramnath & Company, 1992.
3. D. S. Hira and Prem Kumar Gupta, *Operations Research*, S. Chand and Company Ltd., 2016.
4. R. Panneerselvam, *Operations Research*, PHI Learning Pvt. Ltd., 2nd edition, 2009.
5. H. M. Wagner, *Principles of Operations Research*, PHI Learning Pvt. Ltd., 2nd Edition.

CO-PO Mapping

Weightage: 1 – weak or low relation, 2 – moderate or partial relation, 3 – strong or direct relation

Programme Outcomes	Course Outcomes						
	CO1	CO2	CO3	CO4	CO5	CO6	CO7
PO01	3	3	3	3	3	3	3
PO02	2	3	3	3	3	3	3
PO03	1	1	1	1	2	2	2
PO04	2	3	3	3	3	3	3
PO05	3	3	3	3	3	3	3
PO06	1	2	2	2	2	2	2
PO07	2	2	3	3	2	2	2
PO08	2	2	2	3	2	2	2
PO09	2	3	3	3	3	3	3
PO10	1	1	1	1	2	2	2
PO11	1	1	1	1	2	2	2
PO12	2	2	2	2	3	3	3
PO13	1	1	1	1	2	2	2

Justification for the mapping

PO1: Comprehensive Knowledge and Understanding - All COs require clear understanding of mathematical concepts, formulations, and optimization theory, hence strong linkage.

PO2: Practical, Professional, and Procedural Knowledge - Solving LPP, transportation, and assignment problems involves applying structured methods (Simplex, MODI, Hungarian), reflecting professional-level procedural knowledge.

PO3: Entrepreneurial Mindset and Knowledge - Optimization techniques help in cost minimization and resource utilization in business contexts (CO5–CO7), hence moderate relation for these.

PO4: Specialized skills and competencies - Learners gain specialized skills in linear programming tools and algorithms, used in industries like logistics, supply chain, and operations.

PO5: Capacity for Application, Problem-Solving, and Analytical Reasoning - Every CO emphasizes problem-solving and analytical reasoning — identifying optimal solutions, analysing cases, and interpreting duality.

PO6: Communication skills and collaboration - Explaining optimization results, interpreting solutions, and teamwork in project applications require communication and collaboration.

PO7: Research-related skills - Understanding duality, infeasibility, and optimization algorithms builds analytical and research-oriented thinking for modeling real-life systems.

PO8: Learning how to learn skills - Students continuously learn from iterative problem-solving and algorithmic improvements (especially dual and MODI methods).

PO9: Digital and technological skills - Implementing algorithms through computational tools (Excel Solver, LINGO, Python, R, etc.) strengthens digital competence.

PO10: Multicultural competence, inclusive spirit, and empathy - Optimization in social and economic contexts (e.g., equitable resource allocation) promotes inclusive thinking and fairness.

PO11: *Value inculcation and environmental awareness* - Optimization techniques can support sustainable and resource-efficient systems, linking to value-based and eco-conscious decision making.

PO12: *Autonomy, responsibility, and accountability* - Independent problem-solving and validating optimal solutions foster accountability and responsible decision-making.

PO13: *Community engagement and service* - Optimization methods help in social projects (transport efficiency, public services) fostering community-oriented applications.

CBCS Syllabus as per NEP 2020 for T.Y.B.Sc. (Comp. Sci.) Mathematics (2023 Pattern)

Name of the Programme	: B.Sc. (Comp. Sci.)
Program Code	: USCS
Class	: T.Y.B.Sc. (Comp. Sci.)
Semester	: VI
Course Type	: Minor
Course Name	: Operations Research Practical using Python Programming Language
Course Code	: COS-362-MN (B)
No. of Teaching Hours	: 60
No. of Credits	: 2

Course Objectives:

1. To introduce students to the concepts and formulations of Linear Programming Problems (LPP) for real-world applications.
2. To develop computational skills in solving LPPs using graphical and simplex methods.
3. To enable understanding and handling of special cases in LPP such as multiple, unbounded, and infeasible solutions.
4. To impart knowledge of duality concepts and their practical verification through primal-dual relationships.
5. To provide hands-on experience in solving transportation problems using initial feasible and optimal solution methods.
6. To train students in solving assignment problems using the Hungarian method, including special cases.
7. To build proficiency in using Python programming and optimization libraries for implementing operations research techniques.

Course Outcomes:

- CO1:** Student will be able to formulate real-world decision-making problems into standard Linear Programming Models for optimization.
- CO2:** Student will be able to apply graphical and simplex methods to solve Linear Programming Problems manually and computationally.
- CO3:** Student will be able to analyze special cases in LPP such as multiple solutions, unboundedness, and infeasibility using Python programs.
- CO4:** Student will be able to demonstrate the concept of duality by converting primal problems to dual and verifying solutions.
- CO5:** Student will be able to develop and implement algorithms for solving transportation problems using NW corner, VAM, and MODI methods.
- CO6:** Student will be able to apply optimization techniques to assignment problems using the Hungarian method including special cases.
- CO7:** Student will be able to use Python programming and libraries to model, solve, and interpret results for LPP, transportation, and assignment models effectively.

Topics and Learning Points**Teaching Hours****List of Practical:****60**

- 1) Write a Python program to model real-world optimization problems (profit maximization, cost minimization) into standard LPP form.
- 2) Implement a program to solve a two-variable LPP graphically using Python program.
- 3) Write a program to implement the Simplex algorithm step-by-step without using libraries.
- 4) Solve standard LPPs using Python libraries and compare results with manual simplex.
- 5) Program to identify and display multiple optimal solutions.
- 6) Python script to detect and display conditions where LPP has no finite solution or infeasible constraints.
- 7) Write a program to convert a primal problem to dual automatically and solve both to verify results.
- 8) Implement Python code to generate an initial feasible solution for a transportation problem.
- 9) Write a program to compute the initial feasible solution using Vogel's Approximation.
- 10) Develop Python code to apply the Modified Distribution Method (MODI) for finding the optimal transportation cost.
- 11) Handle cases of unbalanced problems, degeneracy, and multiple optimal solutions using Python.
- 12) Implement the Hungarian Algorithm in Python to solve assignment problems and extend it to special cases (unbalanced, maximization cases).

CO-PO Mapping

Weightage: 1 – weak or low relation, 2 – moderate or partial relation, 3 – strong or direct relation

Programme Outcomes	Course Outcomes						
	CO1	CO2	CO3	CO4	CO5	CO6	CO7
PO01	3	3	2	3	2	2	3
PO02	2	3	3	2	3	3	3
PO03	2	2	1	1	2	2	2
PO04	3	3	3	3	3	3	3
PO05	3	3	3	3	3	3	3
PO06	1	2	2	1	2	2	2
PO07	2	2	3	3	2	2	3
PO08	2	2	2	2	2	2	3
PO09	2	2	3	2	3	3	3
PO10	1	1	1	1	1	2	2
PO11	1	1	1	1	1	1	1
PO12	2	2	2	2	2	2	3
PO13	1	1	1	1	2	2	2

Justification for the mapping

PO1: Comprehensive Knowledge and Understanding - PO1 emphasizes strong understanding of Linear Programming, transportation, and assignment models. CO1, CO2, CO4, and CO7 are strongly related, while CO3, CO5, and CO6 have a moderate relation due to applied problem-solving.

PO2: Practical, Professional, and Procedural Knowledge - PO2 focuses on practical application and professional procedures. CO2, CO3, CO5, CO6, and CO7 are strongly related, while CO1 and CO4 have moderate relation through problem formulation and duality concepts.

PO3: Entrepreneurial Mindset and Knowledge - PO3 highlights decision-making and opportunity recognition. CO1, CO5, CO6, and CO7 have moderate relation by modelling real-world optimization problems, while CO2, CO3, and CO4 are weakly related.

PO4: Specialized skills and competencies - PO4 emphasizes specialized analytical and computational skills. All COs are strongly related as they involve algorithm development, Python programming, and optimization techniques.

PO5: Capacity for Application, Problem-Solving, and Analytical Reasoning - PO5 focuses on applying knowledge to solve problems. All COs are strongly related since the course develops modeling, solving, and analytical reasoning skills.

PO6: Communication skills and collaboration - PO6 involves effective communication and teamwork. CO2, CO3, CO5, CO6, and CO7 are moderately related through documentation and group activities, while CO1 and CO4 have weak relation.

PO7: Research-related skills - PO7 emphasizes skills for research and experimentation. CO3, CO4, and CO7 are strongly related through analysis and algorithm implementation, while CO1, CO2, CO5, and CO6 have moderate relation.

PO8: Learning how to learn skills - PO8 highlights independent learning. CO7 is strongly related due to self-learning of algorithms and Python, while CO1–CO6 have moderate relation through learning problem-solving methods.

PO9: *Digital and technological skills* - PO9 focuses on using technology effectively. CO3, CO5, CO6, and CO7 are strongly related, while CO1, CO2, and CO4 have moderate relation due to partial technological application.

PO10: *Multicultural competence, inclusive spirit, and empathy* - PO10 emphasizes collaboration and inclusion. CO6 and CO7 are moderately related through group activities, while other COs have weak relation due to technical focus.

PO11: *Value inculcation and environmental awareness* - PO11 focuses on ethical and sustainable decision-making. All COs have weak relation as the course mainly emphasizes technical and analytical skills.

PO12: *Autonomy, responsibility, and accountability* - PO12 highlights independent work and accountability. CO7 is strongly related, while CO1–CO6 are moderately related through structured problem-solving activities.

PO13: *Community engagement and service* - PO13 emphasizes societal applications. CO5, CO6, and CO7 have moderate relation through optimization in community or group projects, while other COs are weakly related.