

Artificial Intelligence
M.Sc. Diploma Exam – Sample Topics

1. Comparison of network models of Hopfield, Grossberg, Kohonen.
2. Choosing the appropriate neural network architecture for a particular task realization.
3. Resolution method in predicate calculus.
4. Main problems in reasoning about actions.
5. Differences between functions and macros in Lisp.
6. Logic implemented in Prolog. Inference, resolution principle.
7. Methods for finding extrema of a nonlinear function.
8. Methods for finding extrema of a function subject to constraints.
9. Completeness and compactness of metric spaces, the Banach Contraction Principle and applications in optimization.
10. Algorithms complexity.
11. Algorithm design paradigms (dynamic programming, divide and conquer, etc.).
12. Polynomial-time approximation scheme.
13. Basic data structures
 - a) list, queue, b) heap, c) priority queue.
14. Trees
 - a) binary trees, b) balanced trees (e.g. AVL), c) B-trees.
15. Sorting algorithms.
16. Travelling salesman problem – definition, solution algorithms.
17. Algorithms for finding the shortest path in a graph.
18. Deterministic and non-deterministic Turing machine.
19. Regular expressions, finite automata, regular languages.
20. Context free grammars, push down automata, context free languages.

21. Nondeterministic finite automaton.
22. Chomsky hierarchy.
23. Parallel processes synchronization methods
 - a) deadlock, b) starving.
24. Distributed programming. Discuss one of the following algorithms: creating the global time, choosing coordinator, termination of the computation, deadlock avoiding.
25. Interpolation and its applications.
26. Methods for solving systems of linear equations.
27. Methods for finding zeros of functions of one variable.
28. Numerical integration methods.
29. Colour models in computer graphics.
30. Raster image processing methods (contrast enhancement, histogram based operations, spatial image filtration).
31. Relational databases
 - a) normal forms, b) data structures.
32. Basic features of object-oriented languages.
33. Generic programming (idea, generic programming in object-oriented languages).
34. What is UML?
35. Design patterns in object-oriented programming.
36. Software developing models.
37. Goals and methods of software testing.
38. Heuristic search methods.
39. Dictionary as an abstract type/kind of data (operations). Structures used for dictionary implementation.
40. Provide definition of information, examples of source models and measures.
41. Describe Huffman codes (algorithm, optimality, efficiency limits).
42. Give examples of lossy compression codings.