[3] P. Ramamurthy, *Operations Research*, New Age International, 2007.

[4] J. K. Sharma, *Operations Research- Theory and Applications*, Macmillan Publishers, 4th ed 2009.

| | | |
|---|---|---|
| **MTS 515** | **Design and Analysis of Algorithms** | **4 Credits (48 hours)** |

**Course Outcome:** To introduce the concepts and to develop working knowledge on fundamentals algorithm analysis using time complexity. Students will have the knowledge and skills to apply the concepts of the course in algorithm design methodologies in the areas involving logical problem solving including computer science.

**Course Specific Outcome:** At the end of the course students will have the knowledge and skills to understand, explain in depth and apply the fundamental concepts-

- Algorithm analysis, Graph Algorithms
- Divide and conquer, greedy technique, backtracking and dynamic programming
- Dynamic programming, NP-completeness
- Implementing algorithmic strategies on machine with mathematical background.

**Unit I**

Introduction to algorithms, Analyzing algorithms- space and time complexity; growth functions; summations; recurrences; sets, asymptotic etc. Sorting, searching and selection- Binary search, insertion sort, merge sort, quicksort, Radix sort, counting sort, heap sort, etc. Median finding using quick-select, Median of medians.

**(8 Hours)**

**Unit II**

Graph algorithms - Depth-first search; Breadth first search; Backtracking; Branch and bound, etc. Algorithm design - Divide and Conquer: Greedy Algorithms: some greedy scheduling algorithms, Dijkstra's shortest paths algorithm, Kruskal's minimum spanning tree algorithm.

**(16 Hours)**

**Unit III**

Dynamic programming - Elements of dynamic programming, The principle of optimality, The knapsack problem; dynamic programming algorithms for optimal polygon triangulation, optimal binary search tree, longest common subsequence, Shortest paths, Chained matrix multiplication, all pairs of shortest paths.

**(16 Hours)**

**Unit IV**

Introduction to NP-Completeness - Polynomial time reductions, verifications, verification algorithms, classes P and NP, NP-hard and NP-complete problems.

**(8 Hours)**

**References**

[1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stelin, *Introduction to Algorithms*, 3rd Ed., MIT Press, 2009.

[2] T. H. Cormen, C. E. Leiserson, R. L.Rivest and C. Stein. *Introduction to Algorithms*, 2nd ed McGraw-Hill, 2001.

[3] V. Aho, J. E. Hopcroft, J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, 1998.

[4] E. Horowitz, S. Sahni, S. Rajasekaran, *Fundamentals of Computer Algorithms*, University Press (India) Pvt. Ltd., 2009 .

[5] David Harel, *Algorithms*, The spirit of Computing, 3rd Ed., Addison-Wesley, 2004.

[6] Baase S and Gelder, A.V, *Computer Algorithms*, 3rd Ed., Addition- Wesley, 2000.

[7] Garey, M.R, and Johnson, D.S, Computers and Intractability: *A Guide to the Theory of NP-Completeness*, W. H. Freemann & Co, 1976.

[8] M. T. Goodrich and R. Tomassia . *Algorithm Design*: *Foundations, Analysis and Internet examples* , John Wiley and sons, 2001.