CS-CO-412(?) COURSE DESCRIPTION - Spring 2016

|  |  |
| --- | --- |
| **Course Information** | |
| **Course Number:** | CS-CO-412(?) |
| **Course Title:** | Discrete Mathematics |
| **Credits:** | (6-0-6)[[1]](#footnote-1) |
| **Instructor:** | Adil Mehmood Khan (a.khan@innopolis.ru) |
| **Grading Mode:** | Normal (A-F) |
| **Prerequisite(s):** | None |

**Knowledge Areas that contain topics covered in the course**

* Sets, Relations, and Functions
* Logic
* Proof Techniques
* Counting
* Graph Theory
* Discrete Probability
* Basic Machine Learning (Tentative)

**Course outline**

This course presents material in Discrete Mathematics, which provides a foundation for computer science that many other areas of computer science require. The material covered in this course is pervasive in the areas of data structures and algorithms but appears elsewhere in computer science as well. Topics covered in this course include set theory, logic, graph theory, probability theory, and basic machine learning. The part on machine learning will be included specially to show the application of discrete math to the analysis of machine learning problems.

**Required background knowledge:**

There are no formal prerequisites for this course. However, a strong familiarity with high school mathematics is assumed.

**Course Syllabus:**

Topics are listed separately for each knowledge area along with the learning outcomes that students are expected to achieve with respect to the topics covered.

1. **Sets, Relations, and Functions**
   1. Topics Covered

* Venn diagrams
* Union, intersection, complement
* Cartesian product
* Power sets
* Cardinality of finite and infinite sets
* Reflexivity, symmetry, transitivity
* Equivalence relations, partial orders
* Surjections, injections, bijections
* Inverses
* Composition
  1. Learning Outcomes
* Explain with examples the basic terminology of functions, relations, and sets.
* Perform the operations associated with sets, functions, and relations
* Relate practical examples to the appropriate set, function, and relation model, and interpret the associated operations and terminology in context.

1. **Logic**
   1. Topics Covered

* Propositional logic
* Logical connectives
* Truth tables
* Propositional inference rules
* Predicate logic
  1. Learning Outcomes
* Convert logical statements from informal language to propositional and predicate logic expressions.
* Use the rules of inference to construct proofs in propositional and predicate logic.
* Apply formal logic proofs and/or informal, but rigorous, logical reasoning to real problems.

1. **Proof Techniques**
   1. Topics Covered

* Notions of implications, equivalence, converse, inverse, contrapositive, negations and contradiction
* The structure of mathematical proofs
* Direct proofs
* Disproving by counterexample
* Proof by contradiction
* Mathematical induction
* Weak and strong induction
  1. Learning Outcomes
* Identify the proof technique used in a given proof.
* Outline the basic structure of each proof technique.
* Apply each of the proof technique.
* Determine which type of proof is appropriate for a given problem.
* Explain the relationship between weak and strong induction and give examples of the appropriate use of each.

1. **Counting**
   1. Topics Covered

* Set cardinality and counting
* Sum and product rule
* Arithmetic and geometric progressions
* The pigeonhole principle
* Permutations and combinations
  1. Learning Outcomes
* Apply counting arguments, including sum and product rules, and arithmetic/geometric progressions.
* Compute permutations and combinations of a set, and interpret the meaning in the context of the particular application.
* Map real-world applications to appropriate counting formalisms.

1. **Graph Theory**
   1. Topics Covered

* Undirected graphs
* Directed graphs
* Weighted graphs
* Spanning trees/forests
* Graph isomorphism
  1. Learning Outcomes
* Illustrate by example the basic terminology of graph theory, as well as some of the properties and special cases of each type of graph.
* Show how concepts from graphs appear in data structures, algorithms, proof techniques, and counting.
* Determine if two graphs are isomorphic.

1. **Discrete Probability**
   1. Topics Covered

* Finite probability space, events
* Conditional probability, Bayes’ theorem
* Independence
* Expectation
* Variance
* Conditional independence
  1. Learning Outcomes
* Calculate probabilities of events and expectations of random variables for elementary problems.
* Differentiate between dependent and independent events.
* Apply Bayes’ theorem to determine conditional probabilities in a problem.
* Compute the variance for a given probability distribution.

1. **Basic Machine Learning**
   1. Topics Covered

* Definition and examples of broad variety of machine learning tasks, including classification and clustering
* Inductive learning
* Naïve Bayesian Classifier, decision trees
* The over-fitting problem
* Measuring classifier accuracy
  1. Learning Outcomes
* List the differences among the three main styles of learning: supervised, unsupervised, and reinforcement.
* Identify examples of classification tasks, including the available input features, and output to be predicted.
* Apply a simple learning algorithm such as Naïve Bayesian classifier to a classification task and the measure the classifier’s accuracy.

**Textbook:** Susanna S. Epp, *Discrete Mathematics with Applications*, 4th Edition.

**Reference Materials**

Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, 4th Edition.

**Course Delivery:** This course will be given over a period of four months. There will be a 2-hour lecture every week, followed by a 2-hour tutorial session. Tutorial exercises will be set periodically. There will be weekly homework, three one-hour tests, a mid-term exam, and a final examination.

**Laboratory Resources:** No laboratory resources are required for this course.

**Assessment:** Three One-hour Tests (60%), Mid-term Exam (20%), and Final Exam (20%).

1. (Credit-hours – Laboratory hours per week – Lecture hours per week) [↑](#footnote-ref-1)