TED UNIVERSITY, COURSE SYLLABUS

Faculty	Engineering	**Department**	Computer Engineering

Course Code & Number	**CMPE 475**	**Course Title**	**Category Theory in Computer Science**
Type of Course	☐ Compulsory ☑ Elective	**Semester**	☑ Fall ☐ Spring ☐ Summer
Level of Course	BSc	**Year of Study**	Senior
Course Credit Hours	(3+0+0) 3	**Number of ECTS Credits**	6
Pre-requisite	N/A	**Co-requisite**	N/A
Mode of Delivery	☑ Face-to-face ☐ Distance learning	**Language of Instruction**	☑ English ☐ Turkish
Course Lecturer	Burak Ekici	**Course Assistant**	NONE
Required Reading	[B1] Michael Barr and Charles Wells, Category Theory For Computer Science. [B2] Steve Awodey, Category Theory.	**Recommended Reading**	[B3] Bartosz Milewski, Category Theory for Programmers. [B4] Saunders Mac Lane, Categories for the Working Mathematician.

Course Catalog Description	Introduction and history. Categories. The category of sets and functions. Commutative diagrams. Category theoretic isomorphism. Initial and terminal objects. Monoids. (co) Products. Exponential objects. Cartesian closed categories (CCC). Constructive Logic with natural deduction and its categorical interpretation. Simply typed lambda calculus and its semantics in CCCs. The Curry-Howard-Lambek correspondence. Functors. Contravariance. Functor compositions. Natural transformations. Functor categories. Hom-functors. Adjunctions. Kleisli adjunctions. Eilenberg-Moore adjunctions. Dependent types and their categorical interpretations. The Yoneda lemma. Monads with applications in functional programming.	
Course Objectives	Category theory unifies mathematical properties of structures in terms of morphisms bridging them. It provides a precise framework when it comes to compare different domains of mathematics (constructed as categories) so as to transform problems of one domain into the other. Following Moggi's seminal idea of modeling programming languages with computational side effects employing monads, category theoretic constructions made good ground on computer science. Within the scope of this course, we aim at 1. introducing basic notions of category theory: categories, functors, natural transformations, monads and adjunctions, and 2. employing such notions to organize and develop structures arising in semantics of programming languages and logical reasoning systems.	
Course Learning Outcomes	Upon successful completion of this course, a student will be able to 1. observe roles of category theoretical objects in programming paradigms, especially in functional programming 2. program with monads in Haskell and/or in OCaml 3. perform semantic modeling of programming languages from the category theoretic viewpoint 4. reason in categorical logic and mechanize such reasonings in the Coq proof assistant	
Teaching Methods & Learning Activities	☑ Telling/Explaining ☑ Discussions/Debates	☐ Simulations & Games ☐ Video Presentations

	☑ Questioning ☑ Reading ☐ Peer teaching ☐ Scaffolding/Coaching ☑ Demonstrating ☑ Problem solving ☐ Inquiry ☐ Collaborating ☐ Think-Pair-Share ☐ Predict-Observe-Explain ☐ Microteaching ☑ Case Study/Scenario Analysis	☐ Oral presentations/Reports ☐ Concept Mapping ☐ Brainstorming ☐ Drama/Role Playing ☐ Seminars ☐ Field Trips ☐ Guest Speakers ☑ Hands-on Activities ☐ Service Learning ☐ Web Searching ☑ Experiments ☐ Other(s): .....
Assessment Methods (Formal & Informal)	☐ Test/Exam ☑ Quiz/Homework ☐ Oral Questioning ☐ Laboratory work ☐ Performance Project	☐ Observation ☐ Self-evaluation ☐ Peer-evaluation ☐ Portfolio ☐ Presentation (Oral, Poster) ☐ Other(s): .....

Student Workload (Total 147 Hrs)	☑ Lectures ..... 42 hrs ☑ Course Readings ..... 30 hrs ☐ Workshop ..... hrs ☐ Online Discussion ..... hrs ☐ Debate ..... hrs ☐ Work Placement ..... hrs ☐ Field Trips/Visits ..... hrs ☐ Observation ..... hrs ☐ Laboratory Applications ..... hrs ☑ Quizzes ..... 15 hrs ☐ Hands-on Work ..... hrs ☑ Homework .....60 hrs	☐ Midterm I ..... hrs ☐ Midterm II ..... hrs ☐ Final ..... hrs ☐ Resource Review ..... hrs ☐ Research Review ..... hrs ☐ Report on a Topic ..... hrs ☐ Case Study Analysis ..... hrs ☐ Oral Presentation ..... hrs ☐ Poster Presentation ..... hrs ☐ Demonstration ..... hrs ☐ Web Designs ..... hrs ☐ Mock Designs ..... hrs ☐ Team Meetings ..... hrs ☐ Other ..... hrs

COURSE ASSIGNMENTS
A. Homeworks [80 points]
There are 4 homeworks; 20 points each.
B. Quizzes [20 points]
There are 3 quizzes; 10 points each. The lowest grade among all is discarded.

COURSE GRADING
A. Quizzes and Homeworks [30%]
B. Midterm [30%]
C. Final [40%]

COURSE POLICIES

I. Attendance

Attendance to the course is mandatory.
The students attending less than 70% of Lecture Hours will get "FX" grade.

II. Missed Work

III. Late Assignment Submission Policy

Late submissions will not be graded

IV. Extra Credit

Extra credits will not be offered.

V. Assignment Rules

All assignment works must be done individually. A student can submit only one work. In case of multiple submissions, only the latest submission will be considered. Students cannot submit work on other students' behalf.

VI. Plagiarism

All of the following are considered plagiarism:

- turning in someone else's work as your own
- copying words or ideas from someone else without giving credit
- failing to put a quotation in quotation marks
- giving incorrect information about the source of a quotation
- changing words but copying the sentence structure of a source without giving credit
- copying so many words or ideas from a source that it makes up the majority of your work, whether you give credit or not" (www.plagiarism.org)

Plagiarism is a very serious offense and will be penalized accordingly by the university disciplinary committee. The best way to avoid accidentally plagiarizing is to work on your own before you ask for the help of other resources.

VII. Cheating

Cheating has a very broad description which can be summarized as "acting dishonestly". Some of the things that can be considered as cheating are the following:

- Copying answers on examinations, homework and laboratory works,
- Using prohibited material on examinations,
- Lying to gain any type of advantage in class
- Providing false, modified or forged data in a report
- Plagiarizing
- Modifying graded material to be regraded.
- Causing harm to colleagues by distributing false information about an examination, homework or laboratory

Cheating is a very serious offense and will be penalized accordingly by the university disciplinary committee.

VIII. Class Participation

Participation in class is necessary but not mandatory. By actively participating in class, you can improve your learning process and immediately confirm what you have earned and what you have not internalized. Do not forget that you are not expected to know all of the material being discussed in class. Actually, you are expected not to know it. Therefore, there is no point in being hesitant to join a conversation or ask a question.

IX. Class Readings

Class readings are necessary but not mandatory. The material covered in class by your instructor will only provide a fundamental understanding of the general context. If you are willing to effectively learn something, you must actively work on it yourself. Reading is one of the most successful ways of learning about a topic.

TENTATIVE COURSE OUTLINE				
		Topic	**Reading**	**Homeworks / Exams**
W0	04.10 – 08.10	Introduction to Categories The Coq Proof Assistant	• Chapter 2.1 [B1] • Chapter 2.2 [B1] • Chapter 1.1 [B2] • Chapter 1.3 [B2]	
W1	11.10 – 15.10	The Category of Sets and Functions Commutative Diagrams Categorical Isomorphism	• Chapter 2.3 [B1] • Chapter 2.4 [B1] • Chapter 4.1 [B1] • Chapter 1.5 [B2]	
W2	18.10 – 22.10	Duality Principle Initial and Terminal Objects	• Chapter 2.7 [B1] • Chapter 3.1 [B2] • Chapter 2.2 [B2]	
W3	25.10 – 29.10	Binary (co)Products Exponential Objects Cartesian Closed Categories (CCC)	• Chapter 5.1 [B1] • Chapter 6.1 [B1] • Chapter 6.2 [B1] • Chapter 3.2 [B2] • Chapter 6.1 [B2] • Chapter 6.2 [B2]	*HW-1*
W4	01.11 – 05.11	(Natural Deduction Style) Constructive Logic (CL) Semantics of CL in a Cartesian Closed Preorder	• Chapter 5.5 [B1] • Chapter 5.6 [B1]	*Quiz-1*
W5	08.11 – 12.11	Simply Typed Lambda Calculus (STLC) STLC Types and Terms in CCC The Curry-Howard-Lambek Correspondence	• Chapter 6.3 [B1] • Chapter 6.4 [B1] • Chapter 6.5 [B1] • Chapter 6.5 [B2]	
W6	15.11 – 19.11	Functors and Contravariance Functor Composition	• Chapter 3.1 [B1] • Chapter 3.2 [B1] • Chapter 3.3 [B1] • Chapter 7.1 [B2]	*HW-2*
W7	22.11 – 26.11	**Midterm Exam**		
W8	29.11 – 03.12	The Size of a Category (Locally) Small Categories Finite Products of Categories	• Chapter 2.1 [B1] • Chapter 1.8 [B2]	
W9	06.12 – 10.12	Natural Transformatios Functor Categories	• Chapter 4.2 [B1] • Chapter 4.3 [B1] • Chapter 7.4 [B2] • Chapter 7.5 [B2] • Chapter 7.7 [B2]	*Quiz-2*
W10	13.12 – 17.12	Hom-Functors Adjunctions Kleisli Adjunctions Eilenberg-Moore Adjunctions	• Chapter 3.1 [B1] • Chapter 13.2 [B1] • Chapter 9.1 [B2] • Chapter 9.2 [B2] • Chapter 9.3 [B2]	*HW-3*
w11	20.12 – 24.12	The Yoneda Lemma	• Chapter 4.5 [B1] • Chapter 8.1 [B2] • Chapter 8.2 [B2] • Chapter 8.3 [B2] • Chapter 8.4 [B2]	
W12	27.12 – 31.12	The Yoneda Lemma	• Chapter 4.5 [B1] • Chapter 8.1 [B2] • Chapter 8.2 [B2] • Chapter 8.3 [B2] • Chapter 8.4 [B2]	*Quiz-3*
W13	03.01 – 07.01	Monads in Functional Programming	• Philip Wadler,	*HW-4*

			Monads for Functional Programming	
W14	10.01 – 14.01	**Final Exam**		