1.9 Logic and Discrete Math

- **Course name:** Logic and Discrete Math
- **Course number:** XYZ

1.9.1 Course Characteristics

This course consists of two distinct but overlapping parts: i. Logic; and ii. Discrete Mathematics.

The first part of the course is an introduction to formal symbolic logic. Philosopher John Locke once wrote that "logic is the anatomy of thought." This part of the course will teach students to analyse and evaluate arguments using the formal techniques of modern symbolic logic.

The second part of the is designed for students to teach them basic notions of graph theory, discrete optimization and dynamic programming. This part will give practical experience with basic algorithms in discrete mathematics.

1.9.1.1 Key concepts of the class

- Deductive Reasoning
- Inductive Reasoning
- Categorical Logic
- Propositional Logic
- Predicate Logic
- Basic elements of Graph Theory
- Planar Graphs
- Algorithms on Graphs

1.9.1.2 What is the purpose of this course?

The first part of the course (on Logic) sets students upon a path of finely honed logical skills essential for life in the modern world. This part of the course is thus specifically designed to improve writing, thinking and oral presentation skills that are applicable to all areas of academic study and relevant to working life. The Discrete Math part of the course introduces students to the basic concepts of structuring algorithms through graph theory. This gives students the skills of deep algorithmic thinking. Thus, the successful completion of the course by students will give them the skills for further deep mastering of computer science courses.

1.9.1.3 Course objectives based on Bloom's taxonomy

- What should a student remember at the end of the course?

By the end of the course, the students should be able to know the basics of Logic Reasoning. In particular, they should be able to remember:

- how to use Venn diagrams
- how to calculate truth values
- the formal structures of some arguments
- the difference between deduction and induction
- how to build spanning trees
- how to find Euler tour and Hamilton path
- how to use Dijkstra's algorithm
- how to solve maximum flow problem

- What should a student be able to understand at the end of the course?

By the end of the course, the students should be able to understand:

- what is categorical logic
- what is propositional logic
- what is predicate logic
- what is the difference between deduction and induction
- what is trees and spanning trees
- what is Euler and Hamilton graphs
- what is planar graphs
- Dijkstra's algorithm

- What should a student be able to apply at the end of the course?

By the end of the course, the students should be able to apply the content of this course in all areas of their academic study but also in their working life, with special reference to the following disciplines:

- computer science
- engineering
- robotics

1.9.1.4 Course evaluation

Table 1.25: Course grade breakdown

Type	Default points	Proposed points
Final Exam (Written, Math)	40	40
Mid Term (Written, Logic)	40	40
Participation/Attendance	20	20

Marks Final Exam – Written – on Math (40The final exam will consist of theoretical and practical questions. The exam will be 90 minutes long.

Mid-Term Exam – Written – on Logic (30The midterm exam will consist of quizzes and various exercises (such as open theoretical questions, multiple choices, and true/false questions). The Mid-Term will be held in class on week 8. The exam will last 75 minutes.

Attendance/Participation Weekly exercises (hereby reports) must be uploaded on moodle the day after the lab sessions are over. Although reports are not marked, they directly count towards participation. Failure to submit reports will incur in a penalty of the course mark, which will be proportional to the number of reports missing.

1.9.1.5 Grades range

Table 1.26: Course grading range

Grade	Default range	Proposed range
A. Excellent	85-100	85-100
B. Good	75-84	75-84
C. Satisfactory	60-74	60-74
D. Poor	0-59	0-54

1.9.1.6 Resources and reference material

There is no mandated textbook. Recommended text books are:

- Hurley, P. J. (2014). A Concise Introduction to Logic. Wadsworth Pub Co
- Priest, G. (2017). Logic: A very short introduction. Oxford, UK: Oxford University Press.
- Lehman, E., Leighton, F. T., Meyer, A. R. (2017). Mathematics for Computer Science. Massachusetts Institute of Technology Press.

Electronic copies of these books will be available on Moodle for download. All the materials for the course will be uploaded by the Professors on Moodle in due time. Please note: The lectures will be highly interactive and the students may be interrogated during the lectures, sometimes via kahoots (this approach is known as flipped learning).

1.9.2 Course Sections

The main sections of the course and approximate hour distribution between them is as follows:

Table 1.27: Course Sections

Section	Section Title	Teaching Hours
1	Basic Concepts	?
2	Fundamentals of Logic	?
3	Undirected graphs	?
4	Directed graphs	?

1.9.2.1 Section 1

Section title: Basic Concept

Topics covered in this section:
- Introduction to Categorical Logic
- Quantity
- Quality
- Distribution
- Venn Diagrams
- Fundamentals of Stoic Logic

What forms of evaluation were used to test students' performance in this section?

Form	Yes/No
Development of individual parts of software product code	0
Homework and group projects	1
Midterm evaluation	1
Testing (written or computer based)	1
Reports	1
Essays	0
Oral polls	1
Discussions	1

Typical questions for ongoing performance evaluation within this section

1. Determine Quantity in categorical propositions

2. Determine Quality in categorical propositions

3. Determine Distribution in categorical propositions

4. Understand the difference between deduction and induction

5. Use Venn Diagrams to represent categorical propositions

Typical questions for seminar classes (labs) within this section

1. Exercises to understand the importance of Categorical Logic for Arguments' Assessment

2. What is Categorical Logic used for?

3. What are the four types of categorical propositions?

4. What are the differences between Stoic Logic and Aristotelian Logic

Test questions for final assessment in this section

1. Solve exercises on Categorical Logic.

2. Understand the importance of Categorical Logic in thinking and reasoning

3. Evaluate categorical propositions

1.9.2.2 Section 2

Section title: Fundamentals of Logic

Topics covered in this section:

- Introduction to Propositional Logic
- Logical Operators
- Truth Tables for Propositions
- Truth Tables for Arguments
- Introduction to Predicate Logic
- Quantifiers
- Translation

What forms of evaluation were used to test students' performance in this section?

Form	Yes/No
Development of individual parts of software product code	0
Homework and group projects	1
Midterm evaluation	1
Testing (written or computer based)	0
Reports	1
Essays	0
Oral polls	1
Discussions	1

Typical questions for ongoing performance evaluation within this section

1. Solve Truth Tables
2. Use Truth Tables to analyse arguments
3. Use Quantifiers to assess inferences
4. What is Propositional Logic used for?
5. What is Predicate Logic used for?

Typical questions for seminar classes (labs) within this section

1. Exercises to understand the potential implications of Propositional Logic for Computer Science
2. Exercises to get a glimpse of the importance of Predicate Logic in Science (e.g. Mathematics)
3. Exercises to understand the relation between connectors and truth tables.

1.9.2.3   Test questions for final assessment in the course

1. What is the difference between Categorical and Propositional Logic?
2. How does Predicate Logic differ from Categorical and Propositional Logic?
3. Why is Predicate Logic so important?
4. What are Truth-Functions and why do we use them?
5. Compute True Tables for Propositions
6. Compute True Tables for Arguments

1.9.2.4   Section 3

Section title: Undirected graphs

Topics covered in this section:

- Basics elements of Graph Theory
- Trees
- Euler tours
- Hamilton paths
- Planar graphs

What forms of evaluation were used to test students' performance in this section?

Form	Yes/No
Development of individual parts of software product code	0
Homework and group projects	1
Midterm evaluation	1
Testing (written or computer based)	0
Reports	0
Essays	0
Oral polls	1
Discussions	1

Typical questions for ongoing performance evaluation within this section

1. What is the characteristic property of trees?
2. How to find an Euler tour?
3. What is a Hamilton path?
4. Why are K3 and K5,5 not planar?

Typical questions for seminar classes (labs) within this section

1. Exercises to build spanning trees.
2. Exercises to find an Euler tour/cycle and a Hamilton path/cycle or determine that one of them or both do not exist.
3. Exercises to determine if a graph is planar.

1.9.2.5 Test questions for final assessment in the course

1. Explain handshaking lemma.
2. Give necessary and sufficient conditions for the existence of an Euler tour.
3. Give sufficient conditions for the existence of a Hamilton path (theorems of Dirac and Ore).
4. Explain Kuratowski's theorem.

1.9.2.6 Section 4

Section title: Directed graphs

Topics covered in this section:
- Introduction to Directed Graphs
- Introduction to Weighted Graphs
- Dijkstra's algorithm
- Maximum flow problem

What forms of evaluation were used to test students' performance in this section?

Form	Yes/No
Development of individual parts of software product code	0
Homework and group projects	1
Midterm evaluation	0
Testing (written or computer based)	1
Reports	0
Essays	0
Oral polls	1
Discussions	1

Typical questions for ongoing performance evaluation within this section

1. What is the difference between undirected and directed graphs?
2. Why do we consider weighted graphs?
3. What practical problems are solved using Dijkstra's algorithm?
4. What is the maximum flow problem?

Typical questions for seminar classes (labs) within this section

1. Introduction to Directed Graphs
2. Introduction to Weighted Graphs
3. Dijkstra's algorithm
4. Maximum flow problem

Test questions for final assessment in this section

1. Explain the difference between undirected and directed graphs.
2. Give the definition of weighted graphs?
3. Explain Dijkstra's algorithm?
4. What is the solution of the maximum flow problem (the Ford-Fulkerson algorithm)?